



JN516x-EK004 Evaluation Kit User Guide

JN-UG-3108
Revision 2.0
20 June 2016

**JN516x-EK004 Evaluation Kit
User Guide**

Contents

Preface	5
Organisation	5
Conventions	5
Acronyms and Abbreviations	6
Related Documents	7
Support Resources	7
Optional Components	7
Trademarks	8
1. Introduction to the Evaluation Kit	9
1.1 Kit Contents	10
1.2 Wireless Network Fundamentals	12
1.2.1 Radio Frequency	12
1.2.2 Network Identification	13
1.2.3 Node Types	13
1.2.4 Network Formation and Topology	14
1.2.4.1 Star Networks	14
1.2.4.2 Tree Networks	15
1.2.4.3 Mesh Networks	15
1.2.5 Wireless Network Protocol Stack	16
1.3 Internet of Things (IoT)	17
1.3.1 IoT Gateway	17
1.3.2 IP Connectivity of the Kit	18
1.4 Network Commissioning	19
1.4.1 NFC Commissioning Process	20
1.4.2 Addressing, Binding and Grouping	21
1.5 ZigBee Smart Home Demonstration	22
2. Using the Smart Home Demonstration	23
2.1 Demo System Overview	23
2.2 Setting Up the IoT Gateway	25
2.3 Assembling the Nodes	28
2.4 Commissioning the Nodes	30
2.5 Controlling the Light	34
2.5.1 Controlling the Dimmable Light from the Dimmer Switch	34
2.5.2 Controlling the Dimmable Light from the PC	35

3. Where Next?	37
3.1 Which Protocol?	37
3.2 Software Developer's Kit (SDK)	38
3.2.1 'BeyondStudio for NXP' Toolchain (JN-SW-4141)	38
3.2.2 IEEE 802.15.4 SDK (JN-SW-4163)	39
3.2.3 ZigBee Home Automation SDK (JN-SW-4168)	39
3.2.4 ZigBee Light Link SDK (JN-SW-4168)	39
3.2.5 ZigBee 3.0 SDK (JN-SW-4170)	40
3.3 Support Resources	41
3.3.1 IEEE 802.15.4 Documentation	41
3.3.2 ZigBee PRO/Home Automation Documentation	42
3.3.3 ZigBee PRO/ZigBee Light Link Documentation	43
3.3.4 ZigBee 3.0 Documentation	44
Appendices	47
A. Kit Hardware	47
A.1 Carrier Boards	48
A.1.1 Power Source Selection	51
A.1.2 Pre-assembled Boards	53
A.2 Lighting/Sensor Expansion Boards	55
A.3 Generic Expansion Boards	57
A.4 JTAG Expansion Board	59
A.5 JN5169 Modules	61
A.6 USB Dongles	61
A.7 Raspberry Pi	63
A.8 NFC Controller	65
A.9 NTAG I2C plus Antenna Boards and Adaptors	66
B. Installing a Packet Sniffer	68
C. Firmware Re-programming	69
C.1 Re-programming JN5169 Modules	69
C.2 Re-programming JN5169 USB Dongles	69
D. Installing the FTDI Device Driver for USB Connections	70

Preface

This manual provides an introduction to the NXP JN516x-EK004 ZigBee Evaluation Kit, based around the JN516x family of wireless microcontrollers. The manual also describes how to run the pre-loaded ZigBee Smart Home Demonstration using components from the kit. In addition, information is provided on the wireless network protocols supported by the evaluation kit and the NXP resources available to develop custom JN516x applications using these protocols.

Users with the pre-production version of the kit should refer to v1.0 of this User Guide.

Organisation

This manual consists of three chapters and four appendices, as follows:

- [Chapter 1](#) introduces the JN516x-EK004 Evaluation Kit and outlines the key concepts for an understanding of wireless networks.
- [Chapter 2](#) describes how to use the evaluation kit to set up and run the pre-loaded ZigBee Smart Home Demonstration.
- [Chapter 3](#) provides guidance on where to go next in order to develop custom JN516x applications that can be run on the components of the kit.
- The [Appendices](#) provide:
 - Details of the individual kit components
 - Guidance on setting up a packet sniffer using one of the supplied dongles
 - Guidance on re-programming the firmware of certain kit components
 - Instructions for installing the FTDI device driver for USB connections

Conventions

Files, folders, functions and parameter types are represented in **bold** type.

Function parameters are represented in *italics* type.

Code fragments are represented in the `Courier New` typeface.



This is a **Tip**. It indicates useful or practical information.



This is a **Note**. It highlights important additional information.



This is a **Caution**. It warns of situations that may result in equipment malfunction or damage.

Acronyms and Abbreviations

ADC	Analogue to Digital Converter
API	Application Programming Interface
DC	Direct Current
DHCP	Dynamic Host Configuration Protocol
DIO	Digital Input/Output
FTDI	Future Technology Devices International
GPIO	General Purpose Input/Output
HA	Home Automation
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IP	Internet Protocol
JTAG	Joint Test Action Group
LAN	Local Area Network
LED	Light Emitting Diode
LPRF	Lower Power Radio Frequency
NFC	Near Field Communication
PSU	Power Supply Unit
PWM	Pulse Width Modulation
RF	Radio Frequency
RTOS	Real Time Operating System
SBC	Single Board Computer
SDK	Software Developer's Kit
SMA	SubMiniature version A
SSID	Service Set Identifier

UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
WAN	Wide Area Network
WPAN	Wireless Personal Area Network
ZCL	ZigBee Cluster Library
ZLL	ZigBee Light Link

Related Documents

JN-UG-3098	BeyondStudio for NXP Installation and User Guide
JN-UG-3099	JN51xx Production Flash Programmer User Guide
JN-UG-3112	NFC Commissioning User Guide
JN-AN-1221	ZigBee HA Lighting with NFC Application Note
JN-AN-1222	ZigBee IoT Gateway Host with NFC Application Note
JN-AN-1223	ZigBee IoT Gateway Control Bridge Application Note
JN-RD-6036	Carrier Board and Expansion Boards Reference Design

Support Resources

To access online support resources such as SDKs, Application Notes and User Guides, visit the Wireless Connectivity area of the NXP web site:

www.nxp.com/products/interface-and-connectivity/wireless-connectivity

All NXP resources referred to in this manual can be found at the above address, unless otherwise stated.

Optional Components

The following additional/optional hardware components are available for use with the JN516x-EK004 Evaluation Kit:

Part Number	Name
JN5169XK010	ZigBee Generic Node Expansion Kit
JN5169XK020	ZigBee Lighting/Sensor Node Expansion Kit
OM15020	JN5169 USB Dongle

These products can be individually ordered via the NXP web site (www.nxp.com).

Trademarks

All trademarks are the property of their respective owners.

1. Introduction to the Evaluation Kit

Welcome to the JN516x-EK004 ZigBee Evaluation Kit, which is based around the NXP JN516x family of wireless microcontrollers. A ZigBee Smart Home Demonstration is pre-loaded into certain kit components, allowing a small wireless network with IP connectivity to be quickly assembled and used. The evaluation kit can also be used in conjunction with NXP's range of Software Developer's Kits (SDKs) to develop new wireless network applications for the JN516x devices. Support resources are also provided to allow the integration of a network into the 'Internet of Things' (IoT).

The wireless network protocols supported by the evaluation kit are as follows:

- IEEE 802.15.4
- ZigBee PRO with Home Automation (HA) profile
- ZigBee PRO with ZigBee Light Link (ZLL) profile
- ZigBee 3.0 with ZigBee Lighting & Occupancy (ZLO) devices



Note: The pre-loaded ZigBee Smart Home Demonstration is based on ZigBee PRO with the Home Automation profile.

This chapter introduces you to the evaluation kit and wireless networks by describing:

- Contents of the kit - see [Section 1.1](#)
- Wireless network fundamentals - see [Section 1.2](#)
- IP connectivity - see [Section 1.3](#)
- Network commissioning - see [Section 1.4](#)
- Pre-loaded ZigBee Smart Home Demonstration - see [Section 1.5](#)



Note 1: Details of the individual hardware components of the kit are provided in [Appendix A](#).

Note 2: If you wish to run the pre-loaded ZigBee Smart Home Demonstration, go to [Chapter 2](#). A Quick Start procedure for running this demonstration is also provided on the *JN516x-EK004 Getting Started sheet (JN-UG-3107)*, which is supplied in the kit box.

Note 3: Users with the pre-production version of the kit should refer to v1.0 of this User Guide.

1.1 Kit Contents

The JN516x-EK004 evaluation kit contains the hardware components required to assemble a wireless network which may be connected to an IP-based network (such as the Internet). The principal network components included in the kit are:

- **Carrier Boards** which host a JN5169 module and which can be fitted with the supplied Expansion Boards (see below) to achieve the desired functionality - the supplied NFC tag board can be attached to facilitate NFC commissioning
- **Expansion Boards** which can be mounted on the Carrier Boards in order to provide them with the required functionality: Lighting/Sensor or Generic
- **Dongles** including JN5169 USB Dongles (featuring the JN5169 wireless microcontroller) and a Wi-Fi adaptor for the Raspberry Pi board (see below)
- **Raspberry Pi Computer** which, together with a JN5169 USB Dongle, forms an IoT Gateway that allows the wireless network to be connected to an IP-based network, including the Internet.
- **NFC Controller Board** which connects to the Raspberry Pi board and can be used to read the supplied NFC tag to implement NFC commissioning
- **NFC Tag Board** which connects to a Carrier Board to facilitate the NFC commissioning of the node

The contents of the kit are shown in the photograph below.

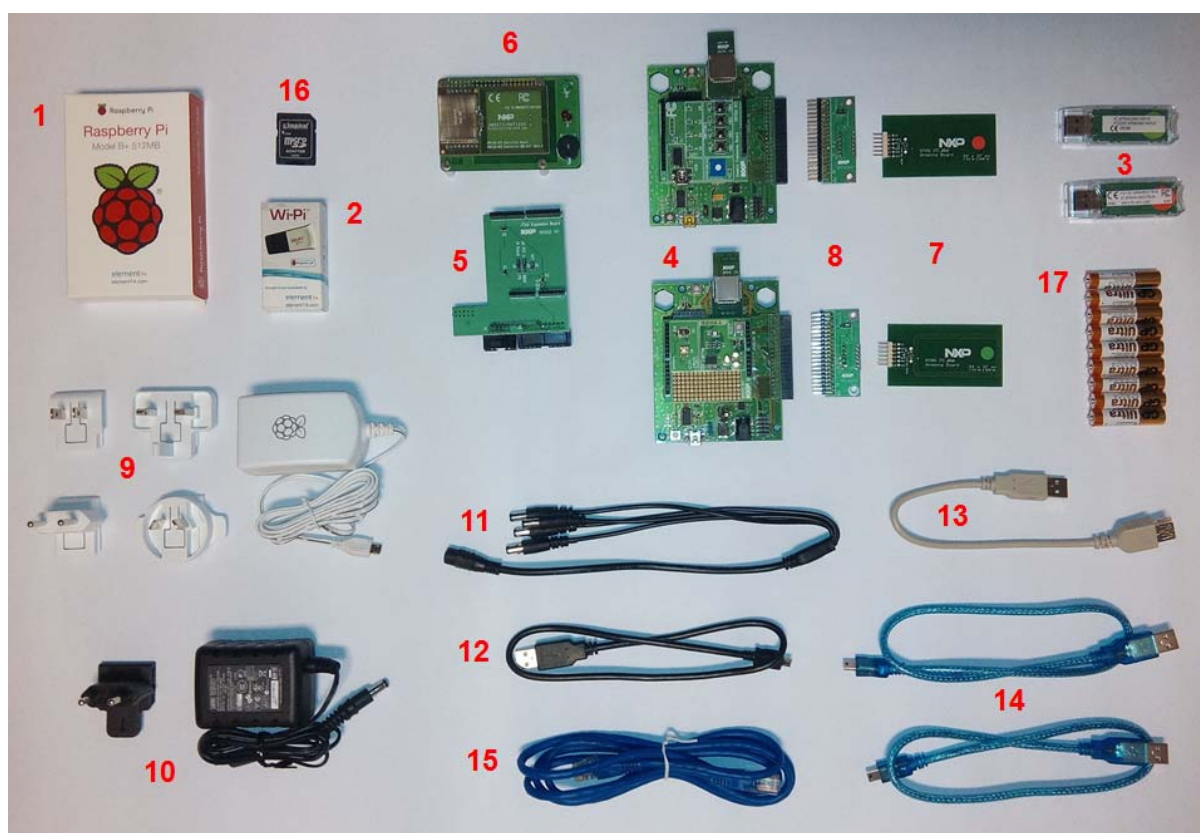


Figure 1: JN516x-EK004 Evaluation Kit

The full contents of the kit are listed below (numbers refer to [Figure 1](#) above).

1. Raspberry Pi single-board computer (Model B+) and microSD card ^(Note a)
2. Wi-Fi Raspberry Pi 802.11n wireless adaptor (for Wi-Fi connectivity)
3. 2 JN5169 USB Dongles with integrated antenna ^(Note b)
4. 2 Carrier Boards:
 - one pre-fitted with a Generic Expansion Board and a JN5169 module with integrated antenna ^(Note b)
 - one pre-fitted with a Lighting/Sensor Expansion Board and a JN5169 module with integrated antenna ^(Note b)
5. JTAG Expansion Board
6. PN7120 NFC Controller Board with Raspberry Pi Interface Board
7. 2 NTAG I²C plus Antenna Boards
8. 2 40-way connection adaptors for NTAG I²C plus Antenna Boards
9. 5V DC power supply unit (universal type) with USB port for Raspberry Pi
10. 12V DC power supply unit for Carrier Boards ^(Note c)
11. 5-way 2.1mm daisy-chain power extender-cable (for use with Carrier Boards)
12. 'USB to Micro USB' cable (for powering Raspberry Pi from 5V PSU)
13. 'USB to USB' extension cable (for use with a JN5169 USB Dongle)
14. 2 'USB A to Mini B' cables
15. RJ45 Ethernet cable
16. microSD card adaptor
17. 9 AAA batteries

Notes:

- a) The Raspberry Pi board's microSD card is programmed with NXP firmware (an adaptor is provided to allow re-programming of the microSD card from a PC).
- b) The pre-fitted JN5169 modules and the JN5169 USB Dongle with a green sticker are pre-programmed with the relevant binaries for the ZigBee Smart Home Demonstration. The JN5169 USB Dongle with a red sticker is pre-programmed as a packet sniffer.
- c) The NTAG I²C plus Antenna Board with a green sticker should be used with the Lighting/Sensor Expansion Board and the one with the red sticker should be used with the Generic Expansion Board.
- d) The 12V DC PSU accepts universal input voltages (100-240V, AC 50/60Hz) but is fitted with a European plug. It can be used outside Europe by plugging it into the universal AC outlet socket of the 5V DC PSU.

Additional Nodes (Available Separately)

- **JN5169XK010 ZigBee Generic Node Expansion Kit:** Contains a Carrier Board with a Generic Expansion Board and JN5169 module
- **JN5169XK020 ZigBee Lighting/Sensor Node Expansion Kit:** Contains a Carrier Board with a Lighting/Sensor Expansion Board and JN5169 module
- **JN5169 USB Dongle (OM15020):** Pre-programmed as a packet sniffer

1.2 Wireless Network Fundamentals

This section covers the basic concepts that are relevant to the wireless network protocols supported by the JN516x-EK004 evaluation kit, notably IEEE 802.15.4 and ZigBee PRO (which is itself built on IEEE 802.15.4). For a more complete introduction to wireless networks, refer to the manual for the relevant protocol (see [Section 3.3](#)).

The topics covered are:

- Radio frequency - see [Section 1.2.1](#)
- Network identification - see [Section 1.2.2](#)
- Node types - see [Section 1.2.3](#)
- Network formation and topology - see [Section 1.2.4](#)
- Wireless network protocol stacks - [Section 1.2.5](#)

1.2.1 Radio Frequency

The JN5169-based devices of the evaluation kit communicate in the 2400-MHz radio frequency (RF) band. This is one of the standard RF bands used by the IEEE 802.15.4 wireless network protocol (on which all the supported protocols are based), and is available for unlicensed use in most geographical areas (check your local radio communication regulations).

