



# Query Zigbee Node Settings with Digi Remote Manager®

---

November 2017

90002266

## Contents

Overview .....	3
Query Zigbee node settings using the Remote Manager UI .....	3
Query Zigbee node settings programatically .....	4
Example: Send the request .....	4
Example: Response to the request.....	5
Where to find more examples .....	5

### Overview

Within Digi Remote Manager®, you can interact with Zigbee nodes, IP gateways, and routers. Interacting with Zigbee nodes is supported natively with Digi gateways and XBee devices.

For other Zigbee devices, you can gain limited information based on common Zigbee parameters. However, you can use a Python program on the IP gateway. This helps you to achieve a similar level of functionality on a Zigbee device that you have on an XBee node, as long as the Python program understands the proprietary communications protocols for non-Digi nodes.

**NOTE:** Information about the proprietary communications protocols for non-Digi nodes is outside the scope of this document.

In this topic, you will learn how to get Zigbee node setting information from Remote Manager.

### Query Zigbee node settings using the Remote Manager UI

You can view the Zigbee node settings in Remote Manager.

1. Log into [Remote Manager](#).
2. Click **Device Management > XBee Networks**.
3. Search for the desired device. You can filter your search by entering a Zigbee node ID or address in the search field.
4. Double-click the desired device. The **Properties** page appears.
5. Click either the **Basic** or **Advanced** menu item to display the most recent cached settings in the right-hand pane.
6. Click **Refresh** to update the settings. This action sends a request to the gateway to read the current settings and the state of the node, and then sends a response back to Remote Manager. See the following graphic.

## Query Zigbee node settings

The screenshot shows the 'Advanced' settings for a Zigbee node. The parameters and their values are as follows:

Parameter	Value
(NO) Node discovery options:	0x0
(EE) Encryption enable:	0
(EO) Encryption options:	0x0
(ID) Extended PAN identifier:	0x0013a20040abe511
(GT) Guard times:	1,000 msec
(II) Initial PAN identifier:	0x00000000
(JN) Join notification:	0
(JV) Join verification:	0
(KY) Link encryption key:	
(NH) Maximum hops:	30
(NK) Network encryption key:	
(NW) Network watchdog timeout:	0 min
(RO) Packetization timeout:	3 chars
(PO) Polling rate:	0 x 10 msec
(PM) Power mode:	1
(PD) Pull-up/down direction:	0x00000000

## Query Zigbee node settings programatically

Querying the Zigbee node settings follows the standard process for querying any device settings in Remote Manager. However, when you query the Zigbee node settings, you instruct the gateway to send requests out to the gateway's local PAN (Personal Area Network).

### Example: Send the request

This example uses a specific RCI command within the SCI interface. You will send an HTTP POST to <https://remotemanager.digi.com/ws/sci> and include the code shown below as the data content of the request.

```
<sci_request version="1.0">
  <send_message cache="false">
    <targets>
      <device id="00000000-00000000-00409DFF-FF5C525A"/>
    </targets>
    <rci_request version="1.1">
      <do_command target="zigbee">
        <query_setting addr="00:13:A2:00:40:48:59:BD!"/>
      </do_command>
    </rci_request>
  </send_message>
</sci_request>
```

Note the following:

1. The **cache="false"** parameter is part of the send\_message operator. This instructs Remote Manager to send the request to the gateway and bypass any server-side cache that may exist for this node.

2. The **targets** group contains the device ID of the gateway that has the node of interest associated with its PAN.
3. The **query\_state** command under the RCI do\_command specifies the specific EUI-64 address of the node that is being queried.

### Example: Response to the request

In this example, the response contains the same information that is available in Remote Manager.

```
<sci_reply version="1.0">
  <send_message>
    <device id="00000000-00000000-00409DFF-FF5C525A">
      <rci_reply version="1.1">
        <do_command target="zigbee">
          <query_state addr="00:13:A2:00:40:48:59:BD!">
            <radio>
              <pan_id>0x9dda</pan_id>
              <ext_pan_id>0x0013a20040abe511</ext_pan_id>
              <channel>0xc</channel>
              <net_addr>0x6302</net_addr>
              <association>0x0</association>
              <firmware_version>0x2241</firmware_version>
              <hardware_version>0x1a43</hardware_version>
              <device_type>0x3000f</device_type>
              <children>12</children>
              <max_payload>84</max_payload>
              <rssi>60</rssi>
              <supply_voltage>3211</supply_voltage>
            </radio>
          </query_state>
        </do_command>
      </rci_reply>
    </device>
  </send_message>
</sci_reply>
```

## Where to find more examples

You can view more examples and experiment with the web services APIs in Remote Manager. To access this information, you can log in to Remote Manager and click **Documentation > API Explorer > Examples > SCI > RCI > XBee > Query node state**, as shown in the following graphic.

The screenshot shows the DIGI Remote Manager interface. At the top, the logo 'DIGI REMOTE MANAGER' is visible, along with navigation links for 'Dashboard' and 'Device Mana'. Below this is a dark header with 'Resources' and 'API Explorer'. The main area contains a search bar with 'Path: /ws/sci' and 'HTTP Method: GET'. A dropdown menu is open, listing various services like 'Alarm', 'Carrier', 'DeviceCore', etc. The 'Xbee' service is selected, and a sub-menu is open showing options like 'Descriptors', 'DIA', 'File System', 'Query device settings', 'Query device state', 'Set device setting', and 'XBee'. The 'Query node state' option is highlighted with a blue box. On the right, a 'Web Services Responses' table shows several 'POST /ws/sci 200' entries. Below that, a 'Documentation' section includes a link to 'Programming Guide' and a 'Discover nodes' button.