Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

PRODUCT MANUAL

RabbitCore RCM4500W

C-Programmable ZigBee Core Module

User's Manual

019–0161 • 090515–G

RabbitCore RCM4500W User's Manual

Part Number 019-0161 • 090515–G • Printed in U.S.A.

Digi International reserves the right to make changes and improvements to its products without providing notice.

Trademarks

Rabbit, RabbitCore, and Dynamic C are registered trademarks of Digi International Inc.

ZigBee is a registered trademark of the ZigBee Alliance.

Digi is a registered trademark of Digi International Inc.

Rabbit 4000 is a trademark of Digi International Inc.

The latest revision of this manual is available on the Rabbit Web site, www.rabbit.com, for free, unregistered download.

Digi International Inc.

www.rabbit.com

TABLE OF CONTENTS

1. INTRODUCTION

The RCM4510W next-generation RabbitCore modules add ZigBee®/ 802.15.4 functionality to the existing Rabbit® 4000 microprocessor features to allow you to create a low-cost, low-power, embedded wireless control and communications solution for your embedded control system. The Rabbit® 4000 microprocessor features include hardware DMA, clock speeds of up to 60 MHz, I/O lines shared with up to six serial ports and four levels of alternate pin functions that include variable-phase PWM, auxiliary I/O, quadrature decoder, and input capture. Coupled with more than 500 new opcode instructions that help to reduce code size and improve processing speed, this equates to a core module that is fast, efficient, and the ideal solution for a wide range of wireless embedded applications.

The Development Kit has the essentials that you need to design your own wireless microprocessor-based system, and includes a complete Dynamic C software development system. This Development Kit also contains a Prototyping Board that will allow you to evaluate the RCM4510W modules and to prototype circuits that interface to the RCM4510W modules. You will also be able to write and test software for these modules.

In addition to onboard ZigBee/802.15.4 functionality, the RCM4510W model has a Rabbit 4000 microprocessor operating at 29.49 MHz, static RAM, flash memory, two clocks (main oscillator and timekeeping), and the circuitry necessary for reset and management of battery backup of the Rabbit 4000's internal real-time clock and the static RAM. One 50-pin header brings out the Rabbit 4000 I/O bus lines, parallel ports, and serial ports. A separate 14-pin auxiliary I/O header brings out up to nine additional I/O pins (up to four of which may be configured as analog inputs) made possible by the onboard XBee RF module.

The RCM4510W receives its +3.3 V power from the customer-supplied motherboard on which it is mounted. The RCM4510W can interface with all kinds of CMOS-compatible digital devices through the motherboard.

1.1 RCM4510W Features

- Small size: 1.84" × 2.85" × 0.54"
  (47 mm × 72 mm × 14 mm)
- Microprocessor: Rabbit 4000 running at 29.49 MHz
- Up to 40 general-purpose I/O lines configurable with up to four alternate functions
- Up to 9 additional general-purpose I/O lines (up to four of which may be set up as analog inputs) available through the XBee RF module
- 3.3 V I/O lines with low-power modes down to 2 kHz
- Six CMOS-compatible serial ports — four ports are configurable as a clocked serial port (SPI), and two ports are configurable as SDLC/HDLC serial ports.
- Alternate I/O bus can be configured for 8 data lines and 6 address lines (shared with parallel I/O lines)
- 512K flash memory, 512K data SRAM
- Real-time clock
- Watchdog supervisor

Currently there is one production model with a choice of XBee firmware. Table 1 summarizes its main features.

Table 1. RCM4510W Features

<table>
<tr><th>Feature</th><th colspan="2">RCM4510W</th></tr>
<tr><td>Microprocessor</td><td colspan="2">Rabbit® 4000 at 29.49 MHz</td></tr>
<tr><td>Flash Memory</td><td colspan="2">512K</td></tr>
<tr><td>Data SRAM</td><td colspan="2">512K</td></tr>
<tr><td>Serial Ports</td><td colspan="2">6 shared high-speed, CMOS-compatible ports:
6 are configurable as asynchronous serial ports;
4 are configurable as clocked serial ports (SPI);
2 are configurable as SDLC/HDLC serial ports;
1 asynchronous serial port is shared with the XBee RF module
1 asynchronous serial port is used during programming</td></tr>
<tr><td>XBee RF Module</td><td colspan="2">Digi International® XBee® ZB (802.15.4 standard, ISM 2.4 GHz)</td></tr>
<tr><td rowspan="2">XBee Firmware</td><td>ZNet 2.5</td><td>RCM4510W (ZNet) (20-101-1207)</td></tr>
<tr><td>ZigBee PRO</td><td>RCM4510W (ZB) (20-101-1269)</td></tr>
</table>

Section 1.3.2.1 provides additional information about the two types of XBee firmware.