The 2400-MHz band spans radio frequencies 2400 to 2483.5 MHz and is divided into 16 channels, numbered 11 to 26 (the numbers up to 10 are used for other RF bands) - see [Figure 2](#) below.

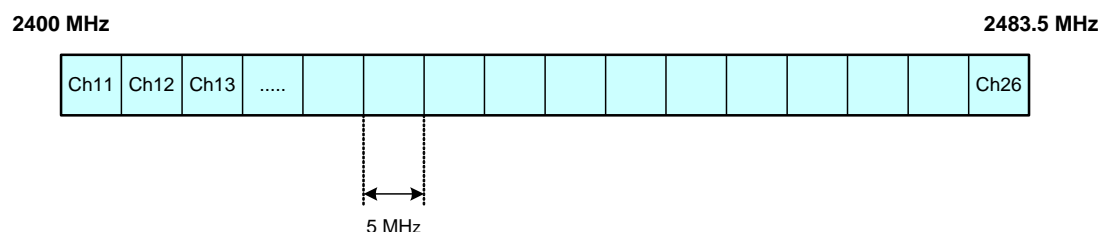


Figure 2: 2400-MHz RF Band

The channel of operation for a network can be specified by the application or, alternatively, the network can search for the best channel (the one with least detected activity) at system start-up.

1.2.2 Network Identification

A wireless network must have a unique identifier which allows it to be distinguished from other networks in the same operating neighbourhood. IEEE 802.15.4-based protocols use a 16-bit value called the PAN ID. This identifier can be pre-defined in the user application or chosen by the PAN Co-ordinator which creates the network (see [Section 1.2.3](#)), but it must not clash with the PAN ID of a neighbouring network.

A wireless network protocol may also use its own network identifier (in addition to the PAN ID) - for example, in the ZigBee PRO protocol, a 64-bit Extended PAN ID (EPID) is also used.

1.2.3 Node Types

In addition to running an application (e.g. temperature measurement), each node of a wireless network has a networking role. Most IEEE 802.15.4-based networks can contain three types of node differentiated by their networking roles, as follows:

Node Type	Description
Co-ordinator	A wireless network has one and only one Co-ordinator, or PAN Co-ordinator. This node has a role in starting and forming the network, and can also have a routing role (passes messages from one node to another). Messages can also originate and terminate here.
Router	This node passes messages from one node to another. Messages can also originate and/or terminate at a Router. This node type is also sometimes referred to as a Local Co-ordinator.
End Device	This node is simply a place where messages can originate and/or terminate (the node does not have a routing role). In order to conserve power, an End Device can sleep when it is not required to be active.

Table 1: Node Types in a Wireless Network

1.2.4 Network Formation and Topology

A wireless network is formed as follows:

1. The first node to be started is the Co-ordinator, which performs network initialisation including the selection of the radio channel in which the network will operate (see [Section 1.2.1](#)) and the PAN ID of the network (see [Section 1.2.2](#)).
2. Once the initialisation is complete, the Co-ordinator allows other nodes to join it as its 'children' - the Co-ordinator is referred to as a 'parent'. These child nodes can be Routers and/or End Devices.
3. Each Router may itself become a parent by allowing child nodes to join it (which may themselves be Routers and/or End Devices). In this way, the network grows.

A number of network shapes or topologies are possible, which are dependent on the wireless network protocol used. These are the Star, Tree and Mesh topologies, described in the sub-sections below.

1.2.4.1 Star Networks

A Star network contains only one routing node, which is the Co-ordinator. Thus, all messages between the network nodes are routed via the central Co-ordinator.

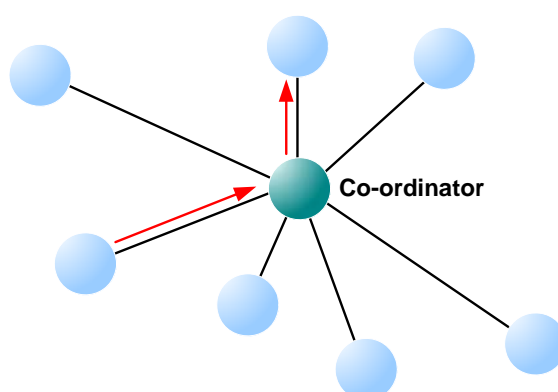


Figure 3: Star Network

1.2.4.2 Tree Networks

A Tree network may contain the full range of node types - Co-ordinator, Routers and End Devices - with the Co-ordinator at the top of the (inverted) tree. Any node can only communicate directly with its parent and children (if any). Thus, a message for a remote node must be passed up the tree until it can be passed down to its destination.

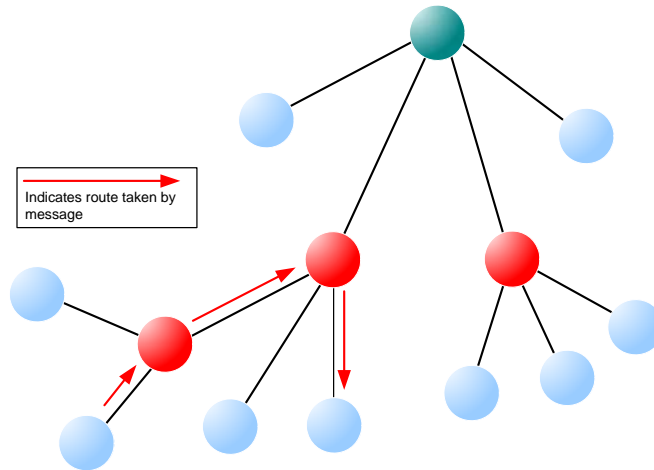


Figure 4: Tree Network

1.2.4.3 Mesh Networks

A Mesh network may contain the full range of node types - Co-ordinator, Routers and End Devices. Any routing node (Router or Co-ordinator) can communicate directly with any other routing node within radio range. However, an End Device can only communicate directly with its parent. When sending a message between two nodes that cannot communicate directly, the message is routed via intermediate routing nodes (for which there may be more than one possible route).

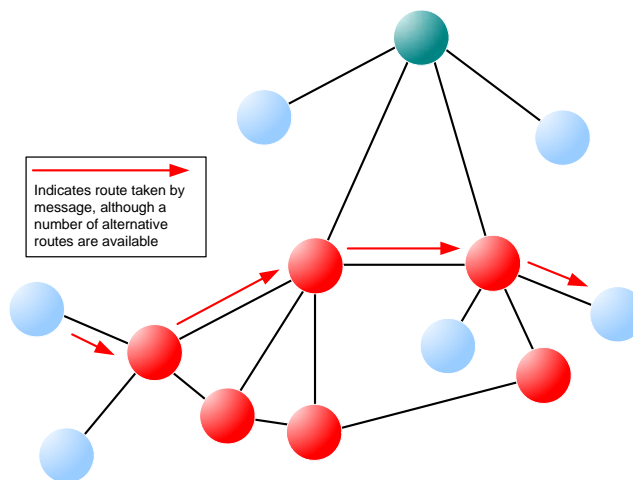


Figure 5: Mesh Topology

1.2.5 Wireless Network Protocol Stack

The software which runs on a wireless network node deals with both application-specific tasks (e.g. temperature measurement) and networking tasks (e.g. assembling a message to send to another node). The software on a node is organised as a number of layers, forming a stack, with the user application at the top. The basic layers of a wireless network protocol stack are typically as illustrated in [Figure 6](#) below.

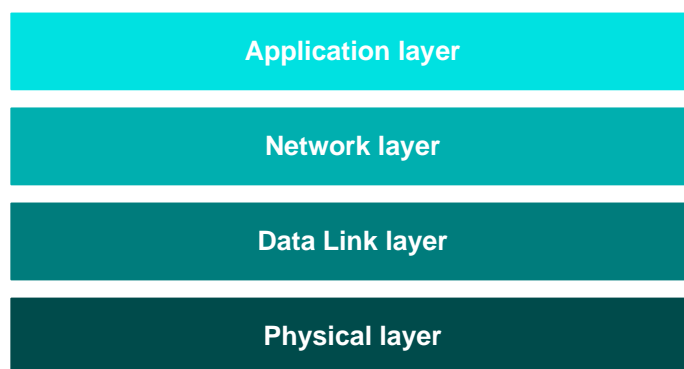


Figure 6: Basic Layers of Stack

The stack layers are outlined below, from top to bottom:

- **Application layer:** Contains the user applications which run on the network node and provides an interface to the lower layers
- **Network layer:** Provides the networking functionality and an interface to the IEEE 802.15.4 layers (see below) - for example, this layer could be provided by the ZigBee PRO protocol, but is not required for a pure IEEE 802.15.4 network
- **Data Link layer:** Provided by the MAC (Media Access Control) layer of the IEEE 802.15.4 standard, and responsible for message assembly/disassembly and addressing (e.g. specifying the destination of outgoing messages)
- **Physical layer:** Provided by the PHY (Physical) layer of the IEEE 802.15.4 standard and concerned with interfacing to the physical transmission medium (radio, in this case)

A number of C APIs (Application Programming Interfaces) are provided to allow a user application to easily interact with the layers of the stack. The required APIs depend on the protocol used (e.g. IEEE 802.15.4, ZigBee PRO) and are supplied in the relevant NXP Software Developer's Kit (SDK) - see [Section 3.1](#).

The software components that implement the Data Link and Physical layers of the stack are pre-programmed in on-chip ROM. The user application is developed and built on a PC. The resulting application binary file also incorporates the Network layer of the stack and is loaded into the Flash memory of the node.

In order to use the ZigBee Smart Home Demonstration that is pre-loaded in the JN516x-EK004 evaluation kit boards, no knowledge of the protocol software is required. However, you should familiarise yourself with the software for your chosen protocol before starting your own application development - refer to [Section 3.3](#) for details of the user documentation for the different protocols.

1.3 Internet of Things (IoT)

The 'Internet of Things' (IoT) refers to the concept of electrical devices being connected via the Internet, irrespective of their basic functionality. This allows the devices to be remotely controlled and monitored from other devices on the Internet - for example, from a PC, tablet or smart phone.

The IoT idea includes the possibility of connecting a WPAN, such as a ZigBee network, to the Internet. In this case, the WPAN may have a single connection to the Internet shared by all the nodes of the network, but the nodes will be individually accessible from the Internet. The Internet connection for a WPAN is provided by a device called an IoT Gateway. A basic system is shown in [Figure 7](#) below.

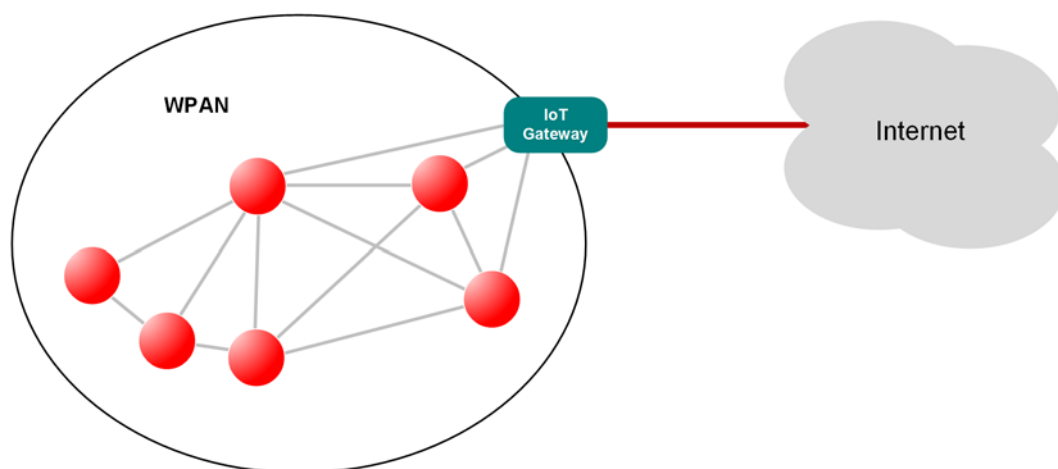


Figure 7: WPAN Connected to IoT

1.3.1 IoT Gateway

The IoT Gateway allows commands and data to be passed between the Internet and WPAN domains, in both directions. The device runs a protocol stack for each of its two sides, which are as follows:

- **IP Host:** This side runs an IP stack to send/receive data packets to/from the Internet - this is normally an IPv6 stack.
- **Control Bridge:** This side runs a wireless network protocol stack to send/receive data packets to/from the WPAN - this may be an IEEE 802.15.4-based stack, such as the ZigBee PRO stack. This part of the Gateway usually acts as a WPAN node, such as a Co-ordinator or Router.

The two stacks are connected within the IoT Gateway. The basic architecture of an IoT Gateway is illustrated in [Figure 8](#) below.

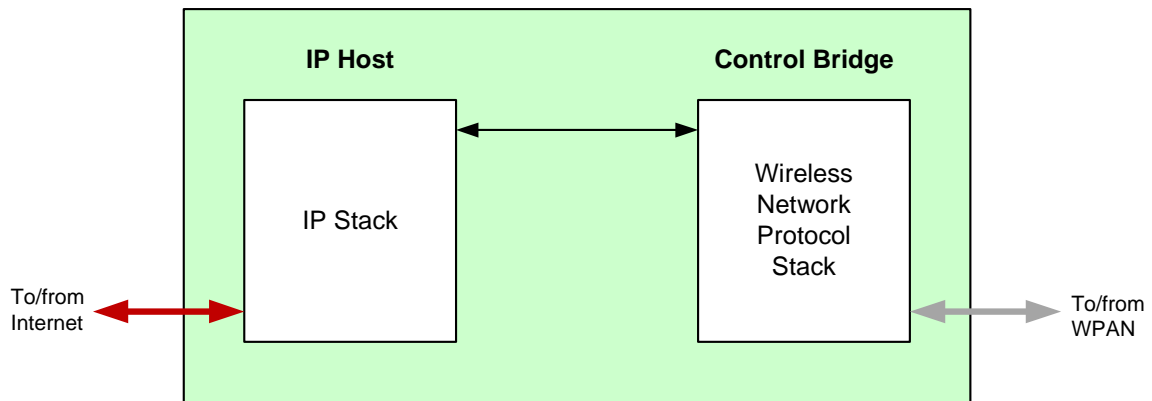


Figure 8: Basic Architecture of an IoT Gateway

1.3.2 IP Connectivity of the Kit

In the JN516x-EK004 evaluation kit, an IoT Gateway can be formed from the Raspberry Pi board together with a JN5169 USB Dongle:

- The Raspberry Pi provides the IP Host, interfacing to the IP domain
- The JN5169 USB Dongle provides the Control Bridge, interfacing to the WPAN, and also acts as the WPAN Co-ordinator

This IoT Gateway implementation is illustrated in Figure 9 below.

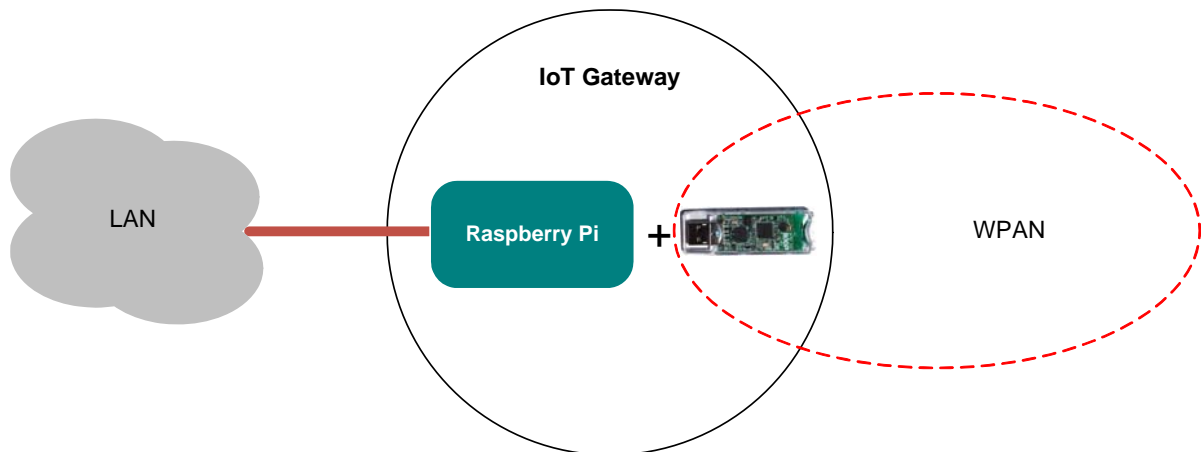


Figure 9: IoT Gateway Implementation

The Raspberry Pi microSD card is pre-programmed with NXP firmware which allows the IP Host to operate in conjunction with a ZigBee Control Bridge. The JN5169 USB Dongle with a green sticker is pre-programmed with firmware for a ZigBee Control Bridge and Co-ordinator.

