XBee ZigBee Cloud Kit

Getting Started Guide

Revision history—90001503

Revision	Date	Description
A	March 2016	Baseline release of this document. The XBee ZigBee Cloud Kit information formerly resided in the *XBee Gateway User Guide*. This new documents is a standalone document for the XBee ZigBee Cloud Kit. Updated the descriptions of the following terms in XBee development board components and descriptions on page 30: • Associated LED • Commissioning button • User0/TCP LED • User1/On LED • Buzzer

Trademarks and copyright

Digi, Digi International, and the Digi logo are trademarks or registered trademarks in the United States and other countries worldwide. All other trademarks mentioned in this document are the property of their respective owners.

Disclaimers

Information in this document is subject to change without notice and does not represent a commitment on the part of Digi International. Digi provides this document "as is," without warranty of any kind, expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Digi may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in this manual at any time.

Warranty

To view product warranty information, go to the following website:

http://www.digi.com/howtobuy/terms

Customer support

Digi Technical Support: Digi offers multiple technical support plans and service packages to help our customers get the most out of their Digi product. For information on Technical Support plans and pricing, contact us at 877.912.3444 or visit us at www.digi.com/support.

Online: www.digi.com/support/eservice

Contents

About XBee ZigBee Cloud Kit

The Digi™ XBee® ZigBee® Cloud Kit application makes working with your XBee Gateway and XBee modules quick and easy. The XBee ZigBee Cloud Kit web application integrates with Digi Device Cloud to enable two-way communication over the Internet, giving you remote control of your device wherever you are. This application also allows you to customize your dashboard widgets so that you can create your own connected device systems.

ZigBee is a wireless standard, from the ZigBee Alliance, that supports development of wireless IoT applications. The XBee ZigBee Cloud Kit provides the foundation you need to get started working with XBee devices using the ZigBee protocol, and build cloud-connected prototypes quickly.

This guide describes how to get started with your XBee ZigBee Cloud Kit. This guide is intended for a developer or programmer. It covers these tasks:

- Getting started with your XBee ZigBee Cloud Kit on page 6
- Learn more about the XBee ZigBee Cloud Kit on page 19
- Troubleshoot your XBee ZigBee Cloud Kit on page 24

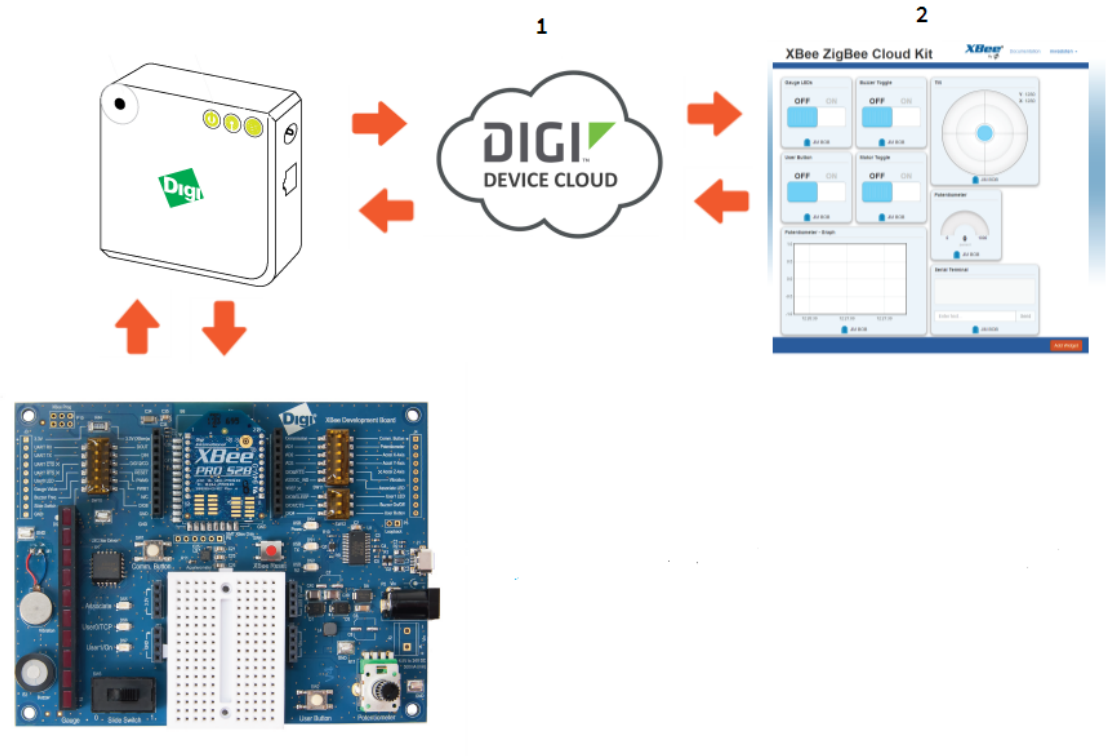
Getting started with your XBee ZigBee Cloud Kit

As you go through the steps in this kit, you will:

- Create a free Device Cloud account
- Add your XBee Gateway to your Device Cloud account
- See your device data on Device Cloud
- Walk through examples demonstrating interactions between the module and the XBee ZigBee Cloud Kit web application
- Configure widgets in the XBee ZigBee Cloud Kit web application
- Learn more on how to build your own solutions with our XBee Gateway products

Here are the steps required to setup your XBee ZigBee Cloud Kit.

1. Create a Digi™ Device Cloud account and add XBee Gateway on page 7.
2. See your device data on page 10.



To learn more about creating customized connected device solutions with your XBee ZigBee Cloud Kit. See Next steps on page 18 and Learn more about the XBee ZigBee Cloud Kit on page 19.

Create a Digi™ Device Cloud account and add XBee Gateway

To use the features of the XBee ZigBee Cloud Kit web application, you must have a Digi Device Cloud account.

Select one of the following options:

- I do not have a Device Cloud account: go to Create a Device Cloud account on page 7.
- I already have an account: go to Add your XBee Gateway to Device Cloud on page 8.

Create a Device Cloud account

To create a Device Cloud account:

1. To create your free account on Device Cloud, open a web browser and go to the Device Cloud product page.
2. Click **Free Developer Acccount/Login** and follow the steps for creating your account. If you need help creating an account, refer to the *Device Cloud User Guide*.

Log onto the XBee ZigBee Cloud Kit web application

To log onto the XBee ZigBee Cloud Kit web application:

1. From a web browser, go to the XBee ZigBee Cloud Kit login page.

2. Type the user name and password that you created in Create a Device Cloud account on page 7 and click **Log in**.

Add your XBee Gateway to Device Cloud

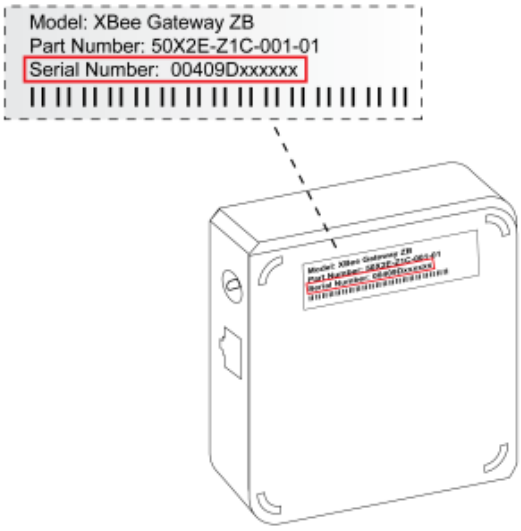
To add your XBee Gateway to Device Cloud:

1. **Optional**: Select a layout for the dashboard from the **Layout** menu. A layout is an arrangement of the widgets in the dashboard. The default layout is **Cloud Kit**, and is recommended for new XBee Gateway users. This layout includes several widgets that exercise key hardware features of the XBee development board. The **Empty** layout creates an empty dashboard and is recommended for advanced users familiar with the dashboard who want to create a custom layout.
2. To add XBee Gateway to Device Cloud, click **Add New Device**.