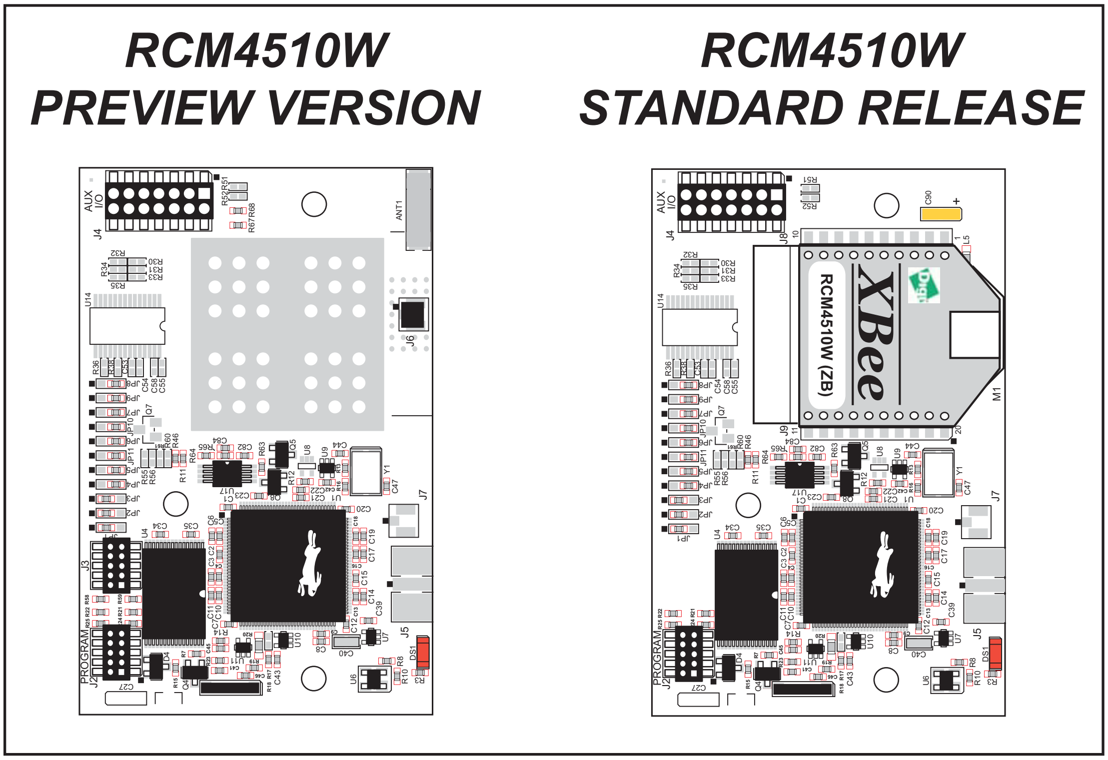


Figure 1. RCM4510W Versions

There are two versions of the RCM4510W model—the *standard release*, available after April, 2007, is identical in form and function to the *preview version*. The difference between them is that the RF module is made of discrete onboard components in the preview version, and is included in the pluggable Digi International® XBee® RF module on the standard release. The height of the preview version is also about 0.01" (0.2 mm) less than that of the standard release; Rabbit recommends that you use the dimensions for the standard release specified in this manual in your design. *The preview version has not undergone certification testing and is intended for development purposes only. The preview version will no longer be offered once the standard release is available.*

NOTE: At the present time the Digi International® XBee® RF modules used with the RCM4510W are not compatible with other XBee® and XBee PRO® RF modules such as those used with Rabbit's ZigBee®/802.15.4 Application Kit

The RCM4510W is programmed over a standard PC USB port through a programming cable supplied with the Development Kit.

NOTE: The RabbitLink cannot be used to program the RCM4510W or other RabbitCore modules based on the Rabbit 4000 microprocessor.

Appendix A provides detailed specifications for the RCM4510W.

1.2 Advantages of the RCM4510W

- Fast time to market using a fully engineered, "ready-to-run/ready-to-program" microprocessor core.
- Competitive pricing when compared with the alternative of purchasing and assembling individual components.
- Easy C-language program development and debugging
- Rabbit Field Utility to download compiled Dynamic C .bin files, and cloning board options for rapid production loading of programs.
- Generous memory size allows large programs with tens of thousands of lines of code, and substantial data storage.
- Reference application uses a low-cost, low-power ZigBee/802.15.4 infrastructure to connect Rabbit-based devices
- Supports ZigBee/802.15.4 point-to-point, point-to-multipoint, and mesh topologies
- Easily scalable for commercial deployment applications
- RCM4510W can function as a network coordinator, router, or end device

1.3 Development and Evaluation Tools

1.3.1 RCM4510W Development Kit

The RCM4510W Development Kit contains the hardware essentials you will need to use the RCM4510W module. The items in the Development Kit and their use are as follows.

- RCM4510W module.
- Prototyping Board.
- Universal AC adapter, 12 V DC, 1 A (includes Canada/Japan/U.S., Australia/N.Z., U.K., and European style plugs). Development Kits sold in North America may contain an AC adapter with only a North American style plug.
- USB programming cable with 10-pin header.
- 10-pin header to DB9 serial cable.
- 14-pin IDC socket connector with bare leads ribbon cable.
- Digi® XBee USB (used as ZigBee coordinator).
- ***Dynamic C***® CD-ROM, with complete product documentation on disk.
- ***Getting Started*** instructions.
- A bag of accessory parts for use on the Prototyping Board.
- ***Rabbit 4000 Processor Easy Reference*** poster.
- Registration card.

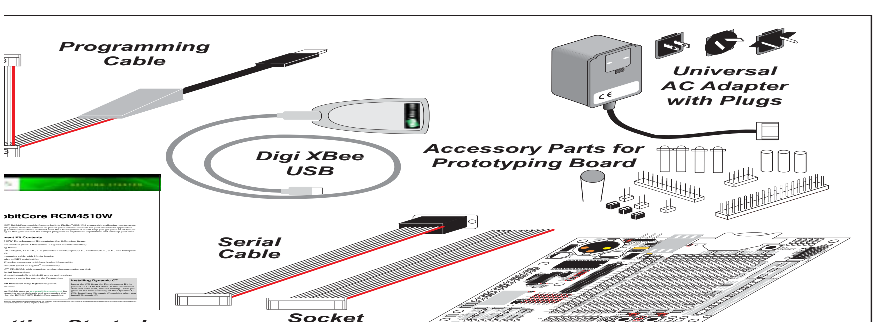


Figure 2. RCM4510W Development Kit

1.3.2 Software

The RCM4510W preview version is programmed using version 10.11 or later of Dynamic C, and the standard version requires version 10.21 or later. A compatible version is included on the Development Kit CD-ROM.

