

# Using X-CTU to Download Firmware and Profiles to XBee RF Modules

## Introduction

X-CTU is a graphical Windows-based serial utility provided by Digi to program and configure the firmware on its RF products, including the XBee ZB and XBee ZNet RF modules and the Digi XBee USB. X-CTU is designed to function with all Windows-based computers running Windows 98 SE and later.

This technical note focuses on the PC settings and modem configuration used to set up and change profiles on XBee RF modules used with Rabbit-based products. Additional information on other aspects of X-CTU is available in Digi's online *X-CTU Configuration and Test Utility Software User's Guide*.

## Using X-CTU

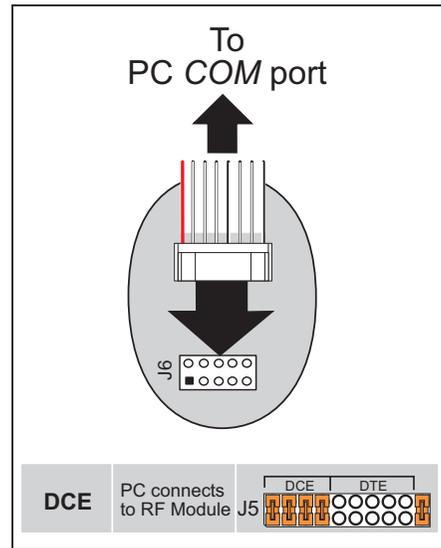
### Install X-CTU

1. Locate and double-click the X-CTU setup executable (**Setup\_XCTU\_5222.exe** in the Dynamic C **DCRabbit... \Utilities \X-CTU** directory) to install the X-CTU application that you will use to download profiles or firmware. You may use a different version of X-CTU from another source, but its behavior may be different.
2. Start X-CTU from the desktop icon.
3. You will be prompted to check Digi's Web site for firmware updates —select no since the correct firmware is already included.

You may need to select yes to get the 2x41 firmware from Digi's Web site if you are using a different version of X-CTU. The “*Firmware Download Options*” section in this Technical Note describes how to install the firmware without an Internet connection.

## Connect Download Cable — XBee RF Module

1. Set the jumpers on header J5 on the RF Interface Board for the DCE configuration to load or modify profiles on the XBee RF modules. Remember to also install the jumper to enable RS-232.
2. Connect the 10-pin connector of the download cable to header J6 on the RF Interface Board. Line up the colored edge of the download cable with pin 1 of header J6 as shown. Connect the DB9 connector end of the download cable to a COM (serial) port on the PC. You may use an RS-232/USB converter cable (Rabbit Part No. 20-151-0178) if your computer does not have a COM port.



## Connect Digi XBee USB

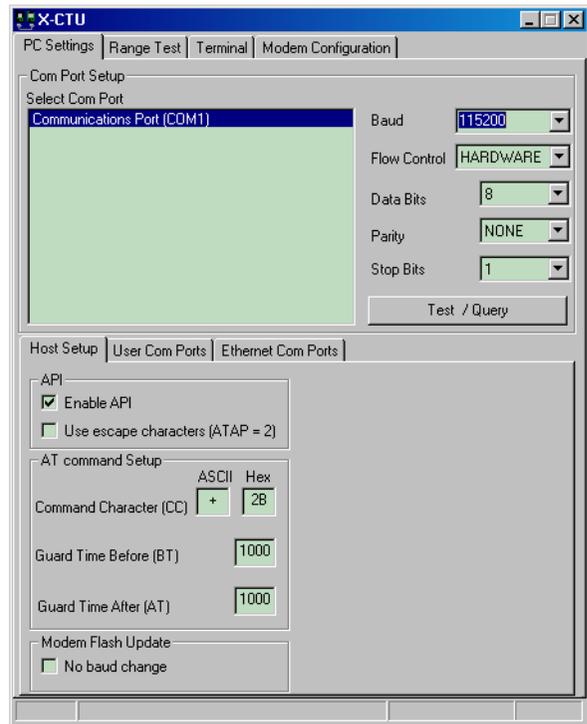
If you are planning to configure the Digi XBee USB, simply connect the USB connector to a USB port on your PC.

## PC Settings Tab

1. Set the “PC Settings” tab to **115200** baud, **HARDWARE** flow control, **8** data bits, parity **NONE**, **1** stop bit.

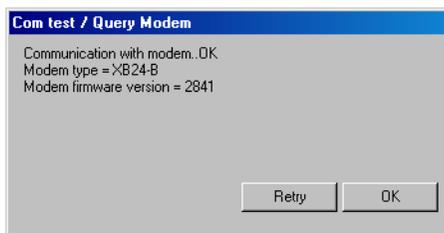
This is the usual setup for XBee RF modules that have been configured for use with Rabbit products. Set the “PC Settings” tab to **9600** baud for XBee RF modules that were obtained from sources other than Rabbit.