   The **Add a New Gateway to your Account** page appears.

3. Type the serial number and a name or description for your XBee Gateway.

   If you wrote down the XBee Gateway serial number while using the *XBee ZigBee Cloud Kit Quick Start Guide*, use it. Otherwise, locate the Serial Number on the label attached to XBee Gateway, as shown below.



4. Click **Add Device** to add XBee Gateway to your Device Cloud account. The serial number for the XBee Gateway appears in the **XBee Gateway** field.

5. On the **Dashboard Creation** page, choose the XBee RF module on the XBee development board from the **XBee Module** drop-down list.

   Tip If you don't see your device in the **XBee Module** drop-down list, click the link **Don't see your XBee?** to locate the device and see Join your XBee node to your XBee Gateway on page 35 for more detailed troubleshooting instructions.

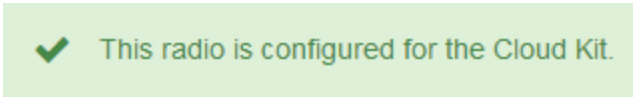
   The application checks the XBee module's configuration to make sure it can communicate with XBee Gateway.

   Add the XBee RF module on the XBee development board to Device Cloud. Your XBee Gateway device hardware must be configured to properly work with the XBee ZigBee Cloud Kit web application.

6. Choose one of the following options, when prompted, to apply the Kit Configuration:

   - **Yes (recommended)**: Configures your device hardware to work with the XBee ZigBee Cloud Kit web application.
   - **Skip**: Leaves the configuration of your device hardware as-is. If you choose this option, some data from the device hardware may not appear properly on the dashboard widgets.

   A green check mark appears when the device is successfully configured.

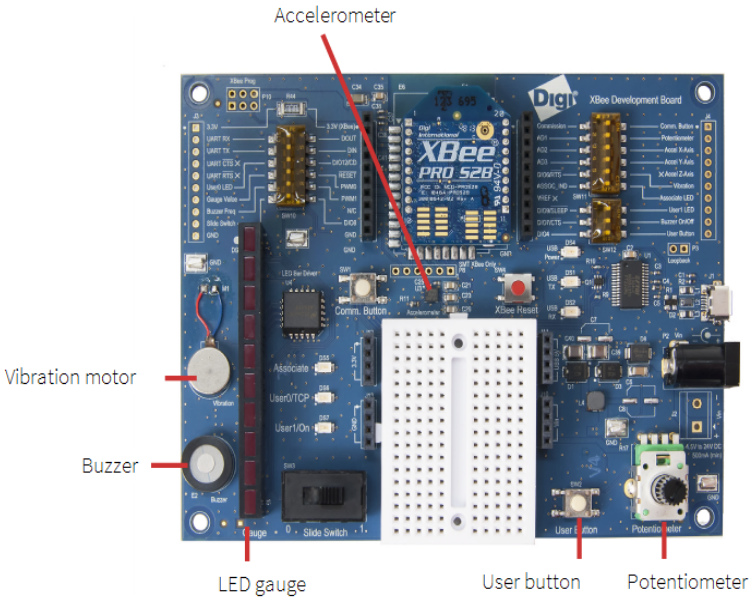
7. Click **Create Dashboard**. The dashboard appears.

See your device data

The XBee ZigBee Cloud Kit allows you to view and customize widgets for built-in components on the development board.

View and exercise widgets and development board components

By default, your dashboard displays the widgets for the components in the XBee ZigBee Cloud Kit. The widgets represent elements of the XBee module on your device hardware, and the data for those elements, in graphical form.



The following example allows you to exercise functions on the development board and observe the results using the XBee ZigBee Cloud Kit web application. This example assumes you are signed in to the XBee ZigBee Cloud Kit web application.

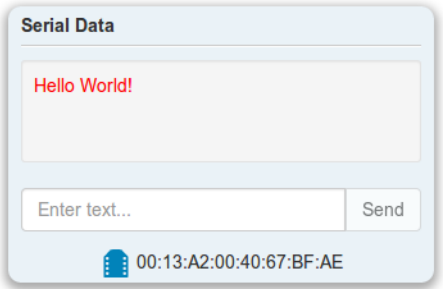
1. On the development board, press and hold the **User Button**. Observe how the **User Button** widget on the dashboard changes from OFF to ON.
2. On the dashboard, turn on the **Gauge LEDs**. Observe how the **Gauge LEDs** on the development board illuminate.
3. On the development board, turn the knob on the **Potentiometer** left or right and wait a few seconds. Observe how the Potentiometer gauge and Potentiometer graph on the dashboard change.

> **Note** The display on the dashboard relative to action on the board is dependent on the sampling rate on the XBee and potential throttling of data by Device Cloud. If you notice that changes made on the development board are not reflected on the dashboard, refresh the web page and try again.

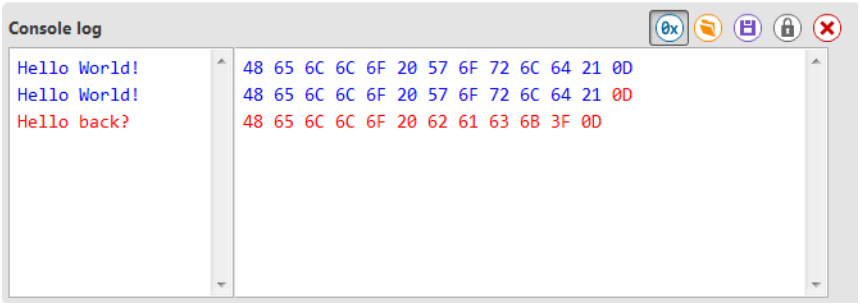
4. Exercise the serial terminal application. Use a serial terminal application such as XCTU to enter serial data. This example uses XCTU though you can use other serial terminal applications, such as HyperTerm or CoolTerm, as well.

    a. In XCTU, click the **Consoles working mode** tab. The Console log window appears.

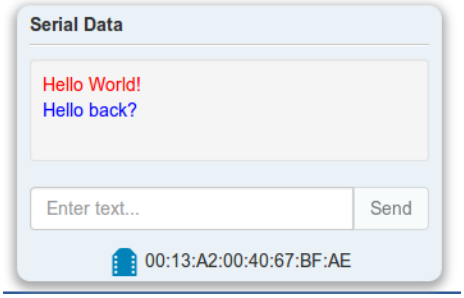
    b. In the Serial Data widget, type some serial data; for example, type **Hello World!**



    c. The XCTU console displays the serial data in text and hexadecimal format; for example, type **Hello back?**



The Serial Data widget displays the serial data:



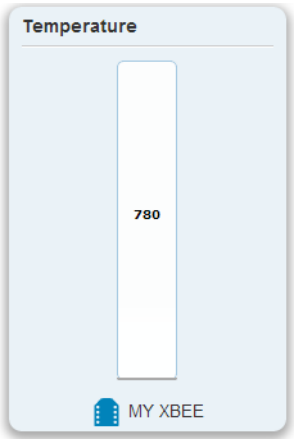
These are a few ways that the XBee ZigBee Cloud Kit web application and your development board interact with one another. To learn more about the functions of your XBee ZigBee dashboard, including how to add additional widgets, see the following sections.