Starting with Dynamic C version 10.40, Dynamic C includes the popular µC/OS-II real-time operating system, point-to-point protocol (PPP), FAT file system, RabbitWeb, and other select libraries. Rabbit also offers for purchase the Rabbit Embedded Security Pack featuring the Secure Sockets Layer (SSL) and a specific Advanced Encryption Standard (AES) library.

In addition to the Web-based technical support included at no extra charge, a one-year telephone-based technical support module is also available for purchase. Visit our Web site at www.rabbit.com or contact your Rabbit sales representative or authorized distributor for further information.

1.3.2.1 XBee Firmware

There are two types of XBee firmware available for the RCM4510W — ZNet 2.5 and ZigBee PRO. The firmware preloaded at the factory on your RCM4510W is indicated with the model number and is reflected in different part numbers (see Table 1). A label on the XBee RF module identifies the factory-installed firmware type. Older RCM4510W modules shipped with ZNet firmware and do not have a sticker on their XBee RF module.

RCM4510W (ZNet)   **RCM4510W (ZB)**

- ZNet 2.5 is a networking solution that was developed around an earlier draft of the ZigBee standard, and will work only with the MaxStream or Digi International® XBee® RF modules used with the RCM4510W or with the preview version of the RCM4510W. Applications built around ZNet 2.5 may be upgraded to the ZB solution by loading the ZB firmware (standard release RCM4510W RabbitCore modules only) and by recompiling the application using Dynamic C v. 10.46 or later.
- ZB firmware was developed using the ZigBee PRO feature set and provides the most advanced ZigBee networking capabilities that will interface with any other devices based on the ZigBee PRO standard adopted in 2007.

  Dynamic C v. 10.46 and later versions support the ZB firmware with a 64-bit extended PAN ID instead of a 16-bit PAN ID

  Rabbit recommends ZB firmware for new customers and new applications.

> **NOTE:** While both ZNet 2.5 and ZB firmware types are technically ZigBee, the protocol layer in them is different and devices with one type of firmware will not function in a network with devices using the other firmware.

1.3.3 Optional Add-Ons

Rabbit has a Mesh Network Add-On Kit available for the RCM4510W.

- Mesh Network Add-On Kit (Part No. 101-1272)
    - ▶ Digi® XBee USB (used as ZigBee coordinator)
    - ▶ Two XBee ZB RF modules
    - ▶ Two RF Interface modules
    - ▶ Digi® XBee USB and serial cables

  The XBee ZB RF module is installed on the RF Interface module, which can be connected via an RS-232 serial connection to a Windows PC for setup. The Mesh Network Add-On Kit enables you to add additional devices to explore a ZigBee mesh network.

Contact your authorized Rabbit distributor or your sales representative for more information, or visit our Web site.

1.3.4 Online Documentation

The online documentation is installed along with Dynamic C, and an icon for the documentation menu is placed on the workstation's desktop. Double-click this icon to reach the menu. If the icon is missing, use your browser to find and load **default.htm** in the **docs** folder, found in the Dynamic C installation folder.

The latest versions of all documents are always available for free, unregistered download from our Web sites as well.

2. GETTING STARTED

This chapter describes the RCM4510W hardware in more detail, and explains how to set up and use the accompanying Prototyping Board.

NOTE: This chapter (and this manual) assume that you have the RCM4510W Development Kit. If you purchased an RCM4510W module by itself, you will have to adapt the information in this chapter and elsewhere to your test and development setup.

2.1 Install Dynamic C

To develop and debug programs for the RCM4510W modules (and for all other Rabbit hardware), you must install and use Dynamic C.

If you have not yet installed Dynamic C version 10.11 (or a later version), do so now by inserting the Dynamic C CD from the Development Kit in your PC's CD-ROM drive. If autorun is enabled, the CD installation will begin automatically.

If autorun is disabled or the installation does not start, use the Windows **Start | Run** menu or Windows Disk Explorer to launch **`setup.exe`** from the root folder of the CD-ROM.

The installation program will guide you through the installation process. Most steps of the process are self-explanatory.

Dynamic C uses a COM (serial) port to communicate with the target development system. The installation allows you to choose the COM port that will be used. The default selection is COM1. You may select any available port for Dynamic C's use. If you are not certain which port is available, select COM1. This selection can be changed later within Dynamic C.

NOTE: The installation utility does not check the selected COM port in any way. Specifying a port in use by another device (mouse, modem, etc.) may lead to a message such as **`"could not open serial port"`** when Dynamic C is started.

Once your installation is complete, you will have up to three new icons on your PC desktop. One icon is for Dynamic C, another opens the documentation menu, and the third is for the Rabbit Field Utility, a tool used to download precompiled software to a target system.

If you have purchased any of the optional Dynamic C modules, install them after installing Dynamic C. The modules may be installed in any order. You must install the modules in the same directory where Dynamic C was installed.

2.2 Hardware Connections

There are three steps to connecting the Prototyping Board for use with Dynamic C and the sample programs:

1. Prepare the Prototyping Board for Development.
2. Attach the RCM4510W module to the Prototyping Board.
3. Connect the programming cable between the RCM4510W and the PC.
4. Connect the power supply to the Prototyping Board.