2. Check “Enable API” under “Host Setup.” This box must be checked unless you reconfigure the XBee RF module with AT firmware.
3. Select the COM port the download cable is connected to, then click **Test/Query**. A window should open to display the firmware version. Note that a failed **Test/Query** is indicated by no entry after the “=” sign when “Enable API” under “Host Setup” is checked, otherwise a clear failure message is displayed. You may have to retry several times if your XBee RF module is currently configured as an end device and has the sleep mode enabled. Click **OK** in this window to close it.

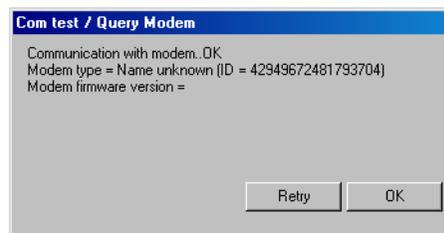


Samples of successful and failed **Test/Query** results for API firmware are shown below.

### Successful Test/Query



### Failed Test/Query



Some displays of a successful **Test/Query** may show **XB24-B** to reflect ZNet firmware instead of **XB24-ZB** to reflect the ZB firmware that is actually on the XBee ZB RF module. This reflects a bug in the X-CTU utility.

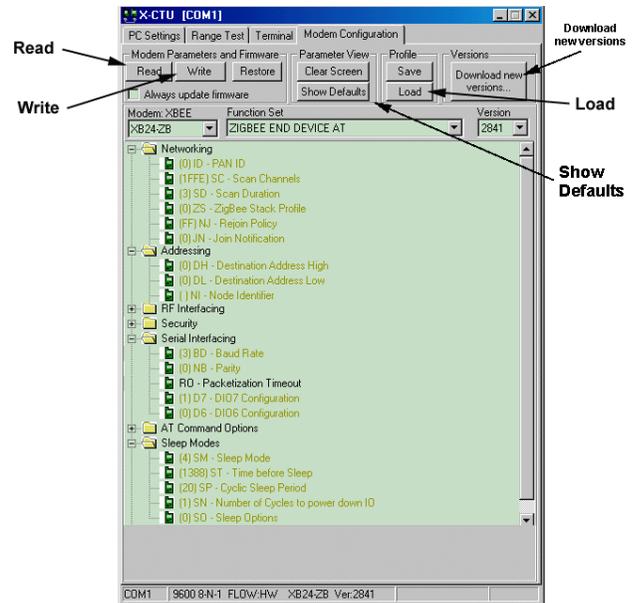
If the **Test/Query** fails, try unchecking or checking “Enable API” under “Host Setup.” Also try changing the baud rate to **9600** or **115200** baud. These are the most common settings you will use for XBee RF modules that are used with Rabbit products.

## Modem Configuration Tab

Profiles, firmware, and configurations for the XBee RF module are set using the “Modem Configuration” tab.

### Load Existing Profile

1. To reinstall a default profile, click the **Load** button under “Profile” and browse to the Dynamic C `DCRabbit... \Utilities \X-CTU` directory to select the desired profile. If your version of Dynamic C does not have any profiles, they are available on our Web site at [www.rabbit.com/support/downloads/](http://www.rabbit.com/support/downloads/).
2. Click **Open** after you select the profile. Click **Write**.



You may get an error message after you click **Write**, particularly if the existing profile had different firmware (for example, if the XBee RF module was set up as a router and you loaded an end-device profile). Return to the “PC Settings” tab to set the baud rate to **9600** baud, check or uncheck “Enable API” depending on whether the new profile is API or AT, then click **Write** on the “Modem Configuration” tab. You will have to return to the “PC Settings” tab and set the baud rate according to the new profile to continue using X-CTU for any additional configuration.

The “*Customizing Configuration*” section explains how to customize the XBee RF module’s configuration, for example, by assigning a name.

## Firmware Download Options

1. Only the firmware supplied on the Dynamic C CD has been tested with Dynamic C and the sample programs. Later versions of the firmware, which will be available from Digi, have not been tested. Click the “Download new versions...” button, select “File,” and browse to the Dynamic C **DCRabbit... \ Utilities \ X-CTU \ MODEMFW** folder to select the **XB24-ZB\_2x... .zip** file.

**NOTE:** All the available new firmware versions will be downloaded if you select “Web” and are connected to the Internet.

2. Select this ZIP file, click **Open, OK**, then **Done**.
3. Choose “XB24-ZB” from the “Modem” pull-down menu for XBee ZB RF modules, or choose “XB24-B” for XBee ZNet RF modules. Other firmware choices have not been tested for compatibility with Dynamic C.
4. Select whether you are configuring an API or AT coordinator, router, or end device from the “Function Set” pull-down menu. Other firmware choices have not been tested for compatibility with Dynamic C.
5. Check the “Always update firmware” box.
6. Click **Write** to load the new firmware.

**NOTE:** The default setup for the XBee ZB RF modules in the ZigBee Mesh Add-On Kit is an API end device. If you download any AT firmware, remember to uncheck “Enable API” under “Host Setup” on the “PC Settings” tab and change the baud rate to **9600** baud before doing any further configuration or attempting to download any more firmware to this XBee RF module.