The JN5169 USB Dongle can be plugged into any USB port of the Raspberry Pi. In the case where the IP connection will be provided via Wi-Fi, the supplied Wi-Pi adaptor must be plugged into another USB port of the Raspberry Pi (the Wi-Pi adaptor creates its own Wi-Fi network). If both the dongle and adaptor are required, the dongle must be connected to a port via a USB cable to ensure isolation of the two devices.

The IP connection can be to an existing wired LAN or via a new Wi-Fi LAN:

- To connect to an existing wired LAN, the supplied Ethernet cable must be used to connect the Raspberry Pi to a router in the LAN. The LAN must include a DHCP server to allocate an IP address to the Raspberry Pi. The DHCP leases in the router will need to be examined in order to determine the IP address allocated to the Raspberry Pi.
- To use a Wi-Fi connection, the Wi-Pi adaptor must be plugged into the Raspberry Pi. The Wi-Pi adaptor will create its own Wi-Fi network (rather than join an existing Wi-Fi network). A PC can then interact with the Raspberry Pi by joining this Wi-Fi network (which has the SSID `lot_GW_NXP`). In this case, the Raspberry Pi is allocated the fixed IP address `192.168.2.1`.

The pre-programmed ZigBee Smart Home Demonstration uses a Wi-Fi connection.

1.4 Network Commissioning

A WPAN is started by the Co-ordinator and other nodes can then join the network, as indicated in [Section 1.2.4](#). Network commissioning may also include the following:

- Network joining and subsequent network operation may employ security keys to implement encrypted communications between nodes, to make the network secure from malicious tampering. NFC commissioning is becoming an important technique in establishing this security and is supported by the JN516x-EK004 evaluation kit. NFC commissioning is outlined in [Section 1.4.1](#).
- Once a node has joined a network, it may need to be linked to one or more other nodes from a functional point-of-view. For example, when a new light is added to a lighting network, it may be required that a particular light switch (already in the network) is used to control the new light. There are different ways of establishing this link and these are described in [Section 1.4.2](#).

1.4.1 NFC Commissioning Process

The commissioning of network nodes can be performed via Near-Field Communications (NFC). This involves the exchange of a unique security key, through the close proximity required by NFC, to ensure that all communications to/from the joining node are encrypted during the commissioning process. Node commissioning is therefore secure and cannot be conducted remotely by third parties to introduce nodes for malicious purposes.

In its basic form, the NFC commissioning of a node to a WPAN requires the following security keys:

- **Unique Link Key:** An NFC tag on the node must be pre-programmed with a link key that is unique to the node and that will be used in the commissioning process.
- **Network Key:** The network Co-ordinator contains a network key that is used to encrypt all radio communications between the nodes during normal operation of the network.

As part of the commissioning process, the network key must be communicated from the Co-ordinator to the joining node in a secure manner, so that the node can later use the key to encrypt and decrypt network-level communications. The commissioning process works as follows:

1. The NFC tag of the joining node is brought close to the NFC Controller (mounted on the Raspberry Pi board, which is itself connected to the JN5169 USB Dongle that acts as the network Co-ordinator). This action is known as an 'NFC tap'.
2. During this NFC tap, the NFC Controller reads the unique link key from the NFC tag of the joining node and this key is internally passed to the Co-ordinator.
3. During this same NFC tap, the Co-ordinator uses the unique link key to encrypt the network key and sends it by NFC to the joining node.
4. The joining node receives the encrypted network key and decrypts it using the unique link key.

The joining node now has the network key which allows it to subsequently participate in secure network-level communications. Therefore, the network credentials are never sent over the air, providing a safer method of commissioning.

The JN516x-EK004 evaluation kit is equipped with the necessary components for NFC commissioning:

- The supplied NTAG I²C plus Antenna Boards (see [Appendix A.9](#)) are 'NFC connected tags' which can be attached to the Carrier Boards via supplied adaptors. A Carrier Board is the basis of a WPAN node and the NFC tag is used to join the node to the network.
- The supplied NFC Controller (see [Appendix A.8](#)) attaches to the Raspberry Pi board. The NFC Controller is able to read data from the NFC tag attached to a Carrier Board (node) as part of the secure commissioning process.

For more information on NFC commissioning, refer to the *NFC Commissioning User Guide (JN-UG-3112)*.

1.4.2 Addressing, Binding and Grouping

The available methods for establishing a functional link between a controlling node or source node and a target node are as follows:

- **Direct Addressing:** The source node can simply send commands to the address of the target node. This may be the permanent 64-bit IEEE/MAC address of the node or a 16-bit network (or short) address allocated to the node when it joined the network.
- **Binding:** In a ZigBee PRO network, 'binding' can be used to set up this link. An application on the source node can be bound to an application on the target node such that commands issued by the source application are always sent to the target node without needing to specify an address. The source node can be bound to multiple target nodes.
- **Grouping:** In a ZigBee PRO network, 'grouping' can alternatively be used to set up this link, particularly when controlling multiple target nodes simultaneously (e.g. several lights in unison). The target nodes are collected into a group with an associated 16-bit group address. A target node 'knows' which groups it belongs to - this information is held in a table on the node. When the source node sends a command to a group address, the command is broadcast to all nodes in the network. A receiving node is able to identify whether it is a member of the relevant group and only the nodes in the group react to the command.

1.5 ZigBee Smart Home Demonstration

Certain components of the kit are pre-loaded with the software for a simple demonstration network. Running this ZigBee Smart Home Demonstration will allow you to use components of the kit without programming them yourself.

The demonstration is a lighting system which allows the white LEDs on a node/board (Dimmable Light) of a WPAN to be controlled (switched on/off, brightness adjusted) in the following ways:

- From within the WPAN using the buttons on a controller node/board (Dimmer Switch)
- From outside the WPAN via an IP connection from a PC

The demonstration system is depicted in its basic form below in [Figure 10](#).

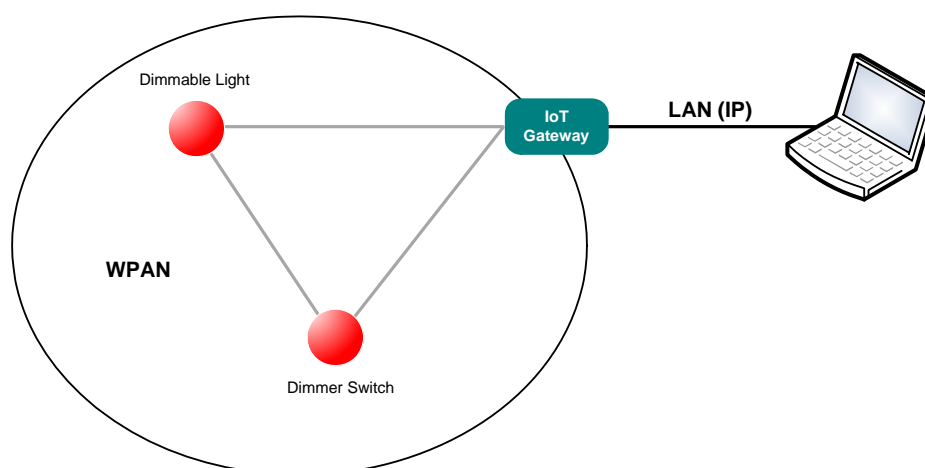


Figure 10: Demonstration System

More detailed information and instructions on how to run the demonstration are provided in [Chapter 2](#).



Note 1: A Quick Start procedure for running this demonstration is also provided on the *JN516x-EK004 Getting Started sheet (JN-UG-3107)*, which is supplied in the kit box.

Note 2: The demonstration application code is described in the Application Note *ZigBee HA Lighting with NFC (JN-AN-1221)*, which also contains the relevant source and binary files.

2. Using the Smart Home Demonstration

The chapter describes how to use the contents of the JN516x-EK004 evaluation kit to set up and run the pre-loaded ZigBee Smart Home Demonstration. The demonstration is concerned with the control of a light in a 'Smart Home' and has the following features:

- **ZigBee wireless network:** This network contains the light to be controlled, as well as a dimmer switch that can be used to wirelessly control the light.
- **'Internet of Things' (IoT) Gateway:** This device connects the ZigBee wireless network to the IP domain, allowing the light to be controlled from a web browser on a PC.
- **NFC commissioning:** The formation of the ZigBee wireless network is simplified by means of NFC commissioning, which allows a node to be joined to the network through the simple action of an NFC tap.



Note 1: A Quick Start procedure for running this demonstration is provided on the *JN516x-EK004 Getting Started sheet (JN-UG-3107)*, which is supplied in the kit box.

Note 2: This demonstration is based on the ZigBee PRO protocol with the Home Automation profile.

Note 3: For further information on this demonstration, refer to the Application Note *ZigBee HA Lighting with NFC (JN-AN-1221)*.

2.1 Demo System Overview

This ZigBee Smart Home Demonstration is based on a ZigBee WPAN containing a light node (Dimmable Light) that can be controlled as follows:

- From within the WPAN using the buttons on a controller node (Dimmer Switch)
- From a web browser on a PC located in a Wi-Fi LAN that is connected via an IoT Gateway to the WPAN

The demonstration system is illustrated in [Figure 11](#).

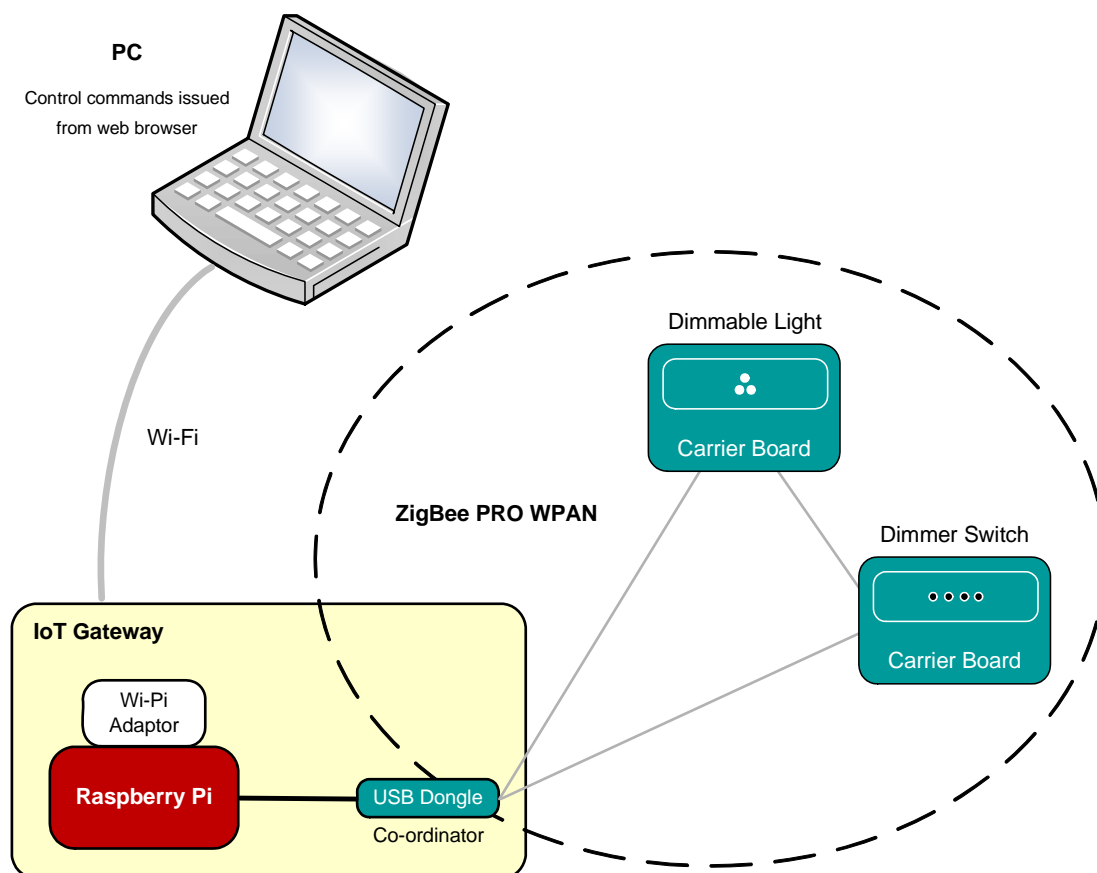


Figure 11: ZigBee Smart Home Demonstration System

The system is made up from components of the evaluation kit as follows:

- **Dimmable Light:**
 - Carrier Board with JN5169 module and NFC antenna
 - Lighting/Sensor Expansion Board, providing the LEDs
- **Dimmer Switch:**
 - Carrier Board with JN5169 module and NFC antenna
 - Generic Expansion Board, providing the control buttons
- **IoT Gateway:**
 - Raspberry Pi board
 - Wi-Pi adaptor, providing the Wi-Fi connection (and DHCP server)
 - JN5169 USB Dongle, acting as the ZigBee Co-ordinator and providing the connection to the WPAN
 - NFC Controller Board, providing NFC commissioning

For detailed descriptions of the above components, refer to [Appendix A](#).

The Dimmable Light and Dimmer Switch nodes both act as ZigBee Routers.

2.2 Setting Up the IoT Gateway

This section describes how to set up the IoT Gateway for the ZigBee Smart Home Demonstration. As detailed in [Section 2.1](#), the IoT Gateway comprises:

- Raspberry Pi board with microSD card
- Wi-Pi adaptor
- JN5169 USB Dongle (with a green sticker)
- NFC Controller



Note: The IoT Gateway is fully detailed in the Application Notes *ZigBee IoT Gateway Host with NFC (JN-AN-1222)* and *ZigBee IoT Gateway Control Bridge (JN-AN-1223)*.

Step 1 Ensure that the PC is up and running

If not already done, power-up the PC and wait for the boot process to complete.

Step 2 Connect the Wi-Pi adaptor to the Raspberry Pi board

Plug the supplied Wi-Pi adaptor into any USB socket of the Raspberry Pi board, but do not power-up the Raspberry Pi yet.

Step 3 Connect the JN5169 USB Dongle to the Raspberry Pi board

Connect the JN5169 USB Dongle with a green sticker (this dongle is programmed as a ZigBee Control Bridge and Co-ordinator) to any USB socket of the Raspberry Pi board via the supplied USB-to-USB cable (use of this cable ensures that the dongle is isolated from the Wi-Pi adaptor).

Also check that the pre-installed microSD card is securely fitted to the Raspberry Pi.

Step 4 Mount the NFC Controller onto the Raspberry Pi board

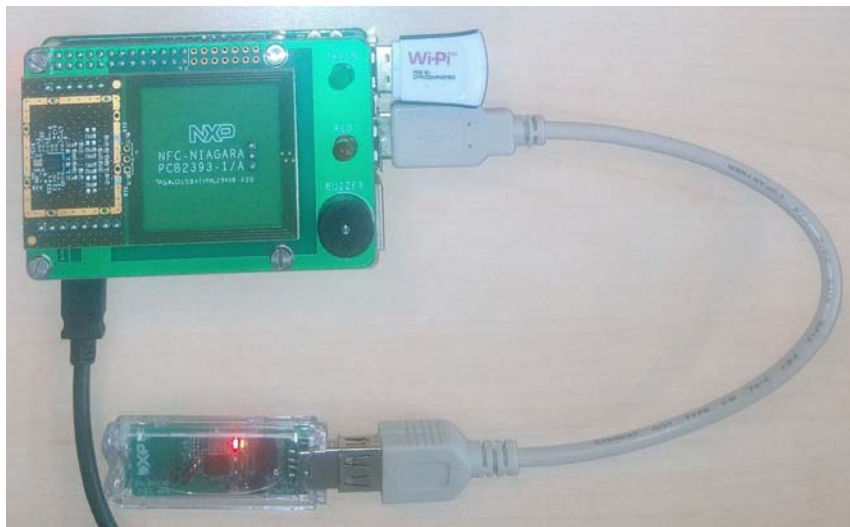
Install the NXP PN7120 NFC Controller assembly (NFC Controller Board pre-mounted on a Raspberry Pi Interface Board) on the Raspberry Pi board. For information on the NFC Controller and related documentation, refer to [Appendix A.8](#). If in doubt about this installation, refer to the *PN7120 NFC Controller SBC Kit Quick Start Guide (AN11646)* - note that the supplied Raspberry Pi board is a B+ model.