2.2.1 Step 1 — Prepare the Prototyping Board for Development

Snap in four of the plastic standoffs supplied in the bag of accessory parts from the Development Kit in the holes at the corners on the bottom side of the Prototyping Board as shown in Figure 3.

> **NOTE:** Pay attention to use the hole that is pointed out towards the bottom left of the Prototyping Board since the hole below it is used for a standoff when mounting the RCM4510W on the Prototyping Board.

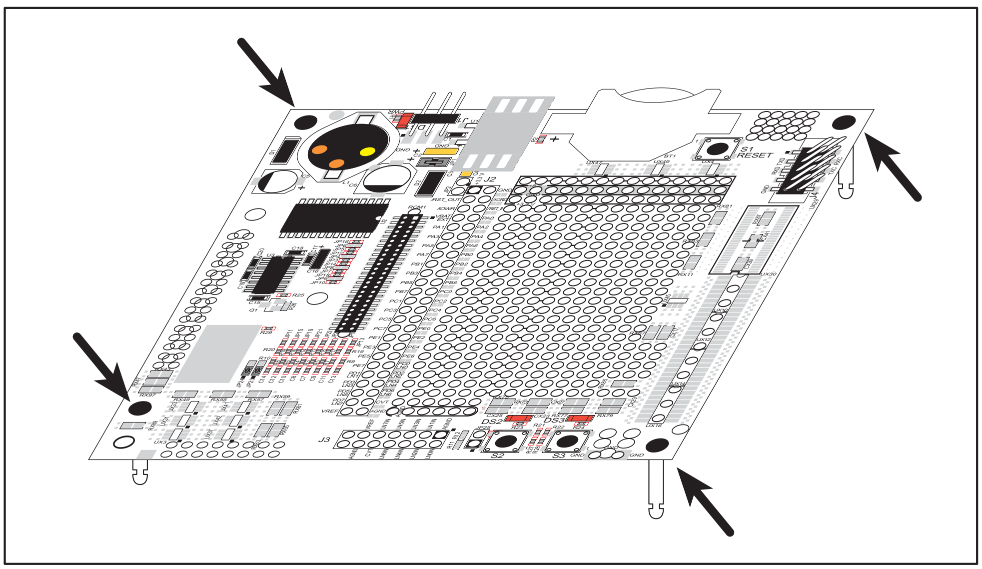


Figure 3. Insert Standoffs

2.2.2 Step 2 — Attach Module to Prototyping Board

Turn the RCM4510W module so that the mounting holes line up with the corresponding holes on the Prototyping Board with the programming header at the top right. Insert the metal standoffs as shown in Figure 4, secure them from the bottom using the 4-40 screws and washers, then insert the module's header J1 on the bottom side into socket RCM1 on the Prototyping Board.

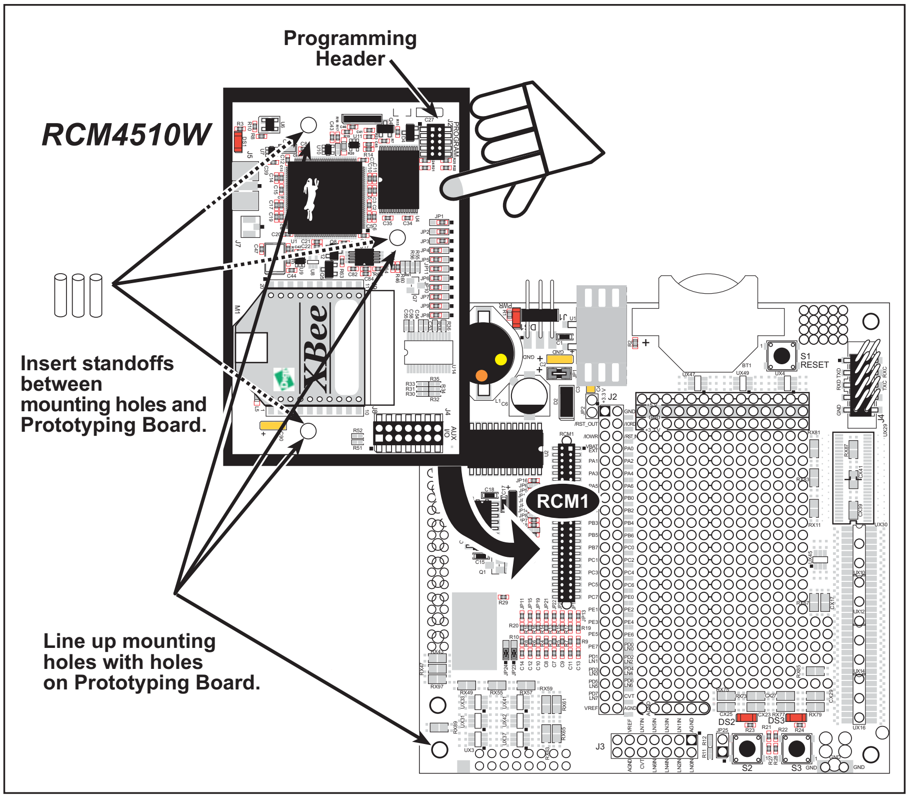
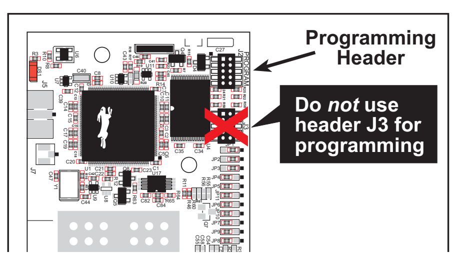


Figure 4. Install the Module on the Prototyping Board

NOTE: It is important that you line up the pins on header J1 of the module exactly with socket RCM1 on the Prototyping Board. The header pins may become bent or damaged if the pin alignment is offset, and the module will not work. Permanent electrical damage to the module may also result if a misaligned module is powered up.