Add and customize widgets

There are a variety of widgets that you can use to customize your dashboard. The following example shows how to add a progress bar widget.

1. Click **Add Widget** located on the bottom-right corner of your XBee ZigBee dashboard.
2. Select the type of widget to create from the **Widget Type** drop-down list; for example, select **Bar Graph Widget (Vertical)**.
3. Type a description of your widget in the **Label** field; for example, type **Temperature**.
4. Select the XBee Gateway device ID to attach to the widget from the **Gateway** drop-down list.
5. Select the XBee module associated with the dashboard from the **XBee Module** drop-down list.
6. Select the input stream from the device that will be attached to the widget from the **Input Stream** drop-down list.

   The rest of the settings are for advanced management of the widget. See Widget settings on page 21 for more information on these settings.
7. Click **Save** to save your widget settings. The browser directs you to your XBee ZigBee dashboard where you can see the new progress bar widget; for example:



View device data and events in the log file for XBee Gateway

The XBee Gateway application is a Python application that resides on XBee Gateway. Its key functions include connecting your XBee modules to Device Cloud, enabling uploads of data to Device Cloud, and receiving remote text and commands. The XBee Gateway application is installed in your XBee Gateway device and automatically starts when the gateway initializes. When you use the XBee Gateway application, device data for your XBee network is captured in the form of events in one of the XBee Gateway log files, **python.log**. Events that may be of note include:

- Serial or I/O data arriving from an XBee node on your RF network
- An RCI command received from Device Cloud
- Attempts to upload data to Data Streams in Device Cloud
- Errors and warnings during execution for debugging and diagnostics

The **python.log** file may be accessed through the XBee Gateway web interface or from Device Cloud.

See XBee Gateway application and Python on page 1 for more information.

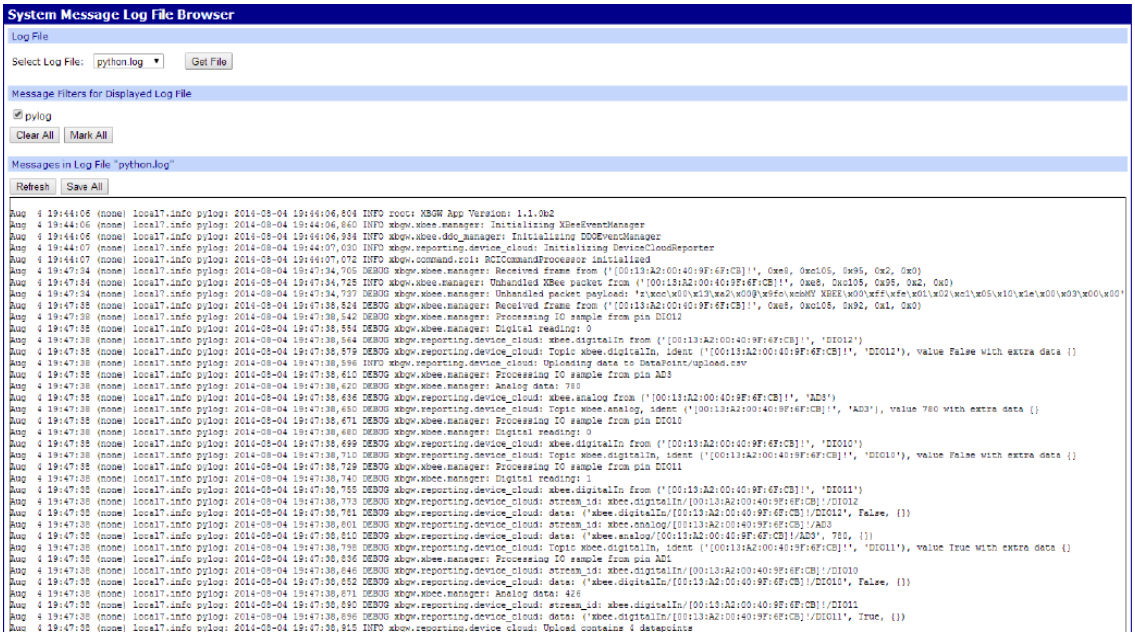
View device data from the XBee Gateway web interface

To access the python.log file from the XBee Gateway web interface:

1. Click **System Log** under Administration.
2. Select **python.log** from the **Select Log File** menu and click **Get File**. The **python.log** file appears. Events in the log file that may be of note include:
    - Serial or I/O data arriving from an XBee node on your RF network
    - An RCI command received from Device Cloud
    - Attempts to upload data to Data Streams in Device Cloud
    - Errors and warnings during execution for debugging and diagnostics

3. The page does not refresh automatically, so when you want to ensure that the data is up to date, click **Refresh**.

   The following example shows the python.log file for an XBee Gateway with an XBee node configured for digital and analog I/O:



   In this log file excerpt:

   - These lines show that the XBee Gateway application has started up successfully:

     ```
     Aug  4 19:44:06 (none) local7.info pylog: 2014-08-04 19:44:06,804
     INFO root: XBGW App Version: 1.1.0b2
     Aug  4 19:44:06 (none) local7.info pylog: 2014-08-04 19:44:06,860
     INFO xbgw.xbee.manager: Initializing XBeeEventManager
     Aug  4 19:44:06 (none) local7.info pylog: 2014-08-04 19:44:06,934
     INFO xbgw.xbee.ddo_manager: Initializing DDOEventManager
     Aug  4 19:44:07 (none) local7.info pylog: 2014-08-04 19:44:07,030
     INFO xbgw.reporting.device_cloud: Initializing DeviceCloudReporter
     Aug  4 19:44:07 (none) local7.info pylog: 2014-08-04 19:44:07,072
     INFO xbgw.command.rci: RCICommandProcessor initialized
     ```

- These lines show a digital I/O reading received by the XBee Gateway application:

```
Aug  4 19:47:38 (none) local7.info pylog: 2014-08-04 19:47:38,524
DEBUG xbgw.xbee.manager: Received frame from ('
[00:13:A2:00:40:9F:6F:CB]!', 0xe8, 0xc105, 0x92, 0x1, 0x0)
Aug  4 19:47:38 (none) local7.info pylog: 2014-08-04 19:47:38,542
DEBUG xbgw.xbee.manager: Processing IO sample from pin DIO12
Aug  4 19:47:38 (none) local7.info pylog: 2014-08-04 19:47:38,554
DEBUG xbgw.xbee.manager: Digital reading: 0
```

- These lines show an analog I/O reading received by the XBee Gateway application:

```
Aug  4 19:47:38 (none) local7.info pylog: 2014-08-04 19:47:38,610
DEBUG xbgw.xbee.manager: Processing IO sample from pin AD3
Aug  4 19:47:38 (none) local7.info pylog: 2014-08-04 19:47:38,620
DEBUG xbgw.xbee.manager: Analog data: 780
```