Step 5 Connect the Raspberry Pi board to a power supply

Power-up the Raspberry Pi board (and therefore IoT Gateway) by connecting it to a power supply. To do this, connect the supplied 5V DC PSU to the mains supply and then use the 'USB to Micro USB' cable to connect the Raspberry Pi board to a USB port on the PSU.

Chapter 2 Using the Smart Home Demonstration

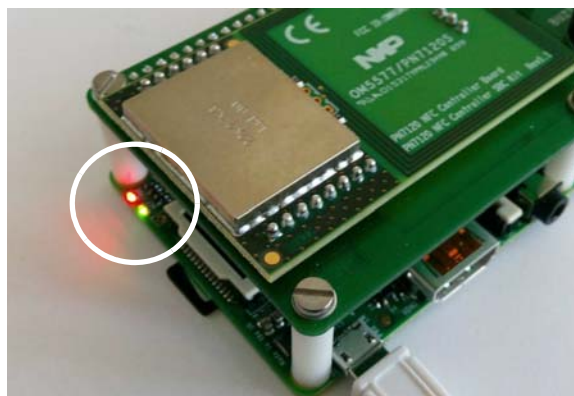
The equipment should now be as shown below.



Step 6 Wait for the IoT Gateway to start up

Wait about 20 seconds for the IoT Gateway to boot. During the boot process, the following will occur:

- a) The red LED on the Raspberry Pi board will illuminate.
- b) The green LED on the Raspberry Pi board will flash.
- c) An audible beep will be sounded by the Raspberry Pi board.



The IoT Gateway is then ready. When in this state, the green LED on the NFC Controller board flashes.

The Wi-Fi adaptor will have created a Wi-Fi network (initially consisting of only itself).

Step 7 Direct a web browser on the PC to the IoT Gateway

- a) Connect the PC to the ZigBee network via the IoT Gateway by specifying the SSID (name) of the network, which is **lot_GW_NXP**.
- b) Direct a web browser on the PC to the IoT Gateway's web interface using either the default IP address **192.168.2.1** or the URL **http://iot-gw.nxp**.

The first page of the interface displayed in the browser is the “IoT Rooms” page (shown below). This shows the different rooms of the Smart Home (Bathroom, Kitchen, Bedroom and Living Room) in which devices may be controlled.

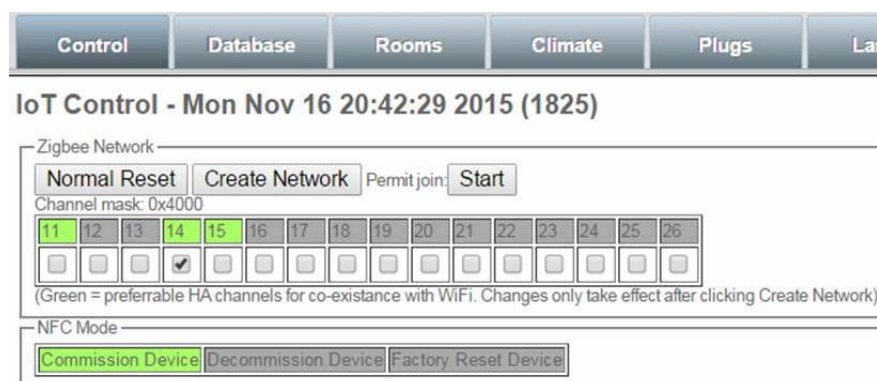


Step 8 Select a ZigBee network channel and create a network

This step is needed only the first time the demonstration is used.

In the IoT Gateway interface, go to the Control page by clicking the **Control** button (left-most button across the top). On the Control page, do the following in the ZigBee Network area:

- a) Set the radio channel to be used in the ZigBee network to channel 14 by clicking (ticking) the box labelled “14”.
- b) Create the ZigBee network (initially containing only the Co-ordinator) by clicking the **Create Network** button.



The ZigBee network will be created on channel 14. A successful network start will be indicated by the illumination of the green and orange LEDs on the JN5169 USB Dongle.

You can now assemble the network nodes as described in [Section 2.3](#) and add them to the network using NFC commissioning as described in [Section 2.4](#).

2.3 Assembling the Nodes

This section describes how to assemble the Dimmer Switch and Dimmable Light nodes to be used in the demonstration. Each node comprises the following components:

- DR1174 Carrier Board fitted with JN5169 module
- Expansion Board - this depends on the node, as follows:
 - Dimmer Switch - DR1199 Generic Expansion Board
 - Dimmable Light - DR1175 Lighting/Sensor Expansion Board
- NTAG I²C plus (NFC) Antenna Board
- OM15044 adaptor for NFC antenna board

The antenna boards carry red and green stickers to indicate which expansion boards they should be used with - red for Generic, green for Lighting/Sensor.

Once assembled as described in this section, a node will look as shown in [Figure 12](#).



Figure 12: Assembled Node



Note: In this demonstration, it is assumed that the nodes will be powered by batteries. However, you can alternatively power the nodes from an external device via the USB Mini B connector or from the 12V DC PSU provided in the evaluation kit (see [Appendix A.1.1](#)). It is possible to power more than one node from the PSU by daisy-chaining the nodes using the supplied cable.

The assembly procedure is provided below - this applies to both nodes.

Step 1 Ensure that the power selector is set for batteries (if required)

The Carrier Boards are factory-configured to use batteries as their power source. However, if this is not the first time the kit has been used and you intend to use the batteries, check that batteries are selected as the power source for the Carrier Board, with the jumper **J4** in the **BAT** position (as described in [Appendix A.1.1](#)). If an expansion board is already fitted to the Carrier Board, it will be necessary to remove the expansion board in order to check this jumper.

Step 2 Fit an expansion board to the Carrier Board (if necessary)

The Carrier Boards are factory-fitted with expansion boards. However, if there is no expansion board fitted then connect one to the Carrier Board via the on-board connectors CN1-CN5 (see [Figure 15](#) on page 49):

- DR1199 Generic Expansion Board for the Dimmer Switch node
- DR1175 Lighting/Sensor Expansion Board for the Dimmable Light node

Step 3 Connect the NFC antenna board to its adaptor

Connect the NTAG I²C plus Antenna Board to the OM15044 adaptor via the 6-way connector.

Step 4 Connect the NFC antenna/adaptor to the Carrier Board and configure them

Connect the NFC antenna/adaptor combination to the Carrier Board via the 40-way connector, using the appropriate NFC antenna board for the expansion board fitted to the Carrier Board:

- Generic Expansion Board: Use NFC antenna board with **red** sticker
- Lighting/Sensor Expansion Board: Use NFC antenna board with **green** sticker

Step 5 Configure the adaptor board for use with the relevant expansion board

Position the jumper **J16** on the OM15044 adaptor according to the expansion board:

- Generic Expansion Board: **J16** in position **DIO/Switch**
- Lighting/Sensor Expansion Board: **J16** in position **DIO17/Lighting**

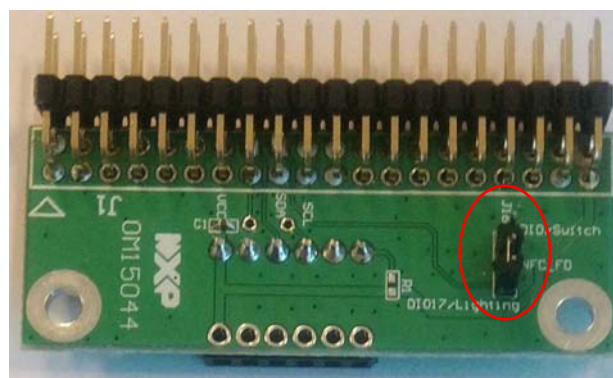


Figure 13: J16 Jumper on OM15044 Adaptor

The node assembly is now complete but do not apply power to the nodes yet. You can add the nodes to the network using NFC commissioning, as described in [Section 2.4](#).

2.4 Commissioning the Nodes

This section describes how to complete the set-up of the ZigBee Smart Home Demonstration by adding nodes to the ZigBee network by means of NFC commissioning.



Note: This section assumes that you have set up the IoT Gateway as detailed in [Section 2.2](#).

You will add two nodes to the network already established by the Co-ordinator (JN5169 USB Dongle):

- Dimmable Light
- Dimmer Switch

These nodes should have already been assembled from components of the kit, as instructed in [Section 2.3](#).

The NFC commissioning of each node into the ZigBee network is identical. It is performed by simply holding the NFC antenna of the node above the NFC antenna of the NFC Controller (mounted on the Raspberry Pi board) while the Raspberry Pi is in commissioning mode - this action is known as an 'NFC tap' and is shown in [Figure 14](#). An audible signal is sounded if the NFC communication between the two devices was successful, after which the node will join the ZigBee network.

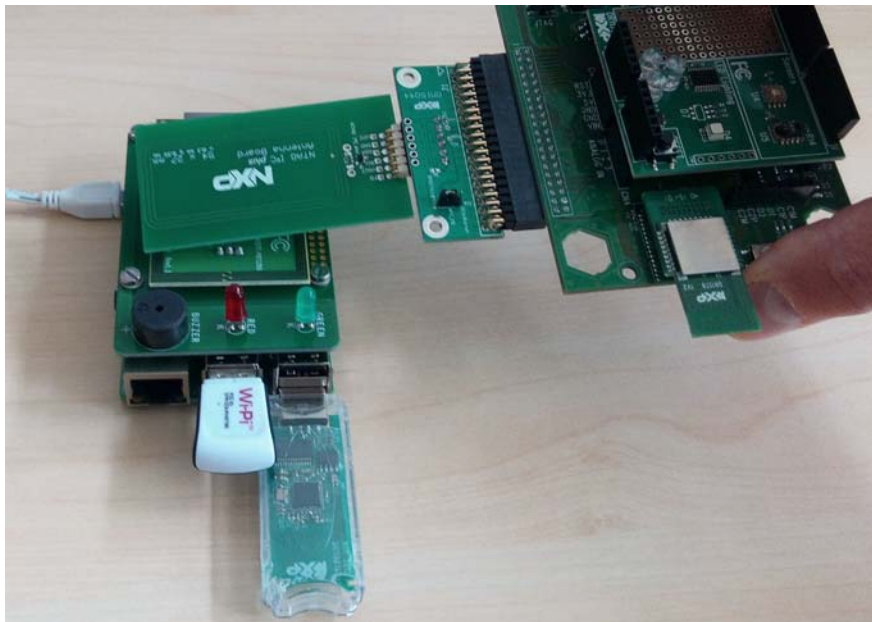


Figure 14: NFC Tap

The commissioning procedure is described below.



Note: If a node has previously been used, it will retain settings (e.g. PAN ID) from the previous network to which it belonged. To clear this information and return to the factory settings, perform a factory reset as follows: *hold down the button 'DIO8' and then press the switch 'RST', both on the underlying Carrier Board.*

Step 1 Ensure that the WPAN is open for joining

Before continuing with this procedure, make sure that the Co-ordinator has created a WPAN that is open for joining, indicated by illuminated green and orange LEDs on the JN5169 USB Dongle.

If the above LEDs are not illuminated, perform a reset by clicking the **Normal Reset** button on the Control page of the IoT Gateway interface in the web browser.

Step 2 Ensure that the Raspberry Pi is in commissioning mode

On the Control page of the IoT Gateway interface in the web browser, ensure that the NFC mode is set to **Commission Device** (which should be highlighted in green).

Step 3 Power-up the Dimmer Switch node

Assuming the batteries have been selected as the power source for the Carrier Board of the Dimmer Switch node (see [Section 2.3](#)), insert four of the supplied AAA batteries onto the rear of the Carrier Board (the required polarities are indicated on the board).



Note: You can alternatively power the board from an external device via the USB Mini B connector or from the 12V DC PSU provided in the evaluation kit (see [Appendix A.1.1](#)).

Step 4 Commission the Dimmer Switch node into the network

Move the NFC antenna of the Dimmer Switch node over the NFC antenna of the NFC Controller Board, as shown in [Figure 14](#). If data is successfully exchanged in this NFC tap, the Raspberry Pi board will sound a long beep.

An unsuccessful NFC tap will be indicated by four short beeps. In this case, click the **Decommission Device** button on the Control page of the IoT Gateway interface and repeat the NFC tap (successful decommissioning is indicated by two beeps), then click the **Commission Device** button and repeat this step.

While the Dimmer Switch node is not in a ZigBee network, the LEDs D1 and D2 flash alternately. Once the node has joined the network, both LEDs D1 and D2 switch off.

During NFC commissioning, all network-related information and a join command are transferred from the IoT Gateway to the joining node. Therefore, after NFC commissioning, the node automatically joins the ZigBee network.

Chapter 2 Using the Smart Home Demonstration

Step 5 Power-up the Dimmable Light node

Assuming the batteries have been selected as the power source for the Carrier Board of the Dimmable Light node (see [Section 2.3](#)), insert four of the supplied AAA batteries onto the rear of the Carrier Board (the required polarities are indicated on the board).



Note: You can alternatively power the board from an external device via the USB Mini B connector or from the 12V DC PSU provided in the evaluation kit (see [Appendix A.1.1](#)).

Step 6 Commission the Dimmable Light node into the network

Move the NFC antenna of the Dimmable Light node over the NFC antenna of the NFC Controller Board, as shown in [Figure 14](#). If data is successfully exchanged in this NFC tap, the Raspberry Pi board will sound a long beep.

An unsuccessful NFC tap will be indicated by four short beeps. In this case, click the **Decommission Device** button on the Control page of the IoT Gateway interface and repeat the NFC tap (successful decommissioning is indicated by two beeps), then click the **Commission Device** button and repeat this step.

The Dimmable Light node will now automatically join the network and the three white LEDs on the node's Lighting/Sensor Expansion Board will flash once.

Step 7 Check that the Dimmable Light has appeared in the web browser on the PC

In the web browser on the PC, go to the "Mobile Site" page by clicking the **Mobile Site** button (right-most button across the top) and check that the Dimmable Light appears on the page - the light will be identified by its IEEE/MAC address.



Step 8 Bind the Dimmer Switch to the Dimmable Light

It is now possible to bind the Dimmer Switch to the Dimmable Light (so that they will operate together).

To bind the Dimmer Switch to the Dimmable Light:

- a)** Press and hold down the Commissioning button (DIO8) on the Carrier Board of the Dimmer Switch node - the white LEDs on the Lighting/Sensor Expansion Board of the Dimmable Light node will start to flash.
- b)** Release the button DIO8 when the white LEDs stop flashing and stay illuminated.
- c)** Press the button SW1 on the Generic Expansion Board of the Dimmer Switch node.

The Dimmer Switch and Dimmable Light nodes are now bound. The Dimmer Switch is in Control mode and can be used to control the Dimmable Light, as described in [Section 2.5](#).



Note: If you wish to unbind the two nodes, you can do this from the Carrier Board of the Dimmer Switch node, as follows: press and hold down the RST button, then press and hold down the DIO8 button, then release the RST button and finally release the DIO8 button.