Press the module's pins gently into the Prototyping Board socket—press down in the area above the header pins. For additional integrity, you may secure the RCM4510W to the standoffs from the top using the remaining three 4-40 screws and washers.

NOTE: If you are using the preview version of the RCM4510W, do not connect the programming cable to header J3 (shown below the programming header at right). Header J3 is used only by the factory.



2.2.3 Step 3 — Connect Programming Cable

The programming cable connects the module to the PC running Dynamic C to download programs and to monitor the module during debugging.

Connect the 10-pin connector of the programming cable labeled **PROG** to header J2 on the RCM4510W as shown in Figure 5. Be sure to orient the marked (usually red) edge of the cable towards pin 1 of the connector. (Do not use the **DIAG** connector, which is used for a standard serial connection.)

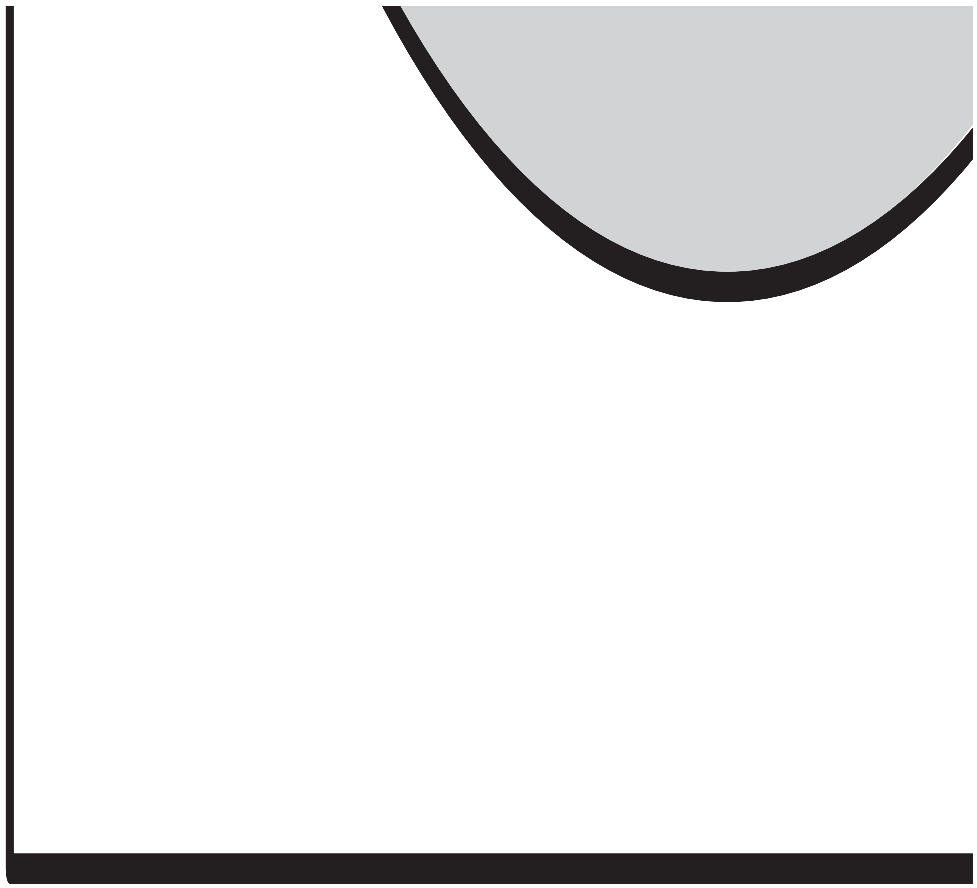


Figure 5. Connect Programming Cable and Power Supply

> **NOTE:** Never disconnect the programming cable by pulling on the ribbon cable. Carefully pull on the connector to remove it from the header.

Connect the other end of the programming cable to an available USB port on your PC or workstation.

Your PC should recognize the new USB hardware, and the LEDs in the shrink-wrapped area of the USB programming cable will flash — if you get an error message, you will have to install USB drivers. Drivers for Windows XP are available in the Dynamic C **`Drivers\Rabbit USB Programming Cable\WinXP_2K`** folder — double-click **`DPInst.exe`** to install the USB drivers. Drivers for other operating systems are available online at www.ftdichip.com/Drivers/VCP.htm.

2.2.4 Step 4 — Connect Power

Once all the other connections have been made, you can connect power to the Prototyping Board.

First, prepare the AC adapter for the country where it will be used by selecting the appropriate plug. The RCM4510W Development Kit presently includes Canada/Japan/U.S., Australia/N.Z., U.K., and European style plugs. Snap in the top of the plug assembly into the slot at the top of the AC adapter as shown in Figure 5, then press down on the spring-loaded clip below the plug assembly to allow the plug assembly to click into place. Release the clip to secure the plug assembly in the AC adapter.

Connect the AC adapter to 3-pin header J1 on the Prototyping Board as shown in Figure 5. The connector may be attached either way as long as it is not offset to one side—the center pin of J1 is always connected to the positive terminal, and either edge pin is ground.