- These lines show six data points uploaded to Device Cloud:

```
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,207
INFO xbgw.reporting.device_cloud: Uploading data to
DataPoint/upload.csv
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,220
DEBUG xbgw.reporting.device_cloud: stream_id: xbee.analog/
[00:13:A2:00:40:9F:6F:CB]!/AD1
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,230
DEBUG xbgw.reporting.device_cloud: data: ('xbee.analog/
[00:13:A2:00:40:9F:6F:CB]!/AD1', 426, {})
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,238
DEBUG xbgw.reporting.device_cloud: stream_id: xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO4
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,245
DEBUG xbgw.reporting.device_cloud: data: ('xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO4', True, {})
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,263
DEBUG xbgw.reporting.device_cloud: stream_id: xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO6
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,280
DEBUG xbgw.reporting.device_cloud: data: ('xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO6', True, {})
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,288
DEBUG xbgw.reporting.device_cloud: stream_id: xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO7
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,300
DEBUG xbgw.reporting.device_cloud: data: ('xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO7', False, {})
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,308
DEBUG xbgw.reporting.device_cloud: stream_id: xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO0
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,324
DEBUG xbgw.reporting.device_cloud: data: ('xbee.digitalIn/
[00:13:A2:00:40:9F:6F:CB]!/DIO0', True, {})
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,332
DEBUG xbgw.reporting.device_cloud: stream_id: xbee.analog/
[00:13:A2:00:40:9F:6F:CB]!/AD2
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,337
DEBUG xbgw.reporting.device_cloud: data: ('xbee.analog/
[00:13:A2:00:40:9F:6F:CB]!/AD2', 780, {})
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,354
```

```
INFO xbgw.reporting.device_cloud: Upload contains 6 datapoints
Aug  4 19:47:44 (none) local7.info pylog: 2014-08-04 19:47:44,526
INFO xbgw.reporting.device_cloud: Upload successful
```

View device data from Device Cloud

The system log is a file maintained in the file system of the device. To retrieve the python.log file from Device Cloud, use **File Management** capabilities in Device Cloud. See the *Device Cloud User Guide* if you need help.

Device data or third party dashboards

The XBee ZigBee Cloud Kit web application is available for viewing device data on Device Cloud. If you want to create your own dashboard or use a third-party dashboard instead, refer to the *Device Cloud Programming Guide* for a description of the Web Services API.

Next steps

The following topics provide additional information on what you can do after you set up the XBee ZigBee Cloud kit.

Expand your kit

Using examples on the XBee ZigBee Examples and Guides page, you can expand your XBee ZigBee Cloud Kit. This page provides additional tutorials and guides that will teach you more about the full capabilities and features of your XBee ZigBee Cloud Kit. For example, you can create your own temperature sensor using the loose components that came with your kit.

Learn more

See Learn more about the XBee ZigBee Cloud Kit on page 19 for more information related to your XBee ZigBee Cloud Kit, including:

- XBee ZigBee configuration methods
- Connecting your application to the Cloud
- Collaborating with other XBee ZigBee users

See the *XBee Gateway User Guide* for more information on XBee Gateway.

Troubleshooting

See Troubleshoot your XBee ZigBee Cloud Kit on page 24 for helpful topics on resolving any problems you may encounter when working with your XBee ZigBee Cloud Kit.

Learn more about the XBee ZigBee Cloud Kit

In addition to the following topics, you can find information related to the Digi devices and applications used throughout this kit on the XBee Gateway product page.

Create your own applications

Visit https://github.com/digidotcom/XBeeZigBeeCloudKit for the complete source code used to create the sample XBee ZigBee Cloud Kit web application. GitHub is a host site for various software development products, and has many features including discussion tools, access to repositories, and the ability to collaborate with other users. Start building your own applications today.

Connecting your application to Device Cloud

Use one of the following methods to connect to Device Cloud:

- **Data Streams**. Visit the XBee ZigBee Device Cloud Kit application to learn about data streams. You can find the data stream section by navigating to **Data Services** > **Data Files** > **Data Streams**. If you have already completed Getting started with XBee Gateway, there are several data streams already available. The data streams are similar to the following example:

  ```
  00000000-00000000-00409DFF-FF123456/xbee.digitalIn/
  [00:13:A2:00:11:22:33:44]!/DIO4
  ```

  You can find all data streams for your XBee Gateway by searching for the device ID for your XBee Gateway; for example:

  ```
  00000000-00000000-00409DFF-FF123456
  ```

  Type the device ID for XBee Gateway in the data stream search box in the upper-right side of the data stream view.

- **Monitoring data streams with a TCP or HTTP monitor**. You can find more information about monitoring data streams by reading documentation about monitors. You also have access to various client libraries for the TCP Device Cloud monitors made available from the Digi International github account. The links to these libraries are located below:
  - Device Cloud Push Monitor API for Python
  - Device Cloud Monitor API Library for Java
- **Web APIs**. Log in to Device Cloud and use the API Explorer to see the available web service APIs.

Collaborate and share your story

Tell us what you have made. Share your story by submitting your own XBee project or other projects that you find interesting here at: http://gallery.digi.com/submitproject/.

Additional services

For information on where to obtain more of our XBee products, professional services, and access to the Digi online support forums, see the links below:

- To purchase more XBee ZigBee modules, go to the XBee and ZigBee product page.
- If you need help building your own application, go to www.digi.com, select **Products & Services** > **Services**.
- When you are ready to upgrade your account, contact a Digi expert at www.digi.com/support.
- If you have any questions, visit the Digi Forum.

Use the XBee ZigBee Cloud Kit web application

You can expand your XBee ZigBee Cloud Kit to create custom connected device applications. To do so, you will use some of the loose components that came with your kit to create a simple application that interacts with your online dashboard. This example is available on the XBee ZigBee Examples and Guides page.

Note To complete the exercises described on the XBee ZigBee Examples and Guides page, you must have a thorough understanding of your XBee ZigBee Cloud Kit and have completed the steps to set up your kit.

Additional Digi XBee examples

Digi provides a variety of examples of XBee modules in various applications on the Digi XBee Examples and Guides page.

Widget settings

To configure the widgets on your dashboard, hover over the widget header, and click the settings icon that appears in the upper-right corner of your device's input output (I/O) channel icon. On the widget settings page, you can edit and customize the display of each widget. The following table briefly describes each widget setting and how it affects the widget.

Widget Setting	Description
Input Stream	Specifies the data stream that the widget subscribes to for data updates. Can be Digital (**DIO#**) or Analog (**AD#**). (The tilt indicator widget has **X-Axis Stream** and **Y-Axis Stream** settings; these are two copies of the **Input Stream** setting, renamed to clarify the use of the stream.)
Output Stream	Specifies the digital I/O pin (**DIO#**) on the XBee module to which the widget sends its output values.
Units	String label used in the widget to display the units for the values (for example, millivolts, decibels).
Timespan	The total timespan, in seconds, to display in the graph. For example, if the timespan is set to 90, the graph will show 90 seconds of data at a time.
X-axis tick size	The spacing, in seconds, between tick marks/lines drawn along the X-axis of the graph
Y-axis autoscale	When you select this check box, the Y-axis of the graph automatically scales to fit the data displayed. If you clear the check box, the minimum and maximum values default to the Y-axis min/max values.
Y-axis minimum/maximum	The minimum and maximum range on the graph's Y axis. You can use this setting if the Y-axis auto scale check box is cleared.
Low value/High value	This setting is available on the progress bar and gauge widgets. It specifies the low and high values for data within the progress bar or gauge. For example, you can set these values to display a data stream’s values ranging from 100 to 1000.