2.5 Controlling the Light

Once you have set up the ZigBee Smart Home Demonstration, as described in [Section 2.2](#) and [Section 2.4](#), you can control the Dimmable Light from inside and/or outside the ZigBee WPAN:

- Control from the Dimmer Switch within the WPAN is described in [Section 2.5.1](#)
- Control from a PC outside the WPAN is described in [Section 2.5.2](#)



Caution: *The LEDs on the Lighting/Sensor Expansion Boards are very bright at maximum intensity. To avoid damage to your eyes, do not look into them directly for an extended period of time.*

2.5.1 Controlling the Dimmable Light from the Dimmer Switch

The Dimmer Switch node can be used to control the Dimmable Light node:

- The controlled lights are the three white LEDs (D1-D3) on the Lighting/Sensor Expansion Board.
- The four buttons SW1-SW4 on the Generic Expansion Board are used to control the lights, as follows:

Button	Action
SW1	Switch on lights
SW2	Switch off lights
SW3	Increase brightness of lights
SW4	Decrease brightness of lights

Table 2: Button Usage on Dimmer Switch

2.5.2 Controlling the Dimmable Light from the PC

The Dimmable Light within the ZigBee network can be controlled from the web browser on the PC, as follows:

Step 1 Access the Dimmable Light in the web browser on the PC

Ensure that the “Mobile Site” page is displayed in the web browser on the PC and click on the lamp icon that represents the Dimmable Light. The following controls for the light will now be displayed:



Step 2 Control the light from the PC

The Dimmable Light can be controlled from the PC using the controls shown above, as follows:

- Toggle the light on and off by clicking on button.
- Vary the brightness of the light by either of the following methods:
 - Enter a brightness level as a decimal fraction in the range 0 to 1.0 into the numerical field.
 - Click on and move the widget on the greyscale background to the desired brightness level.

Chapter 2
Using the Smart Home Demonstration

3. Where Next?

Once you have set up and run the pre-loaded ZigBee Smart Home Demonstration, you may wish to start developing your own wireless network applications to run on the components of the JN516x-EK004 evaluation kit. This chapter helps you to get started in this application development, as follows:

- [Section 3.1](#) provides information on the wireless network protocols supported for JN516x application development on the evaluation kit components.
- [Section 3.2](#) outlines the NXP Software Developer's Kits (SDKs) that are available to support application development using the different protocols.
- [Section 3.3](#) directs you to the user documentation for each protocol.

3.1 Which Protocol?

The JN516x-EK004 evaluation kit supports a number of wireless network protocols that can be used to develop applications for the JN516x wireless microcontroller.

The supported protocols for this evaluation kit are as follows:

- **IEEE 802.15.4:** This is an industry-standard protocol which provides the low-level functionality for implementing wireless network communications. The other supported protocols are built on top of IEEE 802.15.4, but an application can also be designed to interface directly with the IEEE 802.15.4 stack layers and an API is provided to facilitate this interaction. Developing an application directly on top of IEEE 802.15.4 requires lower level programming than using one of the other supported protocols.
- **ZigBee PRO:** This is an industry-standard protocol which provides a Network stack layer (over IEEE 802.15.4) that supports Mesh networking. APIs are provided to facilitate the interaction between the application and the ZigBee PRO stack. These APIs must be used in conjunction with the APIs of JenOS (Jennic Operating System).

The following ZigBee 'application profiles' are supported for use with ZigBee PRO on the evaluation kit components:

- **ZigBee Light Link (ZLL):** This profile is concerned with the control of the lighting in a building.
- **Home Automation (HA):** This profile is concerned with the control of a range of technology in the home, including lighting and security.

Profile-specific APIs are provided for both of the above profiles to aid application coding.

- **ZigBee 3.0:** This is the latest edition of the ZigBee protocol which uses the ZigBee PRO stack, described above, and promotes inter-operability between networks from different market sectors. The NXP ZigBee 3.0 software supports the ZigBee Lighting & Occupancy (ZLO) devices, where ZLO replaces the ZLL and HA profiles.

A full range of support resources, including example applications, are available for using the above protocols with the JN516x-EK004 evaluation kit (see [Section 3.3](#)).

3.2 Software Developer's Kit (SDK)

NXP provide Software Developer's Kits (SDKs) to facilitate the development of JN516x wireless network applications on a PC. These SDKs include Application Programming Interfaces (APIs) and have associated development tools. You will need two installers - a toolchain installer and the SDK installer for your wireless network protocol.

The required toolchain installer has the part number JN-SW-4141. This is based on the 'BeyondStudio for NXP' Integrated Development Environment (IDE) and is for use with SDK installers with part numbers of the form JN-SW-41xx.

The SDK installers for the protocols/profiles supported by the kit are as follows:

- JN516x IEEE 802.15.4 SDK (JN-SW-4163)
- JN516x ZigBee Light Link and Home Automation SDK (JN-SW-4168)
- JN516x ZigBee 3.0 SDK (JN-SW-4170)



Note: The software required to develop applications directly on the IEEE 802.15.4 stack layers is also included in the ZigBee SDK installers.

The above installers can be requested free-of-charge via the NXP web site (see ["Support Resources" on page 7](#)). Toolchain and SDK installation instructions are provided in the *Beyond Studio for NXP Installation and User Guide (JN-UG-3098)*. The contents of the installers are outlined in the sub-sections below.

3.2.1 'BeyondStudio for NXP' Toolchain (JN-SW-4141)

The 'BeyondStudio for NXP' toolchain is tailored for application development for the JN516x devices and must be installed first, before the SDK. It is based on the 'BeyondStudio for NXP' IDE, which is Eclipse-based. The installer includes:

- 'Beyond Studio for NXP' IDE
- Integrated JN51xx compiler
- Integrated JN516x Flash Programmer
- MinGW CLI (Command Line Interface)

For full details of this toolchain, refer to the *BeyondStudio for NXP Installation and User Guide (JN-UG-3098)*.



Note: Application development for the JN516x device is intended to be conducted in the 'BeyondStudio for NXP' IDE supplied in the toolchain. It is, however, possible to develop your application code using another editor and to build it from the command line using makefiles.

3.2.2 IEEE 802.15.4 SDK (JN-SW-4163)

The IEEE 802.15.4 SDK includes the IEEE 802.15.4 stack software and the following components:

- 802.15.4 Stack APIs for developing wireless network applications
- Application Queue APIs for optional use in conjunction with the above API
- JN516x Integrated Peripherals APIs for interacting with on-chip peripherals
- LPRF Board APIs for interacting with evaluation kit board resources

3.2.3 ZigBee Home Automation SDK (JN-SW-4168)

The ZigBee Home Automation SDK includes the ZigBee PRO stack software, the Home Automation (HA) profile and the following components:

- ZigBee PRO APIs for developing wireless network applications
- Home Automation APIs for incorporating HA functionality in applications
- ZigBee Cluster Library (ZCL) and associated APIs
- JenOS APIs for interacting with the RTOS and system management modules
- 802.15.4 Stack APIs
- JN516x Integrated Peripherals APIs for interacting with on-chip peripherals
- LPRF Board APIs for interacting with evaluation kit board resources
- ZPS and JenOS Configuration Editors (NXP plug-ins for Eclipse/BeyondStudio)

The ZigBee Home Automation SDK is supplied in the same installer as the ZigBee Light Link SDK (JN-SW-4168).

3.2.4 ZigBee Light Link SDK (JN-SW-4168)

The ZigBee Light Link SDK includes the ZigBee PRO stack software, the ZigBee Light Link (ZLL) profile and the following components:

- ZigBee PRO APIs for developing wireless network applications
- ZigBee Light Link APIs for incorporating ZLL functionality in applications
- ZigBee Cluster Library (ZCL) and associated APIs
- JenOS APIs for interacting with the RTOS and system management modules
- 802.15.4 Stack APIs
- JN516x Integrated Peripherals APIs for interacting with on-chip peripherals
- LPRF Board APIs for interacting with evaluation kit board resources
- ZPS and JenOS Configuration Editors (NXP plug-ins for Eclipse/BeyondStudio)

The ZigBee Light Link SDK is supplied in the same installer as the ZigBee Home Automation SDK (JN-SW-4168).

3.2.5 ZigBee 3.0 SDK (JN-SW-4170)

The ZigBee 3.0 SDK includes the ZigBee PRO stack software and the following components:

- ZigBee PRO APIs for developing wireless network applications
- ZigBee Base Device and associated APIs for fundamental operations such as commissioning
- ZigBee Lighting & Occupancy (ZLO) device types and associated APIs
- ZigBee Cluster Library (ZCL) and associated APIs
- JN51xx Core Utilities (JCU) and associated APIs for system management
- 802.15.4 Stack APIs
- JN516x Integrated Peripherals APIs for interacting with on-chip peripherals
- LPRF Board APIs for interacting with evaluation kit board resources
- ZPS Configuration Editor (NXP plug-in for Eclipse/BeyondStudio)

3.3 Support Resources

Support resources, including user documentation and example application software, are provided for each wireless network protocol supported by the evaluation kit. These resources are available from the NXP web site (see “Support Resources” on page 7).

The main resources for each protocol are detailed in the following sub-sections:

- IEEE 802.14.5 in [Section 3.3.1](#)
- ZigBee PRO/Home Automation in [Section 3.3.2](#)
- ZigBee PRO/ZigBee Light Link in [Section 3.3.3](#)
- ZigBee 3.0 in [Section 3.3.4](#)

3.3.1 IEEE 802.15.4 Documentation

A complete list of the user documentation relevant to IEEE 802.15.4 is provided in [Table 3](#) below.

1. First study Part I of the *IEEE 802.15.4 Stack User Guide (JN-UG-3024)* to familiarise yourself with essential IEEE 802.15.4 concepts.
2. During application development, you should refer to the above User Guide for details of the API resources to use.

In addition, the Application Note *IEEE 802.15.4 Application Template (JN-AN-1174)* provides a skeleton application as a starting point for your own application coding.

Part Number	Document Title	Description
Application Coding		
JN-UG-3024	IEEE 802.15.4 Stack User Guide	Introduces IEEE 802.15.4 and details the software resources for developing wireless network applications that run directly on the IEEE 802.15.4 stack layers.
JN-AN-1174	IEEE 802.15.4 Application Template	Application Note containing a template from which to start 802.15.4 application coding.
JN-UG-3087	JN516x Integrated Peripherals API User Guide	Details the JN516x Integrated Peripherals API, used in application code to interact with on-chip peripherals.
JN-RM-2003	LPRF Board API Reference Manual	Details the Board API, used in application code to interact with components on boards supplied in the evaluation kit.
Development Tools		
JN-UG-3098	BeyondStudio for NXP Installation and User Guide	Describes how to install the toolchain and SDK, and how to use the BeyondStudio development platform.

Table 3: IEEE 802.15.4 Support Resources

3.3.2 ZigBee PRO/Home Automation Documentation

A complete list of the user documentation relevant to ZigBee PRO with the Home Automation (HA) profile is provided in [Table 4](#) below.

1. First study Part I of each of the following User Guides in order to familiarise yourself with the relevant concepts:
 - *ZigBee PRO Stack User Guide (JN-UG-3101)*
 - *ZigBee Home Automation User Guide (JN-UG-3076)*
 - *JenOS User Guide (JN-UG-3075)*
2. During application development, you should refer to the above three User Guides for details of the API resources to use, as well as to the *ZigBee Cluster Library User Guide (JN-UG-3103)*.

In addition, during application development you should refer to the Application Note *ZigBee HA Lighting with NFC (JN-AN-1221)* for an example of an HA application.

Part Number	Document Title	Description
Application Coding		
JN-UG-3101	ZigBee PRO Stack User Guide	Introduces essential ZigBee PRO concepts and details the ZigBee PRO APIs for developing applications.
JN-UG-3075	JenOS User Guide	Details the JenOS APIs used in ZigBee PRO application code to manage OS resources.
JN-UG-3076	ZigBee Home Automation User Guide	Introduces ZigBee Home Automation (HA) and details the HA API for developing applications.
JN-UG-3103	ZigBee Cluster Library User Guide	Describes the NXP implementation of the ZigBee Cluster Library (ZCL).
JN-AN-1221	ZigBee HA Lighting with NFC *	Application Note which provides an example ZigBee Home Automation application that includes NFC commissioning of the network.
JN-AN-1222	ZigBee IoT Gateway Host with NFC	Application Note which provides information and software for developing the IP host side of an IoT Gateway.
JN-AN-1223	ZigBee IoT Gateway Control Bridge	Application Note which provides information and software for developing the ZigBee Control Bridge side of an IoT Gateway.
JN-UG-3087	JN516x Integrated Peripherals API User Guide	Details the JN516x Integrated Peripherals API, used in application code to interact with on-chip peripherals.
JN-RM-2003	LPRF Board API Reference Manual	Details the functions and associated resources used to interact with components on boards supplied in the evaluation kit.
Development Tools		
JN-UG-3098	BeyondStudio for NXP Installation and User Guide	Describes how to install the toolchain and SDK, and how to use the BeyondStudio development platform.

Table 4: ZigBee PRO/HA Support Resources

* For a broader range of features, including OTA Upgrade, Green Power and more HA devices, refer to the Application Note *ZigBee Home Automation Demonstration (JN-AN-1189)*.

3.3.3 ZigBee PRO/ZigBee Light Link Documentation

A complete list of the user documentation relevant to ZigBee PRO with the ZigBee Light Link (ZLL) profile is provided in [Table 5](#) below.

1. First study Part I of each of the following User Guides in order to familiarise yourself with the relevant concepts:
 - *ZigBee PRO Stack User Guide (JN-UG-3101)*
 - *ZigBee Light Link User Guide (JN-UG-3091)*
 - *JenOS User Guide (JN-UG-3075)*
2. During application development, you should refer to the above three User Guides for details of the API resources to use, as well as to the *ZigBee Cluster Library User Guide (JN-UG-3103)*.

Part Number	Document Title	Description
Application Coding		
JN-UG-3101	ZigBee PRO Stack User Guide	Introduces essential ZigBee PRO concepts and details the ZigBee PRO APIs for developing applications.
JN-UG-3075	JenOS User Guide	Details the JenOS APIs used in ZigBee PRO application code to manage OS resources.
JN-UG-3091	ZigBee Light Link User Guide	Introduces ZigBee Light Link (ZLL) and details the ZLL API for developing applications.
JN-UG-3103	ZigBee Cluster Library User Guide	Describes the NXP implementation of the ZigBee Cluster Library (ZCL).
JN-AN-1222	ZigBee IoT Gateway Host with NFC	Application Note which provides information and software for developing the IP host side of an IoT Gateway.
JN-AN-1223	ZigBee IoT Gateway Control Bridge	Application Note which provides information and software for developing the ZigBee Control Bridge side of an IoT Gateway.
JN-UG-3087	JN516x Integrated Peripherals API User Guide	Details the JN516x Integrated Peripherals API, used in application code to interact with on-chip peripherals.
JN-RM-2003	LPRF Board API Reference Manual	Details the functions and associated resources used to interact with components on boards supplied in the evaluation kit.
Development Tools		
JN-UG-3098	BeyondStudio for NXP Installation and User Guide	Describes how to install the toolchain and SDK, and how to use the BeyondStudio development platform.

Table 5: ZigBee PRO/ZLL Support Resources

3.3.4 ZigBee 3.0 Documentation

A complete list of the user documentation relevant to ZigBee 3.0 is provided in [Table 4](#) below.

1. First study Part I of the *ZigBee 3.0 Stack User Guide (JN-UG-3113)* in order to familiarise yourself with the relevant concepts.
2. During application development, you should refer to:
 - *ZigBee 3.0 Stack User Guide (JN-UG-3113)*
 - *ZigBee Devices User Guide (JN-UG-3114)*
 - *ZigBee Cluster Library User Guide (JN-UG-3115)*

Depending on your application, you may also wish to refer to one or more of the following Application Notes that provide example code which may be used as a starting point or template:

- *ZigBee 3.0 Base Device (JN-AN-1217)*
- *ZigBee 3.0 Light Bulbs (JN-AN-1218)*
- *ZigBee 3.0 Controller and Switch (JN-AN-1219)*
- *ZigBee 3.0 Sensors (JN-AN-1220)*
- *ZigBee 3.0 Control Bridge (JN-AN-1216)*

Part Number	Document Title	Description
Application Coding		
JN-UG-3113	ZigBee 3.0 Stack User Guide	Introduces essential ZigBee PRO concepts and details the APIs for developing ZigBee 3.0 applications.
JN-UG-3114	ZigBee Devices User Guide	Details the ZigBee Base Device and the ZigBee Lighting & Occupancy (ZLO) device types.
JN-UG-3115	ZigBee Cluster Library User Guide	Describes the NXP implementation of the ZigBee Cluster Library (ZCL) for ZigBee 3.0.
JN-UG-3119	ZigBee Green Power User Guide	Describes the use of ZigBee Green Power (GP) on the JN516x microcontrollers.
JN-UG-3116	JN51xx Core Utilities User Guide	Details software utilities that may be incorporated in ZigBee 3.0 applications.
JN-AN-1216	ZigBee 3.0 Control Bridge	Provides an example application for a ZigBee Control Bridge to facilitate an IP connection.
JN-AN-1217	ZigBee 3.0 Base Device	Provides example ZigBee Base Device code, required in all ZigBee 3.0 applications.
JN-AN-1218	ZigBee 3.0 Light Bulbs	Provides example applications for ZigBee lighting devices.
JN-AN-1219	ZigBee 3.0 Controller and Switch	Provides example applications for ZigBee controller and switch devices.
JN-AN-1220	ZigBee 3.0 Sensors	Provides example applications for ZigBee sensor devices.