Plug in the AC adapter. The **PWR** LED on the Prototyping Board next to the power connector at J1 should light up. The RCM4510W and the Prototyping Board are now ready to be used.

> **NOTE:** A **RESET** button is provided on the Prototyping Board next to the battery holder to allow a hardware reset without disconnecting power.

To power down the Prototyping Board, unplug the power connector from J1. You should disconnect power before making any circuit adjustments in the prototyping area, changing any connections to the board, or removing the RCM4510W from the Prototyping Board.

2.3 Run a Sample Program

If you already have Dynamic C installed, you are now ready to test your programming connections by running a sample program. Start Dynamic C by double-clicking on the Dynamic C icon on your desktop or in your **Start** menu.

You may have to select the COM port assigned to the USB programming cable on your PC. In Dynamic C, select **Options > Project Options**, then select this COM port on the "Communications" tab. Then check "Use USB to Serial Converter" in "Serial Options." Click **OK** to save the settings.

Find the file **`PONG.C`**, which is in the Dynamic C **`SAMPLES`** folder. To run the program, open it with the **File** menu, then compile and run it by pressing **F9**. The **STDIO** window will open on your PC and will display a small square bouncing around in a box.

2.3.1 Troubleshooting

If you receive the message **`Could Not Open Serial Port`**, check that the COM port assigned to the USB programming cable was identified and set up in Dynamic C as described in the preceding section.

If you receive the message **`No Rabbit Processor Detected`**, the programming cable may be connected to the wrong COM port, a connection may be faulty, or the target system may not be powered up. First, check to see that the power LED on the Prototyping Board is lit. If the LED is lit, check both ends of the programming cable to ensure that it is firmly plugged into the PC and the programming header on the RCM4510W with the marked (colored) edge of the programming cable towards pin 1 of the programming header. Ensure that the module is firmly and correctly installed in its connectors on the Prototyping Board.

If Dynamic C appears to compile the BIOS successfully, but you then receive a communication error message when you compile and load a sample program, it is possible that your PC cannot handle the higher program-loading baud rate. Try changing the maximum download rate to a slower baud rate as follows.

- Locate the **Serial Options** dialog on the "Communications" tab in the Dynamic C **Options > Project Options** menu. Select a slower Max download baud rate.

If a program compiles and loads, but then loses target communication before you can begin debugging, it is possible that your PC cannot handle the default debugging baud rate. Try lowering the debugging baud rate as follows.

- Locate the **Serial Options** dialog on the "Communications" tab in the Dynamic C **Options > Project Options** menu. Choose a lower debug baud rate.

Press **<Ctrl-Y>** to force Dynamic C to recompile the BIOS. You should receive a **`Bios compiled successfully`** message once this step is completed successfully.

2.3.2 Run a ZigBee Sample Program

This section explains how to run a sample program in which the RCM4510W is used in its default setup as a router and the Digi® XBee USB is used as the ZigBee coordinator.

1. Connect the Digi® XBee USB acting as a ZigBee coordinator to an available USB port on your PC or workstation. Your PC should recognize the new USB hardware.

2. Find the file **`AT_INTERACTIVE.C`**, which is in the Dynamic C **`SAMPLES\XBee`** folder. To run the program, open it with the **File** menu, then compile and run it by pressing **F9**. The Dynamic C **STDIO** window will open to display a list of AT commands. Type **`MENU`** to redisplay the menu of commands.

```
Stdio

Waiting to join network...
done
Cmd  - Description
=====================
ATCH - Read the current channel. Will be zero if we
       are not associated with a network.
ATID - Set or read the current PAN ID. If you set the  ID you
       must write it to non-volatile memory ("WR") and
       then reset the network software ("NR").
ATOP - Read the operating PAN ID.
ATMY - Read the current network address. Will be 0xFFFE
       if we are not associated with a network.
ATSH - Read the upper four bytes of the radio IEEE address.
ATSL - Read the lower four bytes of the radio IEEE address.
ATNI - Set or read the Node Identifier.
ATBH - Set or read the maximum number of Broadcast Hops.
ATNT - Set or read the Node Discovery timeout value (in 0.1s).
ATSC - Set or read the list of channels to scan. This
       value is a bit-field list.
ATSD - Set or read the channel scan duration value.
ATNJ - Set or read the Node Joining Time value.
ATAI - Read the Association Indicator. A zero value
       means we are associated with a network.
ATPL - Set or read the transmission power level.
ATVR - Read the radio software version number.
ATHV - Read the radio hardware version number.

MENU - Display this menu (not an AT command.)

Valid command formats (AT prefix is optional, CC is command):

[AT]CC 0xXXXXXX (where XXXXXX is an even number of hexidecimal characters)
[AT]CC YYYY (where YYYY is an integer, up to 32 bits)
[AT]NI "Node ID String" (where quotes contain string data)

Enter AT Command:
```

Appendix D provides additional configuration information if you experience conflicts while doing development simultaneously with more than one ZigBee coordinator, or if you wish to install new firmware. Different firmware must be installed to use the RCM4510W as either a coordinator or as an end device.

2.4 Where Do I Go From Here?

If the sample program ran fine, you are now ready to go on to the sample programs in Chapter 3 and to develop your own applications. The sample programs can be easily modified for your own use. The user's manual also provides complete hardware reference information and software function calls for the RCM4510W series of modules and the Prototyping Board.

For advanced development topics, refer to the ***Dynamic C User's Manual***, also in the online documentation set. ***An Introduction to ZigBee*** provides background information on the ZigBee protocol, and is available on the CD and on our Web site.