The “*Customizing Configuration*” section explains how to customize the XBee RF module’s configuration, for example, by assigning a name or changing the baud rate.

## Customizing Configuration

1. Click **Read** to read the firmware.
2. There are networking and I/O settings that need to be set. Before changing any configurations, it is a good idea to press the **Show Defaults** button to set all the values to their default states. Here are the most likely settings you will then configure. Refer to Digi's online *X-CTU Configuration and Test Utility Software User's Guide* for more information.

### Networking

ID - Pan ID = 0 – 0xFFFFFFFFFFFFFFFF;

0 to for the coordinator to assign Pan ID randomly to the first available Pan ID detected; router and end devices with Pan ID = 0 will join the first Pan ID detected and connected to successfully—in future firmware releases it is expected that routers and end devices will scan all Pan IDs, and will then connect to a Pan ID in descending order of signal strength

NJ - Node Join Time = 0–0xFF (seconds). The value of NJ determines the time (in seconds) that the device will allow other devices to join to it. If set to 0xFF, the device will always allow joining. This is an important parameter to set for security so that joining is not always allowed.

### Addressing

DH - Destination Address High and DL - destination Address Low are used only with AT firmware

NI - Node ID = up to a 20-character ASCII string

### Serial Interfacing

BD - Baud Rate = 7 (115200) is the default baud rate interface with Rabbit microprocessors

### Sleep Mode

The sleep mode is particularly useful for an end device so that it may be in a power-conserving sleep state when not being used.

SM - Sleep Mode =5 (CYCLIC SLEEP PIN WAKE) is the preferred sleep mode setting, and allows the XBEE RF module configured as an end device to be awakened by a signal on the wake up pin (pin 9).

ST = Time before Sleep (time period of inactivity — no serial or RF data is sent or received — before activating Sleep Mode, 0 – 0xFFFFE milliseconds)

SP = Cyclic sleep period for cyclic sleeping remotes, set to match SP on end devices. On routers and coordinators, SP determines the transmission and buffering timeouts when sending a message to a sleeping end device. (0x20 – 0xAF0 × 10 milliseconds)

SN = Number of cyclic sleep cycles the on/sleep pin should remain low during cyclic sleep. Set to 1 to set on/sleep high after each SP time (default). (1 – 0xFFFF)

3. Click **Write** to load the XBee RF module with the configurations.

**NOTE:** If you changed the baud rate, remember to use the new baud rate on the “PC Settings” tab before doing any further configuration or attempting to download any more firmware to this XBee RF module.

## Troubleshooting

The two most common issues arising while using the X-CTU utility with XBee RF modules on Rabbit-based products are associated with the PC settings and the sleep mode configuration.

The “PC Settings” tab must reflect the state of the XBee RF module at all times for communication to work. Writing new firmware will revert the serial parameters to their defaults, which means the baud rate will revert to **9600** baud. If these defaults differ from the previous configuration, the “PC Settings” tab must be updated before the X-CTU utility can continue to communicate with the XBee RF module. Any modification of the **Serial Interfacing** parameters on the “Modem Configuration” tab, either directly or through a profile, will require a corresponding update of the “PC Settings” tab. If you are having trouble communicating with the XBee RF module through the X-CTU utility, it may be due to such a change. Try different combinations of serial parameters until you can communicate — most often this is simply a matter of switching between **115200** and **9600** baud, or checking/unchecking “Enable API” under “Host Setup.”

If you use SM = 4 (CYCLIC SLEEP), any future attempt to communicate with the XBee RF module using the X-CTU utility will fail while the XBee RF module is asleep. Such an XBee RF module may be awakened for the ST time by cycling the power off/on, which will reset it. (If attempts to awaken it prove insurmountable, the XBee RF module may be reprogrammed by downloading a different configuration using the Dynamic C **SAMPLES\XBEE\MODEMFWLOAD.C** sample program with the XBee RF module installed on a ZigBee RabbitCore module or single-board computer. See the User’s Manual for the ZigBee RabbitCore module or single-board computer for more information on this sample program.)

## Use of API and AT Commands

The XBee RF modules are loaded by default as end devices with API firmware,.

AT commands originated with modems, and generally consist of the letters AT (for attention) followed by a short text string to describe the actual command. If you are connecting a simple serial device that does not have to interact with the XBee RF module, use the AT firmware for transparent serial operation.

API firmware is loaded by default on XBee RF modules in the ZigBee Mesh Add-On Kit, and on ZigBee RabbitCore modules and single-board computers; ZigBee RabbitCore modules and single-board computers are set up as routers by default. The API firmware allows for a larger range of functionality than is possible with the AT firmware. However, even with API firmware, the driver can still send AT commands embedded in API frames to the XBee RF module.

**Rabbit — A Digi International Brand**

[www.rabbit.com](http://www.rabbit.com)