Table 6: ZigBee 3.0 Support Resources

Part Number	Document Title	Description
JN-UG-3087	JN516x Integrated Peripherals API User Guide	Details the JN516x Integrated Peripherals API, used in application code to interact with on-chip peripherals.
JN-RM-2003	LPRF Board API Reference Manual	Details the functions and associated resources used to interact with components on boards supplied in the evaluation kit.
Development Tools		
JN-UG-3098	BeyondStudio for NXP Installation and User Guide	Describes how to install the toolchain and SDK, and how to use the BeyondStudio development platform.

Table 6: ZigBee 3.0 Support Resources

Chapter 3
Where Next?

Appendices

A. Kit Hardware

This appendix details the hardware devices supplied in the JN516x-EK004 evaluation kit (see [Section 1.1](#) for full kit contents):

- Carrier Boards - see [Appendix A.1](#)
- Lighting/Sensor Expansion Boards - see [Appendix A.2](#)
- Generic Expansion Boards - see [Appendix A.3](#)
- JTAG Expansion Board - see [Appendix A.4](#)
- JN5169 Modules - see [Appendix A.5](#)
- JN5169 USB Dongles - see [Appendix A.6](#)
- Raspberry Pi Board (for IoT Gateway) - see [Appendix A.7](#)
- NFC Controller - see [Appendix A.8](#)
- NTAG I²C plus Antenna Boards - see [Appendix A.9](#)

A.1 Carrier Boards

The two carrier boards (part number: DR1174) supplied in the evaluation kit are physically identical. They are both pre-fitted with an Arduino-compatible expansion board and a JN5169-based module with integrated antenna (see [Section 1.1](#)).

Each Carrier Board has the following features (also refer to [Figure 15](#) and [Figure 16](#)):

- JN51xx module site
- Arduino-compatible header (in four parts)
- 10-way JTAG debug header
- 40-way header for NTAG I²C plus Antenna Board (see [Appendix A.9](#))
- Analogue interface header for access to JN51xx DACs and comparators
- USB Mini B port for JN51xx module programming or UART connection
- FTDI FT232 driver chip for USB port
- On-board 4-Mbit Flash memory device (connected to SPISSZ line of module)
- On-board 32-kHz crystal oscillator (connected across DIO9 and DIO10 of module but not routed to Arduino-compatible header)
- 2 apertures for push-through antennas
- Reset button (RST)
- User-input button (labelled DIO8 and connected to DIO8 of JN51xx module)
- 4 green ultra low-power LEDs for Power (LD1), UART Tx (D1), UART Rx (D2) and SPI clock (LD4)
- 2 green LEDs for debug purposes (D3 connected to DIO6 and D6 connected to DIO3 of JN51xx module)
- Jumpers for board configuration (J4, Low PWR, JTAG, SPI, Mod PWR)
- Powered from one of the following (see [Appendix A.1.1](#)):
 - 4 AAA batteries on the board
 - 2 AAA batteries (low-power mode) on the board
 - External 7-12V DC supply
 - External device via USB port



Note: The NTAG I²C plus Antenna Board connects to the 40-way header of the Carrier Board via the supplied adaptor (OM15044) for the purpose of NFC commissioning - see [Appendix A.9](#).

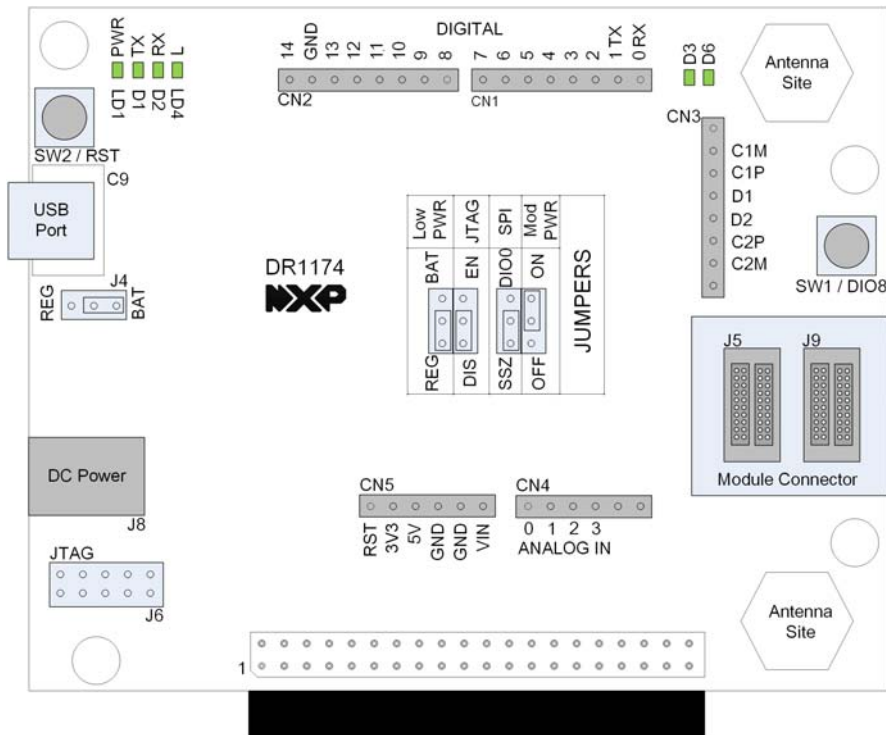


Figure 15: Carrier Board Layout



Figure 16: Carrier Board (with a JN516x Module)

Appendices

Note the following:


- The expansion board is an Arduino-compatible shield attached to the Arduino-compatible header.
- The JN5169-based module which is pre-fitted to the module site has been pre-loaded with the ZigBee Smart Home Demonstration.
- The Carrier Boards are supplied with the correct hardware configurations for the pre-loaded ZigBee Smart Home Demonstration. Once you have run this demonstration, you can re-configure the Carrier Boards as you wish - for example, by replacing the fitted module with another JN516x module and/or replacing the expansion board with a different expansion board.
- The antenna aperture can accommodate a push-through external antenna (not supplied).
- If the board is powered using the 12V DC PSU provided in the evaluation kit, it is possible to power two or more boards from this one power supply by daisy-chaining them together using the supplied cable.
- The 12V DC PSU for the Carrier Boards accepts universal input voltages (100-240V, AC 50/60Hz) but is fitted with a European plug. It can be used outside Europe by plugging it into the universal AC outlet socket of the 5V DC PSU.
- In low-power mode, which uses just 2 AAA batteries, only the JN516x module is powered (this mode can be used to make low-current measurements).
- The board is supplied with the J4 jumper configured for batteries as the power source (see table below).
- The board is supplied with access to the on-board Flash memory device enabled through the SPI jumper (see table below).
- The main jumpers and their default positions (as supplied) are as follows:


Jumper	Use	Default Setting
J4	Selects power source - batteries (BAT) or external DC/USB (REG)	BAT (batteries)
Low PWR	Enables (BAT) or disables (REG) low-power mode	REG (normal-power mode)
JTAG	Enables (EN) or disables (DIS) mapping from UART0 to JTAG header (J6)	DIS (UART0 mapped to USB port)
SPI	Enables (SSZ) or Disables (DIO0) SPI access to on-board Flash memory	SSZ (on-board Flash accessible)
Mod PWR	Enables (ON) or disables (OFF) power to JN516x module	ON (power to JN516x module)

- To return to the factory settings (including for the fitted JN516x module), wait at least 2 seconds following power-up, then press the Reset (RST) button on the Carrier Board 4 times with less than 2 seconds between consecutive presses.
- For the NFC commissioning of the wireless node into a network, the supplied NTAG I²C plus Antenna Board can be connected to the 40-way header on the Carrier Board via the supplied adaptor to provide an NFC tag, which can be used with the supplied PN7120 NFC Controller (see [Appendix A.8](#)).

A.1.1 Power Source Selection

The Carrier Board can be used in normal-power mode or low-power mode, selected using the **J4** and **Low PWR** jumpers. For the locations of these jumpers, refer to [Figure 17](#).

 **Note:** The **Low PWR** jumper is hidden under the expansion board, which must be removed in order to change the jumper setting - there are two jumper positions, **BAT** and **REG**.

 **Tip:** If batteries are being used and the board is going to be left in the idle state for a significant length of time, the batteries should be removed in order to conserve their power.

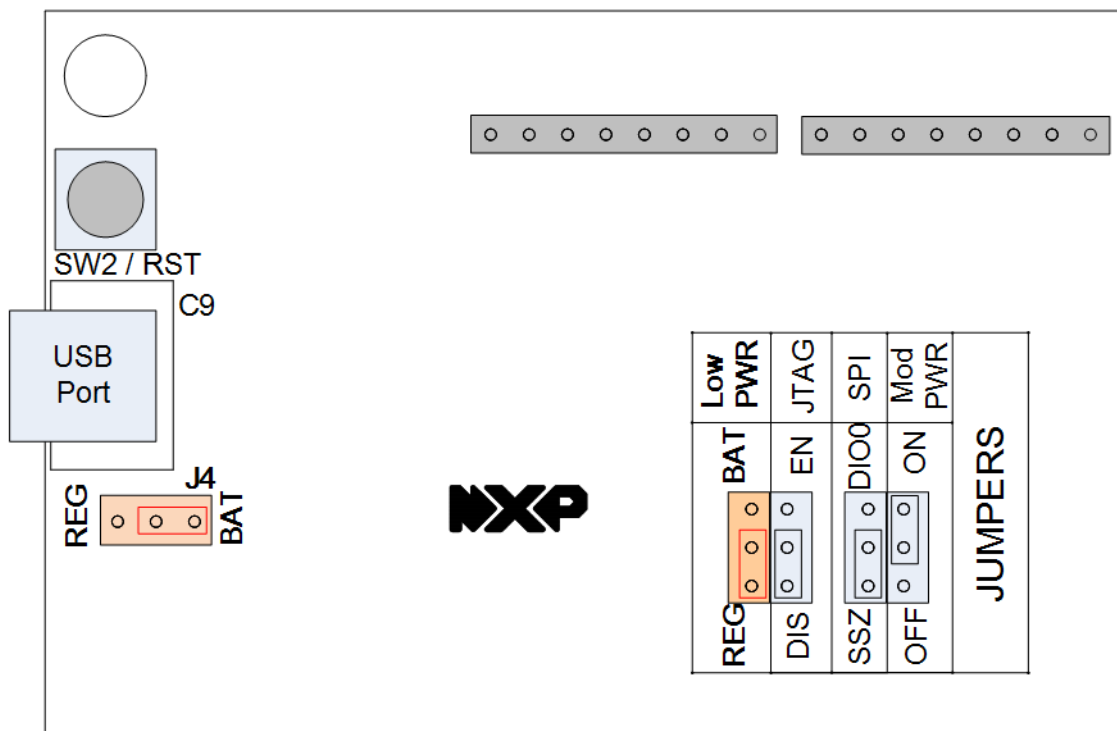


Figure 17: Locations of Power Selection Jumpers (J4 and Low PWR)

Normal-Power Mode

In normal-power mode, the board can be powered from one of the following:

- 4 AAA batteries, connected on the underside of the board
- External 7-12V DC supply via 2.1-mm connector
- External device via USB Mini B connector

The voltage regulator must be selected for the external DC and USB power supplies by putting the jumper **J4** into the **REG** position, as illustrated in [Figure 18](#) - the regulator is not needed for battery power.

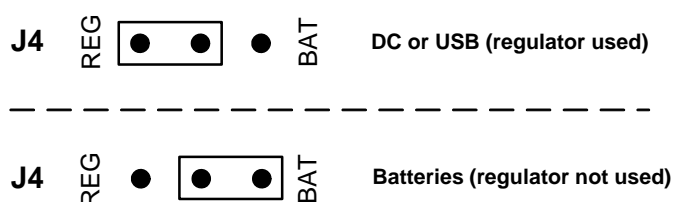




Figure 18: Regulator Selection (using J4 Jumper)

For all sources in normal-power mode, the **Low PWR** jumper must be put in the **REG** position (see [Low-Power Mode](#) below).



Caution: When using an external DC Power Supply Unit (PSU) with a Carrier Board, only a **7-12V** PSU with a 2.1-mm connector must be used. The 12V PSU provided in the evaluation kit can be used for a Carrier Board.



Note: The Carrier Boards can be powered from the same 12V DC PSU by using the supplied 5-way daisy-chain power cable to link them together. The Raspberry Pi board must be powered from the dedicated 5V DC PSU.

Low-Power Mode

In low-power mode, the board is powered from 2 AAA batteries connected in the lower two battery holders (nearest to the 40-way header) on the underside of the board. In this mode, only the JN516x module is powered (this mode can be used to make low-current measurements). To select low-power mode, the **Low PWR** jumper must be put in the **BAT** position, as illustrated in [Figure 19](#). The jumper J4 (used in normal-power mode) has no effect in low-power mode and can be left in any position.

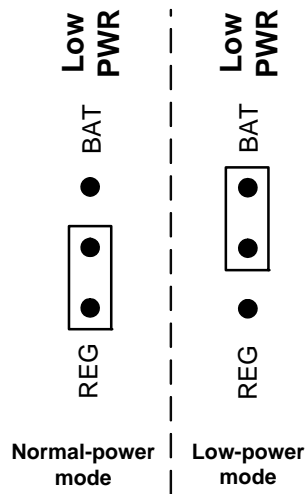


Figure 19: Lower-Power Mode Selection (using Low PWR Jumper)

A.1.2 Pre-assembled Boards

As indicated above, each Carrier Board is supplied pre-fitted with a JN5169 module with integrated antenna and an expansion board.

Figure 20 below shows a pre-assembled board fitted with a Generic Expansion Board (DR1199).

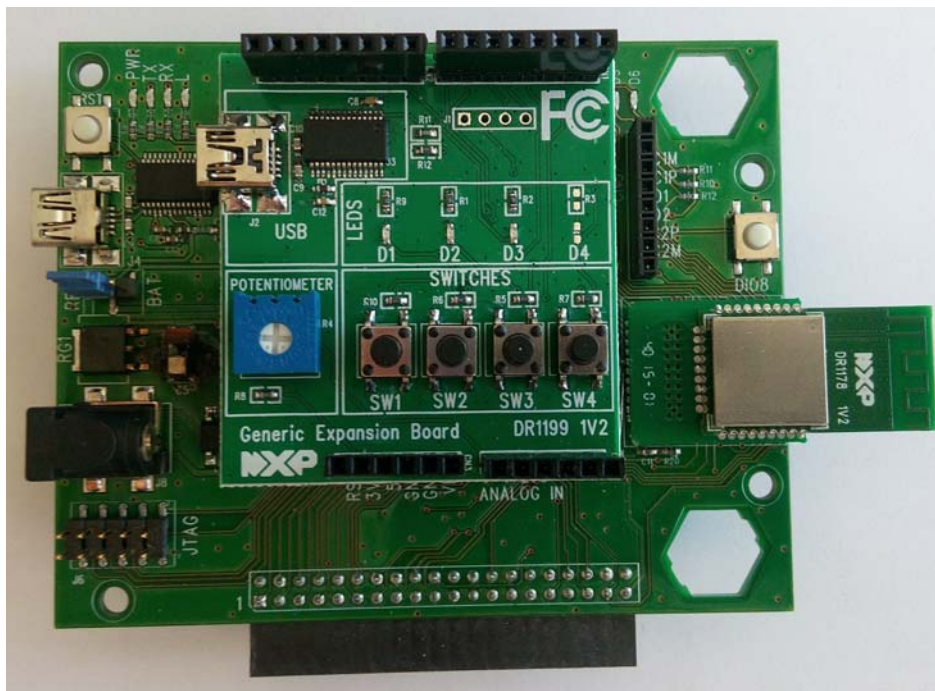


Figure 20: Carrier Board Fitted with Module and DR1199 Expansion Board

Appendices

Figure 21 below shows a pre-assembled board fitted with a Lighting/Sensor Expansion Board (DR1175).