2.4.1 Technical Support

> **NOTE:** If you purchased your RCM4510W through a distributor or through a Rabbit partner, contact the distributor or partner first for technical support.

If there are any problems at this point:

- Use the Dynamic C **Help** menu to get further assistance with Dynamic C.
- Check the Rabbit Technical Bulletin Board and forums at www.rabbit.com/support/bb/ and at www.rabbitcom/forums/.
- Use the Technical Support e-mail form at www.rabbit.com/support/.

3. RUNNING SAMPLE PROGRAMS

To develop and debug programs for the RCM4510W (and for all other Rabbit hardware), you must install and use Dynamic C. This chapter provides a tour of its major features with respect to the RCM4510W.

3.1 Introduction

To help familiarize you with the RCM4510W modules, Dynamic C includes several sample programs. Loading, executing, and studying these programs will give you a solid hands-on overview of the RCM4510W's capabilities, as well as a quick start with Dynamic C as an application development tool.

This chapter provides sample programs that illustrate the digital I/O and serial capabilities of the RCM4510W RabbitCore module. Section 6.2 discusses the sample programs that illustrate the ZigBee features.

> **NOTE:** The sample programs assume that you have at least an elementary grasp of the C language. If you do not, see the introductory pages of the ***Dynamic C User's Manual*** for a suggested reading list.

In order to run the sample programs discussed in this chapter and elsewhere in this manual,

1. Your module must be plugged in to the Prototyping Board as described in Chapter 2, "Getting Started."
2. Dynamic C must be installed and running on your PC.
3. The programming cable must connect the programming header on the module to your PC.
4. Power must be applied to the module through the Prototyping Board.

Refer to Chapter 2, "Getting Started," if you need further information on these steps.

To run a sample program, open it with the **File** menu, then compile and run it by pressing **F9**.

Each sample program has comments that describe the purpose and function of the program. Follow the instructions at the beginning of the sample program.

More complete information on Dynamic C is provided in the ***Dynamic C User's Manual***.

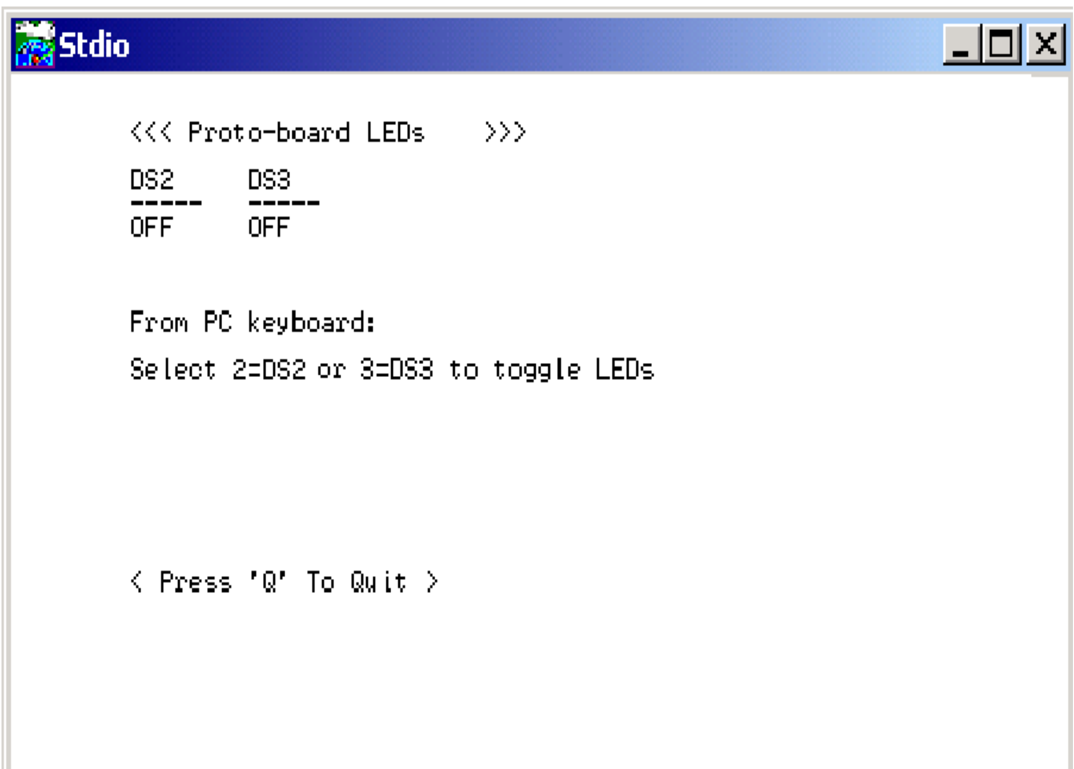
3.2 Sample Programs

Of the many sample programs included with Dynamic C, several are specific to the RCM4510W modules. These programs will be found in the **SAMPLES\RCM4500W** folder.

- **CONTROLLED.C**—Demonstrates use of the digital outputs by having you turn LEDs DS2 and DS3 on the Prototyping Board on or off from the **STDIO** window on your PC.

  Parallel Port B bit 2 = LED DS2
  Parallel Port B bit 3 = LED DS3

  Once you compile and run **CONTROLLED.C**, the following display will appear in the Dynamic C **STDIO** window.



  Press “2” or “3” on your keyboard to select LED DS2 or DS3 on the Prototyping Board. Then follow the prompt in the Dynamic C **STDIO** window to turn the LED ON or OFF. A logic low will light up the LED you selected.