Widget Setting	Description
Input Transform	Allows incoming data to be processed with an expression for display. The transform field is implemented using the AngularJS **$eval** function, and will only evaluate basic Angular expressions. For example, the stock potentiometer gauge widget uses **value/1023 * 100**. The following variables can access the input value: • **value** • **val** • **input** • **x** See Angular Expressions vs. JS Expressions for more information on Angular JS.
Invert Values	When you select this check box, setting the switch to the **On** position sends the value **0**. Setting the switch to the **Off** position sends the value **1**. If you clear this check box, setting the switch to the **On** position sends **1** and setting the switch to the **Off** position sends **0**.
Read Only Switch	When you select this check box, the switch represents the data stream value last received from the server.
Add Carriage Returns	This setting is available on the serial data widget. Specifies whether to wrap serial data sent to the XBee module in carriage return characters first. This use of a carriage returns helps make the text more readable when printed out, displayed in the widget, or XCTU.
Check Radio Configuration button	Verifies that the XBee module's I/O configuration is appropriate for the selected input or output stream. (For example, if the Input Stream is set to DIO0, the **Check Radio Configuration** dialog verifies that D0 on the XBee module is set to act as a digital input.) Click this button to ensure that your XBee module is configured to work with your XBee ZigBee Cloud Kit.

Widget Setting	Description
Discover control (near XBee Module)	Provides available options for refreshing the list of XBee modules. The options include: • **Fetch from Device Cloud**: Gets a list of XBee modules using the Device Cloud XbeeCore web service. • **Fetch from gateway (cached)**: Gets a list of XBee modules from the cached discovery list on XBee Gateway. • **Fetch from gateway (clear device list)**: Clears the cached discovery list on XBee Gateway, performs a new discovery operation, and displays a new list of discovered XBee modules.

View widget source code

Within the XBee ZigBee Cloud Kit web application, you have the opportunity to view Javascript, CSS, and HTML code that generates your widgets. To view the code, hover over the widget header, and click on the view widget code </> icon located in the upper-right corner of your widget.

Advanced users working with the XBee ZigBee Cloud Kit can use the widget source code to create their own widgets and write their own applications. See Learn more about the XBee ZigBee Cloud Kit on page 19 for more information about widget source code as a starting point for creating your own custom applications.

Troubleshoot your XBee ZigBee Cloud Kit

This section covers common issues and troubleshooting information for your XBee ZigBee Cloud Kit.

Common issues

Kit Component	Symptom	Potential Cause	Resolution
XBee Gateway	XBee Gateway fails to connect to and communicate with Device Cloud.	XBee Gateway is not configured properly.	Open the XBee Gateway web interface. Go to **Configuration** > **Device Cloud Connectivity**. Check the **Device Cloud Server Connection**. Make sure it is pointing to the proper Device Cloud server.
	XBee Gateway fails to join the wireless network.		Make sure the wireless network you are trying to connect to is working. Rerun the Wireless wizard.

Kit Component	Symptom	Potential Cause	Resolution
The XBee module on the development board	The XBee module does not join the network started by XBee Gateway.	XBee Gateway and XBee module have incompatible PAN ID and security settings.	Check the **Extended PAN ID** and security settings and make sure they match those set for XBee Gateway. See Join your XBee node to your XBee Gateway on page 35 for instructions.
		The XBee module is not powered on.	Apply power to the development board.
		The XBee module is out of range.	Move the XBee module closer to XBee Gateway. If the XBee module has an external antenna, make sure it is securely attached to the module.
		The XBee module is sleeping.	Press the **Commissioning** button on the development board once. The XBee module wakes and joins a network, if it is not currently joined to a network.

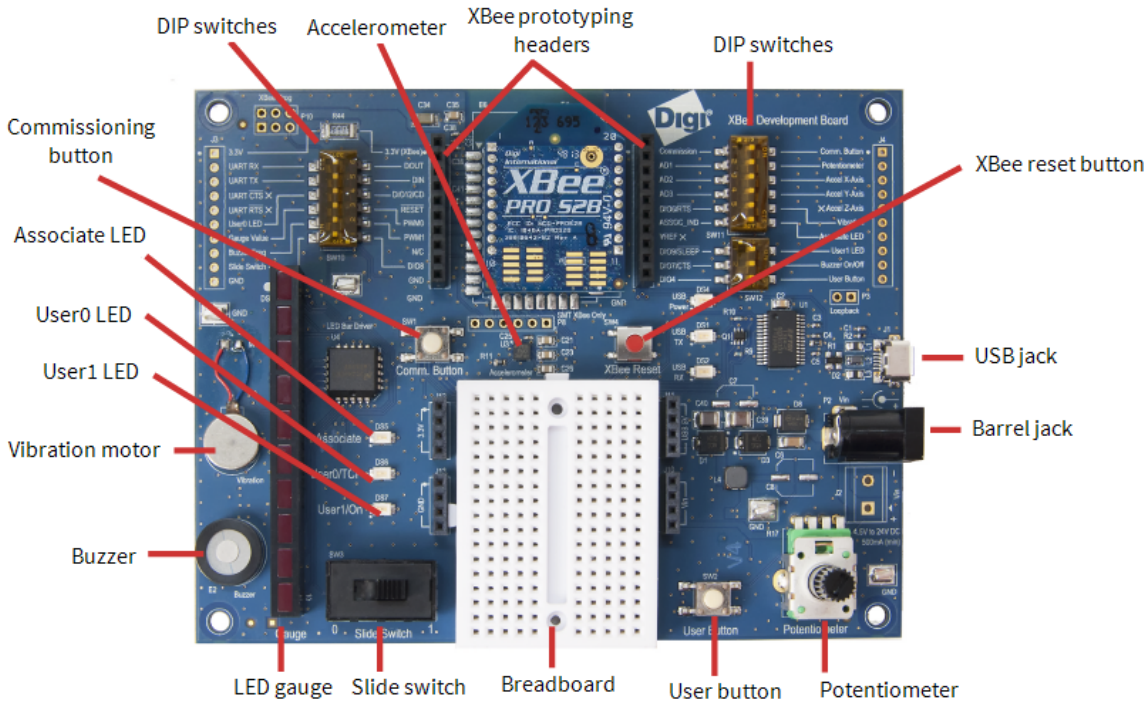
Kit Component	Symptom	Potential Cause	Resolution
		Your device may be listed under another Device Cloud account. By default, an XBee ZigBee device can only be associated with one Device Cloud account at a time. You need to remove the device from the Device Cloud account under which it is currently listed.	1. Depending on the type of Device Cloud account you have, go to the Device Cloud login. 2. Type your user name and password and click **Log in**. 3. Under the **Device Management** tab, right-click the device you want to delete and select **Remove Devices** from the list.

Kit Component	Symptom	Potential Cause	Resolution
The XBee ZigBee Cloud Kit web application dashboard	The dashboard does not work with the XBee device.	The XBee device may not be properly connected to the XBee network.	Make sure the Associate LED is flashing. Refresh the web browser.
		The development board may not be properly set for operation with an XBee ZigBee module.	See XBee ZigBee/Wi-Fi-DIP-switches on the XBee development board on page 35.
	Changes made from the XBee ZigBee development board are not reflected on the dashboard.	Sampling rate and throttling of data on Device Cloud.	1. Make sure your XBee ZigBee is properly connected, and wait at least 10 seconds for the application to update. 2. Refresh your browser and try again.
		The connection between Device Cloud and the XBee ZigBee Cloud Kit web application may have been disconnected.	Refresh your web browser to reload the dashboard page. If the problem persists, the dashboard may be experiencing issues setting up an HTTP push monitor in Device Cloud. In this case, an error appears in the dashboard, including the specific reason for the error. For example, there are too many push monitors set up for your Device Cloud account.