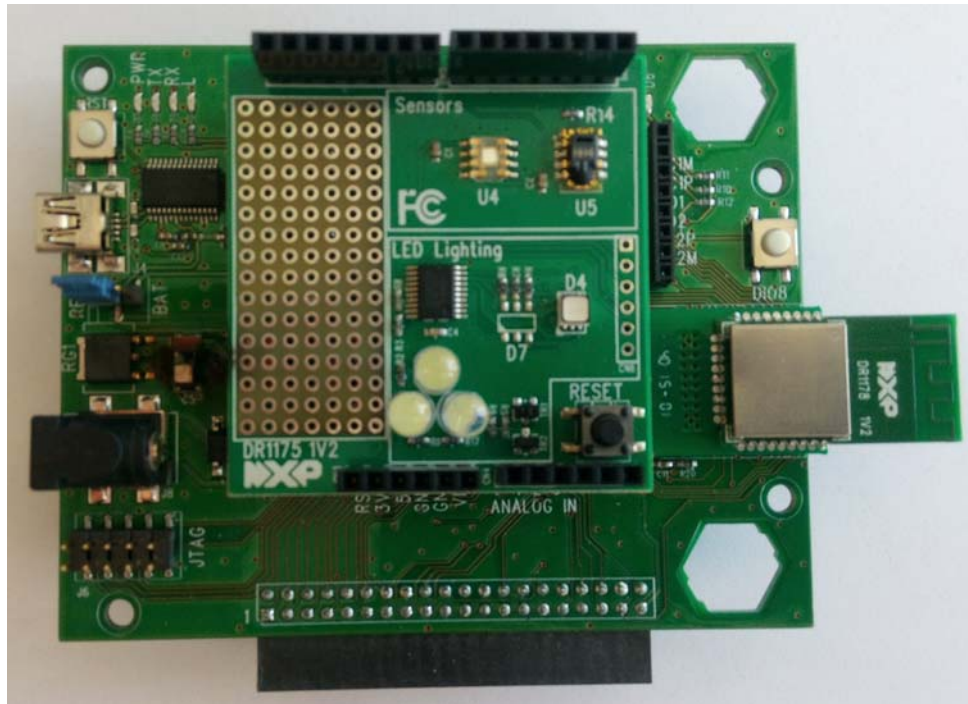


Figure 21: Carrier Board Fitted with Module and DR1175 Expansion Board

The expansion boards are described in [Appendix A.2](#), [Appendix A.3](#) and [Appendix A.4](#). The JN5169 modules are described in [Appendix A.5](#).



Tip 1: The pre-assembled board shown in [Figure 20](#) is also available to purchase separately as the JN5169XK010 ZigBee Generic Node Expansion Kit. See [“Optional Components” on page 7](#).

Tip 2: The pre-assembled board shown in [Figure 21](#) is also available to purchase separately as the JN5169XK020 ZigBee Lighting/Sensor Node Expansion Kit. See [“Optional Components” on page 7](#).

A.2 Lighting/Sensor Expansion Boards

The Lighting/Sensor Expansion Board (part number: DR1175) is an Arduino-compatible shield that is supplied pre-fitted to the Arduino-compatible header of one of the Carrier Boards in the evaluation kit (see [Appendix A.1](#)).



Note: For full details of the board, refer to the *Carrier Board and Expansion Boards Reference Design (JN-RD-6036)*, available from the NXP web site (see [“Support Resources” on page 7](#)).

The Lighting/Sensor Expansion Board has the following features (also see [Figure 22](#)):

- Arduino-compatible expansion header (in four parts, CN1-CN4)
- 3 white dimmable LEDs:
 - Controlled in unison by Pulse Width Modulation (PWM)
 - Driven by the PWM output of JN516x Timer 1 on the module on the Carrier Board, via DIO3 on the module (for details of the JN516x timers and DIOs, refer to the datasheet for the relevant chip).
- Colour LED module containing 3 miniature LEDs:
 - LEDs are independently controlled by PWM from an on-chip multi-channel PMW controller, connected via the 2-wire Serial Interface (see below)
 - Connected to the 2-wire Serial Interface (I²C) of the JN516x module on the Carrier Board, via DIO14 (clock) and DIO15 (data) on the module
- Humidity/temperature sensor (of the type SHT1x) connected to the JN516x module on the Carrier Board, via DIO12 (data) and DIO17 (clock) on the module
- Light sensor (of the type TSL2550), connected to the 2-wire Serial Interface (I²C) of the JN516x module on the Carrier Board, via DIO14 (clock) and DIO15 (data) on the module
- Reset button (to reset the Carrier Board)



Caution: *The LEDs on the Lighting/Sensor Expansion Board are very bright at maximum intensity. To avoid damage to your eyes, do not look into them directly for an extended period of time.*



Note: If NFC commissioning is to be used with the Lighting/Sensor Expansion Board, the J16 jumper on the OM15044 adaptor board must be located in the "DIO17/Lighting" position. For more information, refer to [Appendix A.9](#).

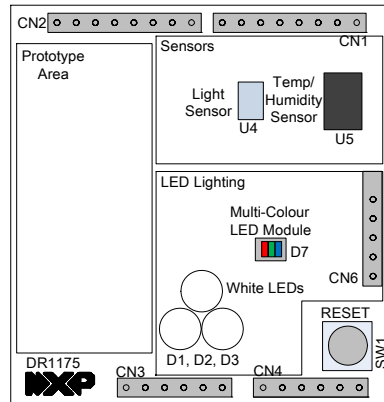


Figure 22: Lighting/Sensor Expansion Board Layout

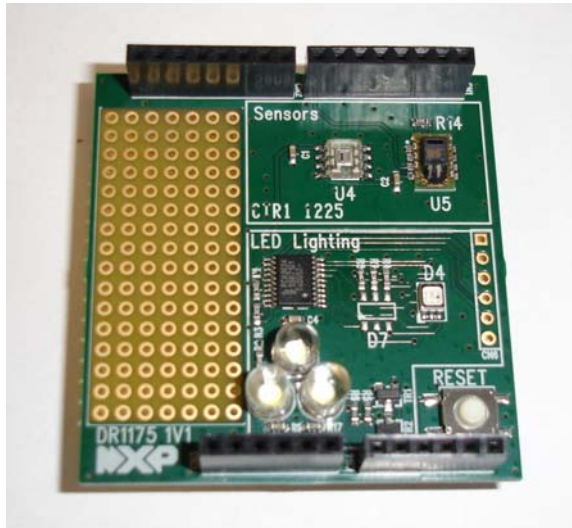


Figure 23: Lighting/Sensor Expansion Board

A.3 Generic Expansion Boards

The Generic Expansion Board (part number: DR1199) is an Arduino-compatible shield that is supplied pre-fitted to the Arduino-compatible header of one of the Carrier Boards in the evaluation kit (see [Appendix A.1](#)).



Note: For full details of the board, refer to the *Carrier Board and Expansion Boards Reference Design (JN-RD-6036)*, available from the NXP web site (see [“Support Resources” on page 7](#)).

The Generic Expansion Board has the following features (also see [Figure 24](#)):

- Arduino-compatible expansion header (in four parts, CN1-CN4)
- 4 user-input buttons (SW1, SW2, SW3, SW4) connected to DIOs on the JN516x module on the Carrier Board:
 - SW1 is connected to DIO1
 - SW2 is connected to DIO11
 - SW3 is connected to DIO12
 - SW4 is connected to DIO17
- 4 LEDs (D1, D2, D3, D4) connected to DIOs on the JN516x module on the Carrier Board:
 - D1 is connected to DIO0
 - D2 is connected to DIO13 (same as for the PWM output used on the Lighting/Sensor Expansion Board to drive the three white LEDs)
 - D3 is connected to DIO16
 - D4 is an optional LED which is not fitted on the supplied board and shares DIO1 with button SW1
- Potentiometer connected to JN516x ADC1, with a 0-1.8V DC voltage swing
- USB Mini B port to provide access to JN516x UART1 via DIO14 and DIO15 of JN516x module (or alternatively to UART0, depending on the internal configuration of the device)
- FTDI FT232 driver chip for USB port

PCB pads for 4-way GPIO header (3V3, VCC, DIO14, DIO15), where DIO14 and DIO15 can be used to access JN516x UART1 (in this case, the FTDI driver chip for the USB port must be isolated by removing resistors R11 and R12)



Note: If NFC commissioning is to be used with the Generic Expansion Board, the J16 jumper on the OM15044 adaptor board must be located in the "DIO/Switch" position. For more information, refer to [Appendix A.9](#).

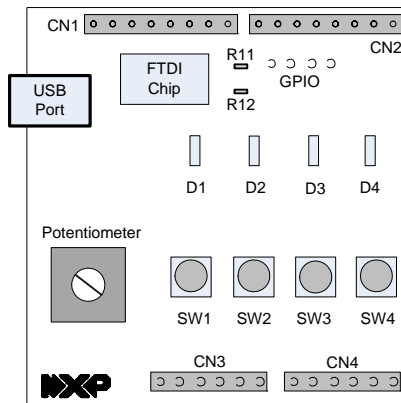


Figure 24: Generic Expansion Board Layout



Figure 25: Generic Expansion Board

A.4 JTAG Expansion Board

The JTAG Expansion Board (part number: DR1222) is an Arduino-compatible shield that is supplied in the evaluation kit. It can be fitted to the Arduino-compatible header and JTAG header of a Carrier Board.



Note: For full details of the board, refer to the *Carrier Board and Expansion Boards Reference Design (JN-RD-6036)*, available from the NXP web site (see [“Support Resources” on page 7](#)).

The JTAG Expansion Board has the following features:

- Arduino-compatible expansion header (in four parts, CN1-CN4)
- 10-way JTAG connector (J2) on side (for connection to external JTAG adaptor)
- 20-way JTAG connector (J1) on side (for connection to external JTAG adaptor)
- 10-way JTAG header on underside (for connection to Carrier Board)
- Three jumpers (J3, J4, J5) for pin selection on the two external JTAG connectors (J1 and J2)

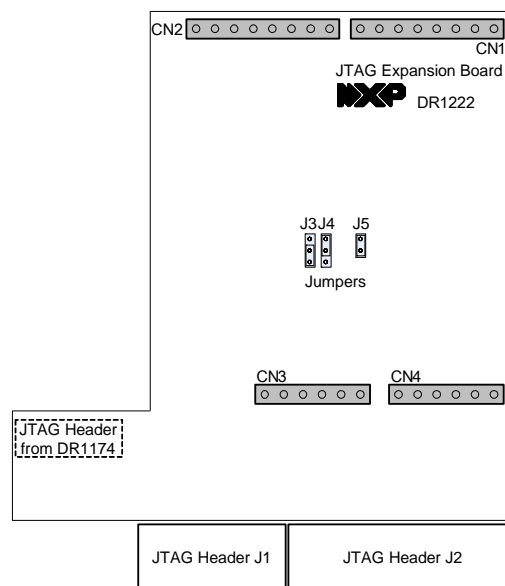


Figure 26: JTAG Expansion Board Layout

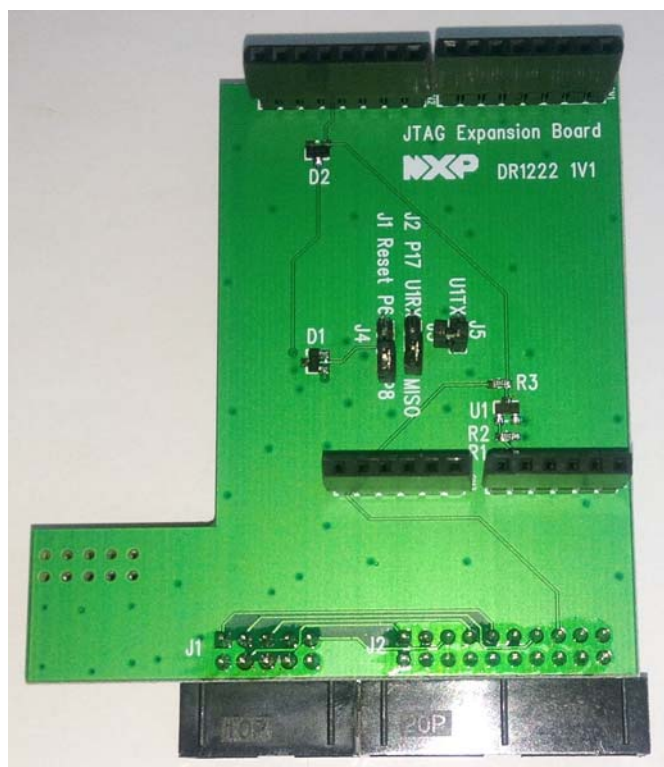


Figure 27: JTAG Expansion Board

Note the following:

- A 10-way JTAG header is provided on the Carrier Board. However, this header does not include the reset signal. Use of the JTAG Expansion Board is necessary in order to connect this signal via a pin on the CN5 connector of the Carrier Board.
- On the Carrier Board, there is a jumper labelled JTAG (see [Figure 15](#) on page 49). This jumper is normally in the **DIS** position. When using the JTAG Expansion Board, first move the jumper to the **EN** position. This remaps the JN516x UART0 signals from the USB serial connection to the JTAG port.
- The JTAG Expansion Board must be connected directly to a Carrier Board. Another expansion board (e.g. Lighting/Sensor Expansion Board) can then be stacked on top of the JTAG Expansion Board.
- The wired connection between the JTAG Expansion Board and the development PC must be made via a JTAG adaptor/dongle (e.g. Beyond Debug Key).
- The JTAG connector used (J1 or J2) depends on the external JTAG adaptor/dongle. The J1 connector corresponds to the Altera pinout.
- If you are using the BeyondStudio for NXP development platform, you should refer to the Application Note *JN516x JTAG Debugging in BeyondStudio (JN-AN-1203)* for guidance on debugging your application via the JTAG Expansion Board. This Application Note is available from the NXP web site (see [“Support Resources”](#) on page 7).

A.5 JN5169 Modules

The evaluation kit includes two JN5169-based modules that plug onto the DR1174 Carrier Board. Each is a JN5169-001-M00 module mounted on a DR1179 Mezzanine Board, the part number of the combined component being JN5169-001-T00. This module features a JN5169-001 wireless microcontroller and an integrated PCB antenna.



Note: The JN5169 modules that are pre-fitted to Carrier Boards are pre-programmed with the ZigBee Smart Home Demonstration. Firmware re-programming of JN516x modules is described in [Appendix C.1](#).

A.6 USB Dongles

Two JN5169 USB Dongles (part number: OM15020) are supplied in the kit. One is intended for use in the ZigBee Smart Home Demonstration and the other can be employed in some other capacity, such as packet sniffing (see below).

The JN5169 USB Dongle has the following features:

- JN5169-001 wireless microcontroller (mounted directly on the board)
- Integrated PCB antenna
- FTDI FT232 driver chip for USB connection
- Two LEDs (one green, one orange)



Figure 28: JN5169 USB Dongle



Note: The JN5169 USB Dongle is also available to purchase separately (part number OM15020). See [“Optional Components” on page 7](#).

Appendices

The two dongles are pre-programmed with different software, as described below. The dongles can be re-programmed with other applications (firmware re-programming of the JN5169 USB Dongles is described in [Appendix C.2](#)).

Control Bridge

The JN5169 USB Dongle with a green sticker is pre-programmed as a Control Bridge that can be used in the ZigBee Smart Home Demonstration. This software allows the dongle to be plugged into a USB port of the Raspberry Pi board (see [Appendix A.7](#)) to form an IoT Gateway:

- The dongle provides the interface with the wireless network (WPAN) side of the demonstration system, also acting as the network Co-ordinator
- The Raspberry Pi board provides the interface with the IP (LAN/WAN) side of the demonstration system

For more details of the demonstration system, refer to [Section 1.5](#) and [Chapter 2](#).



Note: This JN5169 USB Dongle should not be plugged directly into a USB port of the Raspberry Pi board. This connection should be made via the USB-to-USB cable, especially when the Wi-Pi adaptor is also used.

Packet Sniffer

The JN5169 USB Dongle with a red sticker is pre-programmed with an NXP packet sniffer application. To use this application, the dongle should be plugged into the USB port of a PC. You will need to install and use the Ubiqua Protocol Analyzer software on the PC. This software which can be obtained from the Ubilogix web site (www.ubilogix.com/products/ubiqua). A free trial of this software is available.

A.7 Raspberry Pi

The Raspberry Pi single-board computer (Model B+) is intended to be used in an IoT Gateway to form an interface between a low-power wireless network and an IP-based network (see [Section 1.3](#)). It is supplied with accessories consisting of:

- Dedicated 5V DC power supply unit (universal type) with USB port
- microSD card (pre-installed and pre-programmed with NXP firmware)
- Wi-Pi Raspberry Pi 802.11n wireless adaptor
- NXP PN7120 NFC Controller (described in [Appendix A.8](#))
- RJ45 Ethernet cable
- 'USB to Micro USB' cable (for power supply)

The Raspberry Pi board and accessories are shown in [Figure 29](#) below (the NFC Controller is shown detached from the Raspberry Pi board, the 5V DC PSU is not shown and a JN5169 USB Dongle is also shown).

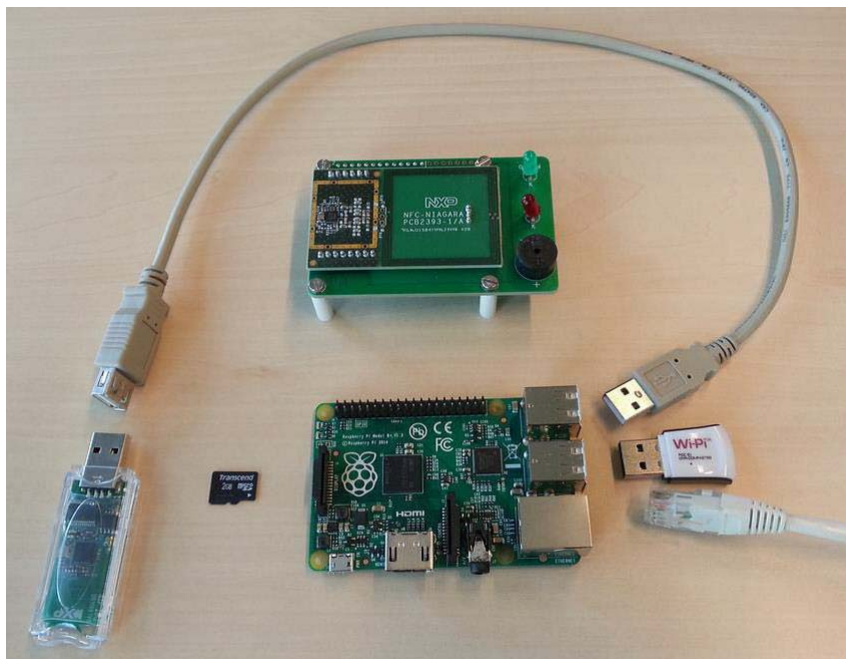


Figure 29: Raspberry Pi Board and Accessories

Appendices

Note the following:

- The 'USB to Micro USB' cable allows the board to be powered from a USB port on the 5V DC PSU.
- A JN5169 USB Dongle can be plugged into any of the USB ports of the Raspberry Pi board to form an IoT Gateway (see [Section 1.3](#)). However, it is recommended that this connection is made via the supplied USB-to-USB cable, especially when the Wi-Pi adaptor is also used (in order to ensure isolation between the dongle and adaptor).
- The JN5169 USB Dongle that is to be connected to the Raspberry Pi board for the ZigBee Smart Home Demonstration is the one with a green sticker (see [Appendix A.6](#)). This dongle is pre-programmed as a Control Bridge.
- The Wi-Pi wireless adaptor can be connected directly to any of the USB ports of the Raspberry Pi board in order to provide Wi-Fi connectivity.
- The supplied NFC Controller (see [Appendix A.8](#)) can be connected to the Raspberry Pi board to facilitate NFC commissioning (see [Section 1.4.1](#)).
- The IoT Gateway software and documentation are available two Application Notes: *ZigBee IoT Gateway Host with NFC (JN-AN-1222)* and *ZigBee IoT Gateway Control Bridge (JN-AN-1223)*.

A.8 NFC Controller

The NXP PN7120 NFC Controller is supplied in the JN516x-EK004 evaluation kit to facilitate NFC commissioning (see [Section 1.4.1](#)), allowing an NFC tag attached to a Carrier Board to be read. The NFC Controller Board is provided pre-mounted on a Raspberry Pi Interface Board - both components come from the NXP PN7120 NFC Controller Single Board Computer (SBC) Kit. This NFC Controller assembly attaches to a Raspberry Pi board, as shown in [Figure 30](#) below.

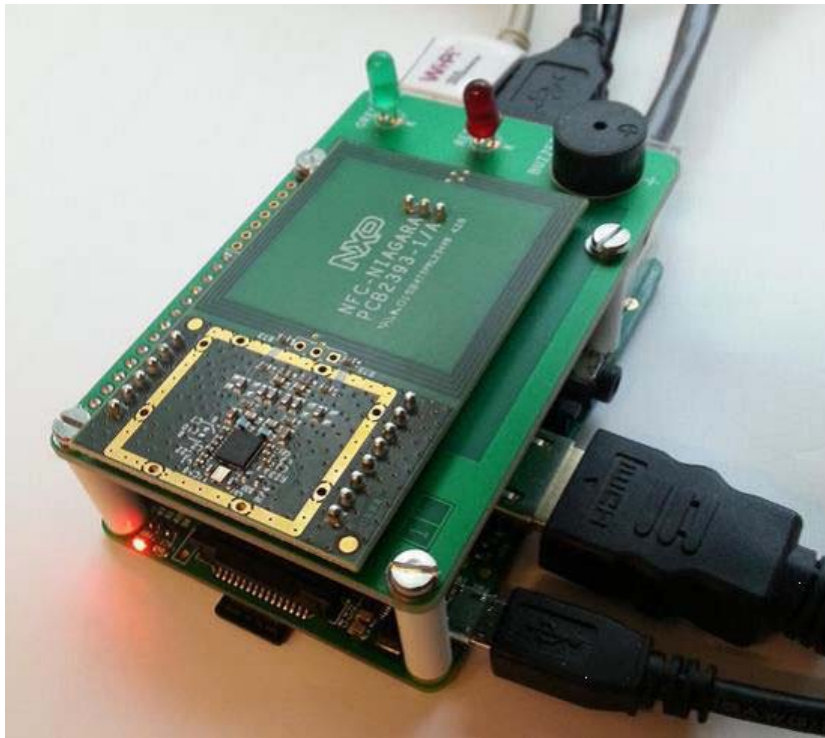


Figure 30: NFC Controller Assembly on Raspberry Pi Board

The supplied NFC Controller assembly is a high-performance fully NFC-compliant expansion board for the Raspberry Pi. It complies with the Reader mode, P2P mode and Card Emulation mode standards. The board features an integrated high-performance RF antenna to ensure good interoperability with NFC devices.

To use the supplied assembly of boards for NFC commissioning, you must mount it on the Raspberry Pi board. For more information and operational instructions relating to the NFC Controller, refer to the following NXP documentation:

- *AN11646: PN7120 NFC Controller SBC Kit Quick Start Guide*
- *UM10878: PN7120 NFC Controller SBC Kit User Manual*

The above documents can be obtained from the NXP web site.

To join a node to the network using NFC commissioning, the supplied NTAG I²C plus Antenna Board (NFC connected tag) must be connected to the Carrier Board of the node via the supplied adaptor - see [Appendix A.9](#).

A.9 NTAG I²C plus Antenna Boards and Adaptors

The supplied NXP NTAG I²C plus Antenna Boards (NT322E) each provide an 'NFC connected tag' that can be attached to a DR1174 Carrier Board to allow the node to be introduced into a network via contactless NFC commissioning. During this commissioning, the PN7120 NFC Controller interacts with the NFC tag (see [Appendix A.8](#)). The board is shown in [Figure 31](#) below.



Figure 31: NTAG I²C plus Antenna Board

The Antenna Board connects to the DR1174 Carrier Board via the 40-way interface on the Carrier Board (see [Appendix A.1](#)). This connection is made via the supplied adaptor (OM15044). A full board assembly is shown in [Figure 32](#) below.



Figure 32: Fully Assembled Board with NFC Tag

The two Antenna Boards are marked with red and green stickers to indicate which expansion boards they must be used with. In addition, the J16 jumper on the OM15044 adaptor must be positioned correctly for the expansion board that is fitted to the Carrier Board (also see [Figure 33](#) below):

- If the **Generic Expansion Board (DR1199)** is fitted:
 - the Antenna Board with the red sticker must be used
 - the J16 jumper must be located in the “DIO/Switch” position to route the NFC field detect signal through DIO0 of the JN516x device
- If the **Lighting/Sensor Expansion Board (DR1175)** is fitted:
 - the Antenna Board with the green sticker must be used
 - the J16 jumper must be located in the “DIO17/Lighting” position to route the NFC field detect signal through DIO17 of the JN516x device

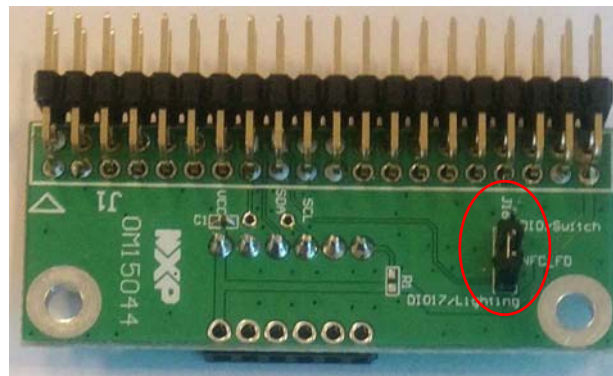


Figure 33: J16 Jumper on OM15044 Adaptor

NFC commissioning of the node can be performed by holding the Antenna Board over (but not touching) the antenna of the PN7120 NFC Controller - this action is known as an ‘NFC tap’.

For more information on the NTAG I²C plus Antenna Board, refer to the following NXP documentation (available on the NXP web site):

- *UG313110: NTAG I²C Explorer Kit user’s manual*
- *AN11597: NTAG I²C demo application for Android*

B. Installing a Packet Sniffer

One of the JN5169 USB Dongles supplied in the evaluation kit is programmed with an NXP packet sniffer application - this dongle is indicated with a red sticker. For more information on the JN5169 USB Dongles, refer to [Appendix A.6](#).

This section describes how re-program either of the dongles with the packet sniffer application. To use a dongle as a sniffer, you must also install the Ubiqua Protocol Analyzer software on the PC from which the packet sniffing will be conducted.

Proceed as follows:

1. Plug the JN5169 USB Dongle into the PC.
2. Program the sniffer application binary into the JN5169 device on the dongle:
 - The sniffer binary file **JennicSniffer_JN5169_1000000_HP.bin** is provided in the JN516x Software Developer's Kit (SDK), in the directory **<SDK root>\bstudio_nxp\sdk\JN-SW-41xx\Tools\JennicSniffer**.
 - For information on re-programming a JN5169 USB Dongle, refer to [Section C.2](#)
3. Download the Ubiqua Protocol Analyzer software and install it on the PC:
 - You can obtain the Ubiqua Protocol Analyzer software from the following web site: **www.ubilogix.com/products/ubiqua** (a free trial version is available)
 - Run the downloaded installer and follow the on-screen installation instructions
4. Start the Ubiqua Protocol Analyzer (ensure that the PC is connected to the Internet).

User documentation for the Ubiqua Protocol Analyzer is available from the above Ubiqua web site.

C. Firmware Re-programming

The following components of the JN516x-EK004 Evaluation Kit are supplied already programmed with the Smart Home Demonstration but can be re-programmed with other applications:

- JN5169 modules (on Carrier Boards) - see [Appendix C.1](#)
- JN5169 USB Dongles - see [Appendix C.2](#)

Flash Programmers

The firmware re-programming described in the sub-sections below require the use of one of the following Flash programmers:

- JN516x Flash Programmer built into 'BeyondStudio for NXP' and described in the *BeyondStudio for NXP Installation and User Guide (JN-UG-3098)*
- JN51xx Production Flash Programmer (JN-SW-4107) described in the *JN51xx Production Flash Programmer User Guide (JN-UG-3099)*



Note: The 'BeyondStudio for NXP' internal Flash programmer cannot be used to erase and re-program JN516x external Flash memory.

C.1 Re-programming JN5169 Modules

In order to re-program the Flash memory of the supplied JN5169 modules, you must connect the Carrier Board (on which the module is located) to your PC (see below). To perform the re-programming from your PC, you must use a JN51xx Flash Programmer utility (see [Flash Programmers](#) above).

When re-programming the JN516x modules on the Carrier Boards in this kit, you must use one of the supplied 'USB A to Mini B' cables to connect a USB port of your PC to the USB Mini B port of the Carrier Board. When you make this connection for the first time, you may be prompted to install the device driver for the FTDI chip which is located on the Carrier Board - if this is the case, refer to [Appendix D](#).

C.2 Re-programming JN5169 USB Dongles

In order to re-program the Flash memory of the supplied JN5169 USB Dongles, you must plug the dongle into a USB port of a PC. To perform the re-programming from your PC, you must use a JN51xx Flash Programmer utility (see [Flash Programmers](#) above).

When you make the USB connection for the first time, you may be prompted to install the device driver for the FTDI chip which is located on the dongle - if this is the case, refer to [Appendix D](#).

D. Installing the FTDI Device Driver for USB Connections

The first time that you make a USB connection between your PC and a kit component which features the FTDI FT232 chip, you may be prompted to install the device driver for the chip on your PC. You will need to obtain the driver from the Internet.

You must obtain the driver for your operating system from the VCP drivers page of the FTDI web site:

www.ftdichip.com/Drivers/VCP.htm

Download the required driver to your desktop and double-click on its icon to install. *To perform the installation, a device or cable containing an FTDI chip must be connected to a USB port of your PC.*

Revision History

Version	Date	Comments
1.0	19-Nov-2015	First release for preliminary kit
2.0	20-June-2016	Re-organised and updated for production edition of kit

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 described 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.

NXP Semiconductors

For online support resources and contact details of your local NXP office or distributor, refer to:

www.nxp.com

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for [Development Boards & Kits - Wireless](#) category:

Click to view products by [NXP](#) manufacturer:

Other Similar products are found below :

[13237ADC-BDM](#) [DA14586-00F02ATDB-P](#) [TWR-13237-KIT](#) [HEATER CLICK](#) [LOAD CELL 2 CLICK](#) [SLWRB4181B](#) [SLWRB4179B](#) [C](#)
[METER CLICK](#) [HALL CURRENT CLICK](#) [COLOR 3 CLICK](#) [COMPASS 2 CLICK](#) [HDC1000 CLICK](#) [RELAY CLICK](#) [RFID CLICK](#)
[RIVERDI CLICK](#) [CUGSM113#UFL](#) [MATRIX G CLICK](#) [3D MOTION CLICK](#) [4-20MA R CLICK](#) [THERMO CLICK](#) [MCP2003B CLICK](#)
[MCP2542 CLICK](#) [MCP25625 CLICK](#) [ATA8520-EK3-E](#) [TOUCHKEY CLICK](#) [MICROSD CLICK](#) [TRF CLICK](#) [BUTTON R CLICK](#) [I2C](#)
[ISOLATOR CLICK](#) [UNIQUE ID CLICK](#) [USB SPI CLICK](#) [CAN-SPI CLICK 3.3V](#) [CAN-SPI CLICK 5V](#) [CAP EXTEND CLICK](#)
[CAPSENSE CLICK](#) [13DOF 2 CLICK](#) [13DOF CLICK](#) [DC MOTOR CLICK](#) [WIFI2 CLICK](#) [4X4 KEY CLICK](#) [4X4 RGB CLICK](#) [DTH22](#)
[CLICK](#) [RS232 CLICK](#) [RS485 CLICK 5V](#) [RTC2 CLICK](#) [EM3588-MLR-AN-C](#) [CC3100 CLICK](#) [FLASH 3 CLICK](#) [NFC CLICK](#) [FLICKER](#)
[CLICK](#)