Kit Component	Symptom	Potential Cause	Resolution
	Certain dashboard functions are not working properly.	Web browser incompatibility.	Try using a different web browser. The recommended web browsers for the XBee ZigBee Cloud Kit web application include Firefox and Google Chrome.
		The development board may not be properly set for operation with an XBee ZigBee module.	See XBee ZigBee/Wi-Fi-DIP-switches on the XBee development board on page 35.

Kit Component	Symptom	Potential Cause	Resolution
The data from the XBee nodes	Data is not appearing in the XBee ZigBee Cloud Kit dashboard or Device Cloud.	The XBee Gateway application is not running, or not loaded, on XBee Gateway.	1. Check the Python Auto Start Settings (**Configuration** > **Python**). The **xbgw_main.py** application should have **Autostart** enabled. If not, enable **Autostart** and click **Apply**. 2. If Autostart for **xbgw_main.py** application is enabled, check the log file, **python.log**, on XBee Gateway for any errors related to the application. Use the File Management function in the XBee Gateway web interface or Device Cloud to determine whether the following files are loaded on XBee Gateway: • **xbgw_main.py** • **build.py** • **xbgw.zip** • **xbgw_ settings.json** If these files are not loaded, use the file management function to load them. For more information on File Management, see the *XBee Gateway User Guide*.

XBee development board components and descriptions



Associate LED

The Associate pin (pin 15) provides an indication of the device's sleep status and diagnostic information.

To enable the associate LED functionality, set the **D5** command to 1; it is enabled by default. If enabled, the Associate pin is configured as an output. This section describes how the pin behaves.

The Associate pin indicates the synchronization status of a sleep compatible XBee Gateway. If a device is not sleep compatible, the pin functions as a power indicator.

Use the **LT** command to override the blink rate of the Associate pin. If you set **LT** to 0, the device uses the default blink time: 500 ms for a sleep coordinator, 250 ms otherwise.

The following table describes the Associate LED functionality.

Sleep mode	LED Status	Meaning
0	On, blinking	The device has power and is operating properly.

Sleep mode	LED Status	Meaning
1, 4, 5	Off	The device is in a low power mode.
1, 4, 5	On, blinking	The device has power, is awake and is operating properly.
7	On, solid	The network is asleep, or the device has not synchronized with the network, or has lost synchronization with the network.
7, 8	On, slow blinking (500 ms blink time)	The device is acting as the network sleep coordinator and is operating properly.
7, 8	On, fast blinking (250 ms blink time)	The device is properly synchronized with the network.
8	Off	The device is in a low power mode.
8	On, solid	The device has not synchronized or has lost synchronization with the network.

Diagnostics support

The Associate pin works with the Commissioning button to provide additional diagnostic behaviors to aid in deploying and testing a network. If you press the Commissioning button once, XBee Gateway transmits a broadcast Node Identification Indicator (0x95) frame at the beginning of the next wake cycle if the device is sleep compatible, or immediately if the device is not sleep compatible. If you enable the Associate LED functionality using the **D5** command, a device that receives this transmission blinks its Associate pin rapidly for one second.

The Commissioning button

The XBee ZigBee development board supports a set of commissioning and LED functions to help you deploy and commission devices. These functions include the Commissioning button definitions and the associated LED functions.

Definitions

To enable the Commissioning button functionality on pin 20, set the **D0** command to 1. The functionality is enabled by default.

You must perform multiple button presses within two seconds.

The following table provides the button definitions.

Button presses	Sleep configuration and sync status	Action
1	Not configured for sleep	Immediately sends a Node Identification broadcast transmission. All devices that receive this transmission blink their Associate LED rapidly for one second. All devices in API operating mode that receive this transmission send a Node Identification Indicator frame (0x95) out their UART.
1	Configured for asynchronous sleep	Wakes the device for 30 seconds. Immediately sends a Node Identification broadcast transmission. All devices that receive this transmission blink their Associate LED rapidly for one second. All devices in API operating mode that receive this transmission send a Node Identification Indicator frame (0x95) out their UART.
1	Configured for synchronous sleep	Wakes the module for 30 seconds or until the synchronized network goes to sleep. Queues a Node Identification broadcast transmission that it sends at the beginning of the next network wake cycle. All devices that receive this transmission blink their Associate LED rapidly for one second. All devices in API operating mode that receive this transmission send a Node Identification Indicator frame (0x95) out their UART.
2	Not configured for synchronous sleep	No effect.
2	Configured for synchronous sleep	Causes a node configured with sleeping router nomination enabled to immediately nominate itself as the network sleep coordinator. For more information, see the SO (Sleep Options) command for your XBee module.
4	Any	Sends an **RE** command to restore device parameters to default values.

Use the Commissioning button

Use the CB command to simulate button presses in software. Send **CB** with a parameter set to the number of button presses to perform. For example, if you send **ATCB1**, the device performs the action(s) associated with a single button press.

The Node Identification Indicator (0x95) frame is similar to the Remote Command Response (0x97) frame—it contains the device's address, node identifier string (NI command), and other relevant data. All devices in API operating mode that receive the Node Identification Indicator frame send it out their UART as a Node Identification Indicator frame.

If you enable the Commissioning button during sleep, it increases the sleeping current draw, especially in Asynchronous pin sleep (**SM** = 1) mode. When asleep, hold down the Commissioning button for up to two seconds to wake the device from sleep, then issue the two or four button presses.

User0/TCP LED

You can control the User0/TCP LED through the XBee ZigBee dashboard.

User1/On LED

You can control the User1/On LED through the XBee ZigBee dashboard.

LED gauge

Controlled through the XBee ZigBee dashboard, the LED gauge can be toggled on or off.

Vibration motor

The vibration motor indicates a change in status. You can control the vibration motor through the XBee ZigBee dashboard.

Accelerometer

The accelerometer drives the Tilt widget. You can control the accelerometer through the XBee ZigBee dashboard.

Buzzer

You can use the buzzer as an alarm to indicate trouble. You can control the buzzer through the XBee ZigBee dashboard. Use the buzzer toggle widget to turn the buzzer on/off.

Slide switch

The state of the slide switch is indicated on the XBee ZigBee dashboard. Similar to a light switch, it stays either on or off.

Push button

The state of the push button appears on the XBee ZigBee dashboard. The push button only displays ON when you are actively pushing the button.

Potentiometer

You can choose to report the data from the potentiometer to the XBee ZigBee dashboard as a speedometer gauge, progress bar, or a graph. You can control the potentiometer by turning the adjustable knob left or right.

Barrel jack

Accepts 4.5 V to 24 V. You can use the barrel jack with the battery pack provided with the kit.

USB jack

You can use the USB jack to power the development board. The USB jack also provides serial port access to the XBee ZB for configuration or sending and receiving data to and from Device Cloud.

XBee reset button

The XBee reset button is connected to the reset pin of the XBee ZigBee.

XBee prototyping headers

Allows connection to all pins of the XBee ZigBee. Use the prototyping headers to connect to your own circuits including those using the loose components that came with your kit.

DIP switches (top of board)

Allows components on the development board to be disconnected from the XBee ZigBee module. If the switch is away from the XBee ZigBee module, the XBee pin is connected to that built-in widget on the XBee ZigBee dashboard. If the switch is towards the XBee, the XBee pin is disconnected from that component.

Note When you use loose components or do advanced prototyping, you may need to disconnect the built-in component from the XBee pin.

DIP switches (back or underside of board)

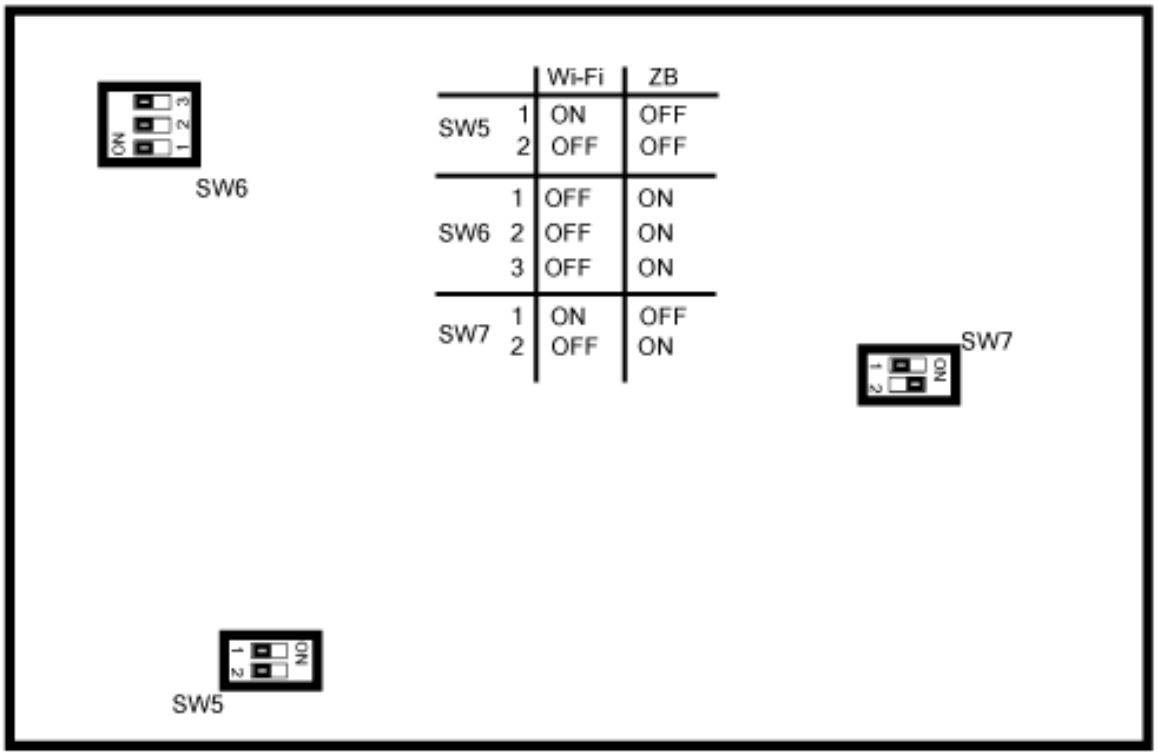
Set the board for use with XBee ZigBee or XBee Wi-Fi modules. For more information, see XBee ZigBee/Wi-Fi-DIP-switches on the XBee development board on page 35.

XBee ZigBee/Wi-Fi-DIP-switches on the XBee development board

The XBee development board in the XBee ZigBee Cloud Kit has three DIP switches on the back of the board. These switches set the board for use with XBee ZigBee or XBee Wi-Fi modules. A table on the board shows the proper switch settings for XBee ZigBee and XBee Wi-Fi modules and is shown in the following figure. These switches are set at the factory for the XBee ZigBee Cloud Kit. In the event the switches are changed away from the proper settings for XBee ZigBee modules, you may experience the following symptoms when working with the development board and XBee ZigBee Cloud Kit web application dashboard:

- The buzzer will not work as expected.
- None of the analog components will work or display properly on the dashboard.

The following figure shows the proper DIP switch settings for the XBee ZigBee Cloud Kit for use with XBee ZigBee modules:



Join your XBee node to your XBee Gateway

If you do not see your XBee node in the list of devices in the XBee ZigBee Cloud Kit web application, your XBee node is not joined to the Gateway network started by XBee Gateway. Use the following sections to join your node to your XBee Gateway.

Creating a ZigBee network

To verify that your XBee ZigBee module has successfully joined the ZigBee network started by the XBee Gateway device:

1. Make sure XBee Gateway is powered on.
2. Make sure the XBee ZigBee node is mounted on the development board and that the development board is powered on.
3. Open a web browser and type the URL of XBee Gateway: **http://192.168.100.1**. This will open the XBee Gateway web interface.
4. Under Configuration, click **XBee Network**. The XBee Configuration page appears.
5. Under the **Remote XBee Devices** table, clear the **Clear list before discovery** check box and then click the **Discover XBee Devices** button. After a few seconds, the table displays your XBee ZigBee node.

Get the PAN ID and other network parameters for the ZigBee network

Use the XBee ZigBee Cloud Kit web application to retrieve the PAN ID of the ZigBee network that you want to join.

Discover XBee radios

You must install XCTU on your computer. If you do not have XCTU installed, go to http://www.digi.com/xctu to download and install XCTU.

To discover the XBee radios attached to your computer:

1. Connect the XBee development board to your computer using a USB cable.
2. Open XCTU and add the XBee node that is connected to your computer to the list of radio modules.
3. Click the **Discover devices**  . The **Discover radio devices** dialog appears.
4. Select the serial port(s) to scan from the list. If you know the serial port you can save time by selecting just that port. The discovery process attempts to identify radios on all selected ports.
5. Click **Next**.

6. Configure serial settings. The defaults are good for most radios and do not need to be changed in most environments.
7. Click **Finish**. The system performs the discovery process. This can take some time. All discovered devices appear upon completion.
8. Choose your radio(s) from the list.
9. Click **Add selected devices**.

The radios that you discovered now appear in the **Radio Modules** pane of the main XCTU application.

Read XBee configuration

To read the configured parameters of your XBee radio:

1. In XCTU, click the settings icon to ensure the XCTU is in Configuration mode.
2. In the **Radio Modules** pane, click the entry for your radio. A dialog box displays progress as the configuration is read from the radio.

Configure the PAN ID and additional network parameters

You will need the PAN ID of the network you want to join before you start this task.

To set the PAN ID parameter for the network you want to join:

1. Type your PAN ID. Replace the value currently displayed in the PAN ID field of the Networking section of the **Radio Configuration** pane with your PAN ID.
2. Click the write icon in the PAN ID row to send your changes to the radio. The radio leaves its current network and searches for the network you have just configured.

3. If XBee Gateway uses XBee security parameters, set the XBee security parameters (**EE**, **EO**, **NK**, **KY**) as needed. The security parameters are as follows:

    - **EE**: Enable or disable security in the network.
    - **EO**: Set the security policy for the network.
    - **NK**: Set the network security key for the network. If set to 0 (default), the device will use a random network security key.
    - **KY**: Set the trust center link key for the network. If set to 0 (default), the device will use a random network security key.

    For more information on these parameters, see the *XBee/XBee-PRO ZigBee RF Modules User Guide* (Digi part number 90000976). This document is available from the XBee/XBee-PRO ZB (S2C) Modules page.

Configuring ZigBee nodes with custom PAN ID

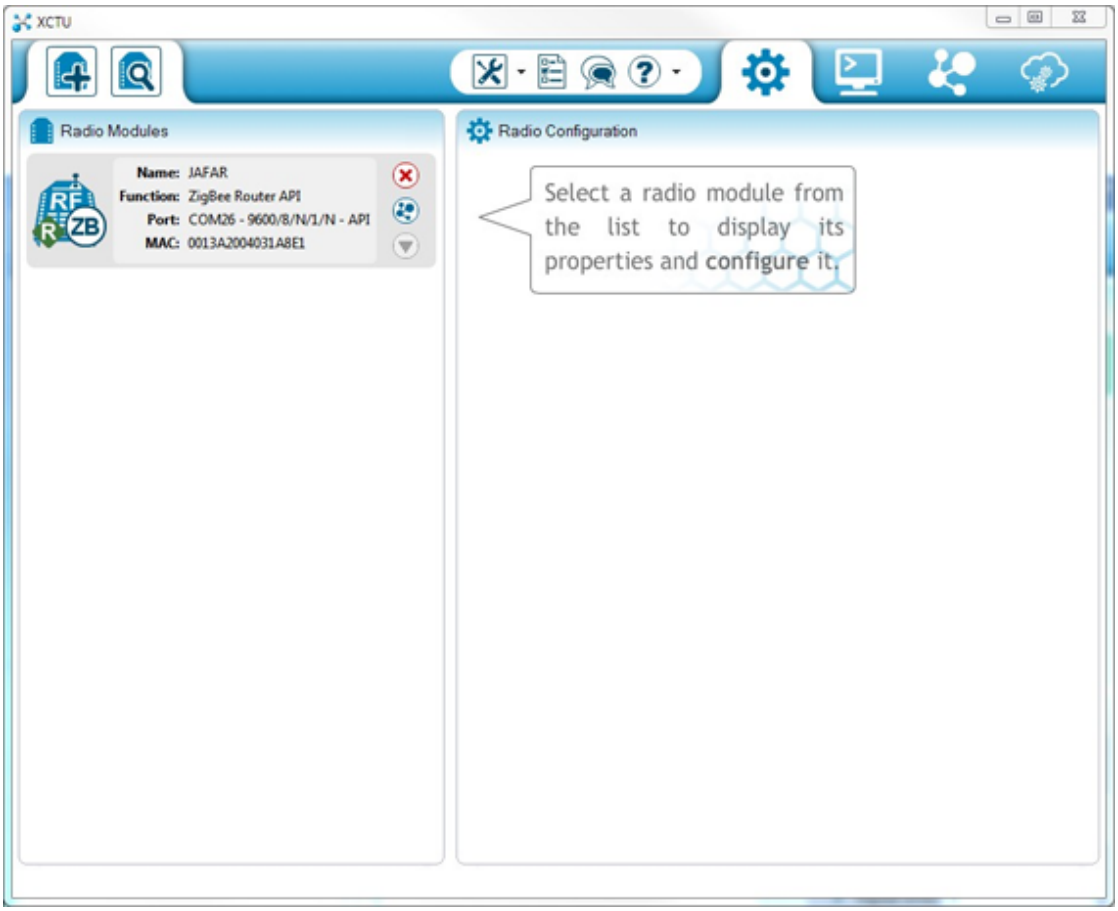
XBee ZigBee Cloud Kit includes an XBee ZigBee module configured as router node. If you want your ZigBee routers or end devices to connect to a ZigBee network with a specific PAN ID, you must configure the ZigBee routers to use the specific PAN ID for the ZigBee network.

You must install XCTU on your computer. If you do not have XCTU installed, go to http://www.digi.com/xctu to download and install XCTU.

To configure other ZigBee nodes with a custom PAN ID:

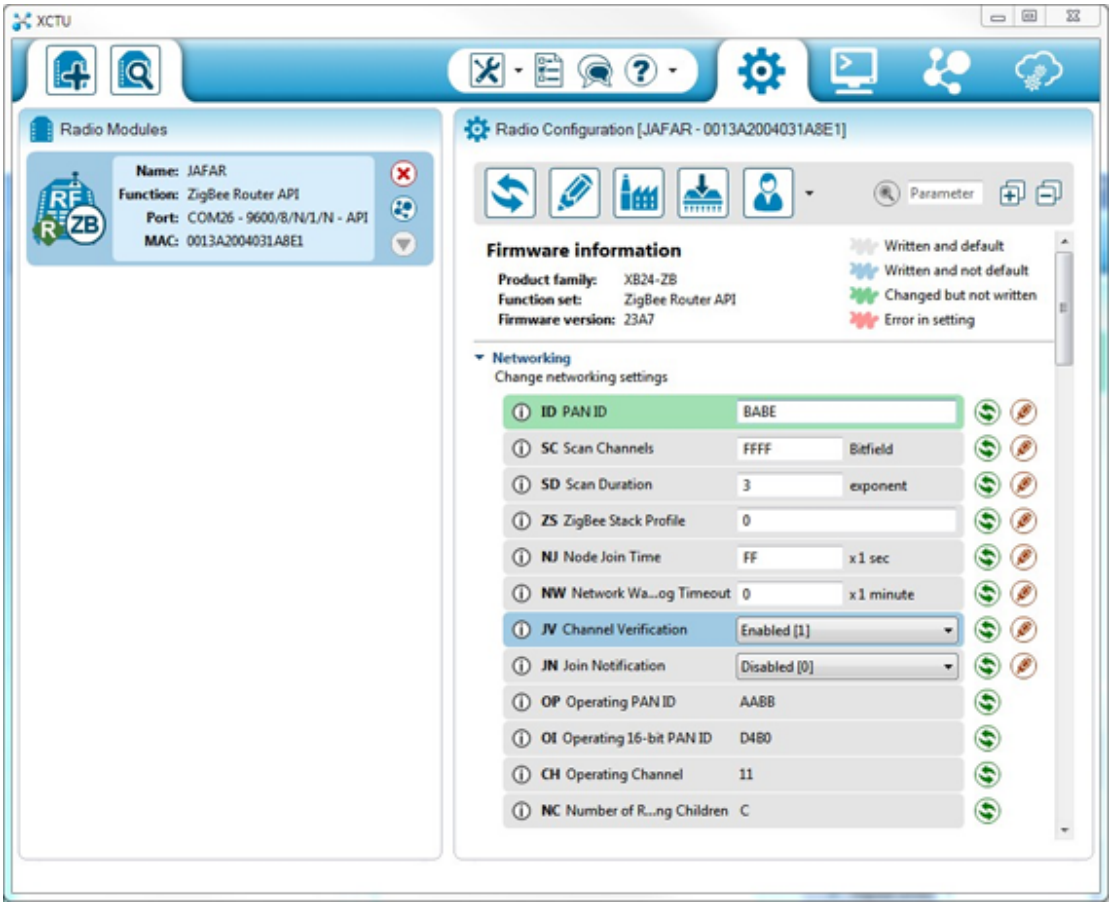
1. Attach the XBee ZigBee node to the XBee development board included in the XBee ZigBee Cloud Kit, and connect it to your computer using a USB cable.

2. Open XCTU and add the XBee node that is connected to your computer to the list of radio modules.



3. Select the node and wait for the application to read all its settings.

4. Under the **Networking** category, replace the current PAN ID with your custom PAN ID in the **ID PAN ID** field.



5. Click the **Write radio settings** button to save the new PAN ID in the XBee module

As soon as you save the new PAN ID in the XBee RF module, the module tries to connect to the ZigBee network that has the PAN ID that you configured.

Verify your radio has joined the correct network

To verify that your radio has joined the correct network:

1. Scroll down the **Radio Configuration** pane until the **Diagnostic Command** section is visible.

2. Click the refresh button in the **Association Indication** row to read the current association status. The Association Indication value, as the name implies, provides information about whether the radio is currently associated with a network or an error value. The value zero (0) indicates that the radio has successfully associated with the network started by XBee Gateway. It may take several seconds for association to occur. Refresh multiple times "if necessary" to get ongoing status. You can find information on error values in the Module Manual for your radio module.