- **FLASHLED1.C**—demonstrates the use of assembly language to flash LEDs DS2 and DS3 on the Prototyping Board at different rates. Once you have compiled and run this program, LEDs DS2 and DS3 will flash on/off at different rates.

- **FLASHLED2.C**—demonstrates the use of cofunctions and costatements to flash LEDs DS2 and DS3 on the Prototyping Board at different rates. Once you have compiled and run this program, LEDs DS2 and DS3 will flash on/off at different rates.

- **`TAMPERDETECTION.C`**—demonstrates how to detect an attempt to enter the bootstrap mode. When an attempt is detected, the battery-backed onchip-encryption RAM on the Rabbit 4000 is erased. This battery-backed onchip-encryption RAM can be useful to store data such as an AES encryption key from a remote location.

  This sample program shows how to load and read the battery-backed onchip-encryption RAM and how to enable a visual indicator.

  Once this sample is compiled and running (you pressed the **F9** key while the sample program is open), remove the programming cable and press the reset button on the Prototyping Board to reset the module. LEDs DS2 and DS3 will be flashing on and off.

  Now press switch S2 to load the battery-backed RAM with the encryption key. The LEDs are now on continuously. Notice that the LEDs will stay on even when you press the reset button on the Prototyping Board.

  Reconnect the programming cable briefly and unplug it again to simulate an attempt to access the onchip-encryption RAM. The LEDs will be flashing because the battery-backed onchip-encryption RAM has been erased. Notice that the LEDs will continue flashing even when you press the reset button on the Prototyping Board.

  You may press switch S2 again and repeat the last steps to watch the LEDs.

- **`TOGGLESWITCH.C`**—demonstrates the use of costatements to detect switch presses using the press-and-release method of debouncing. LEDs DS2 and DS3 on the Prototyping Board are turned on and off when you press switches S2 and S3. S2 and S3 are controlled by PB4 and PB5 respectively.

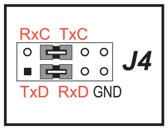
Once you have loaded and executed these five programs and have an understanding of how Dynamic C and the RCM4510W modules interact, you can move on and try the other sample programs.

3.2.1 Serial Communication

The following sample programs are found in the **SAMPLES\RCM4500W\SERIAL** folder.

- **FLOWCONTROL.C**—This program demonstrates how to configure Serial Port D for CTS/RTS flow control with serial data coming from Serial Port C (TxC) at 115,200 bps. The serial data received are displayed in the **STDIO** window.

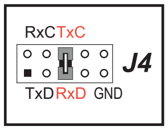
  To set up the Prototyping Board, you will need to tie TxD and RxD together on the RS-232 header at J4, and you will also tie TxC and RxC together using the jumpers supplied in the Development Kit as shown in the diagram.

  A repeating triangular pattern should print out in the **STDIO** window. The program will periodically switch flow control on or off to demonstrate the effect of flow control.

  If you have two Prototyping Boards with modules, run this sample program on the sending board, then disconnect the programming cable and reset the sending board so that the module is operating in the Run mode. Connect TxC, TxD, and GND on the sending board to RxC, RxD, and GND on the other board, then, with the programming cable attached to the other module, run the sample program.

- **PARITY.C**—This program demonstrates the use of parity modes by repeatedly sending byte values 0–127 from Serial Port C to Serial Port D. The program will switch between generating parity or not on Serial Port C. Serial Port D will always be checking parity, so parity errors should occur during every other sequence.

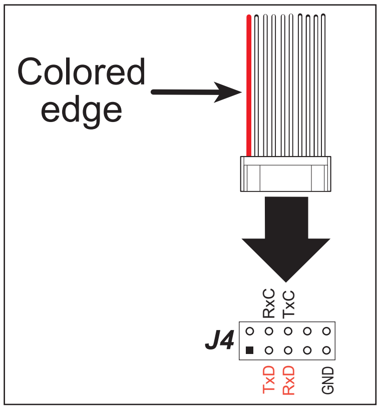
  

  To set up the Prototyping Board, you will need to tie TxC and RxD together on the RS-232 header at J4 using one of the jumpers supplied in the Development Kit as shown in the diagram.

  The Dynamic C **STDIO** window will display the error sequence.

- **SERDMA.C**—This program demonstrates using DMA to transfer data from a circular buffer to the serial port and vice versa. The Dynamic C **STDIO** window is used to view or clear the buffer.

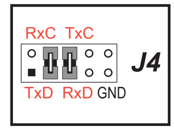
  Before you compile and run the sample program, you will need to connect the RS-232 header at J4 to your PC as shown in the diagram using the serial to DB9 cable supplied in the Development Kit.

  Once you have compiled and run the sample program, start Tera Term or another terminal emulation program to connect to the selected PC serial port at a baud rate of 115,200 bps. You can observe the output in the Dynamic C **STDIO** window as you type in Tera Term, and you can also use the Dynamic C **STDIO** window to clear the buffer.

  The Tera Term utility can be downloaded from hp.vector.co.jp/authors/VA002416/teraterm.html.

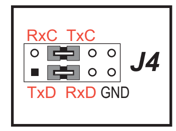
- **`SIMPLE3WIRE.C`**—This program demonstrates basic RS-232 serial communication. Lower case characters are sent on TxC, and are received by RxD. The received characters are converted to upper case and are sent out on TxD, are received on RxC, and are displayed in the Dynamic C **STDIO** window.

  To set up the Prototyping Board, you will need to tie TxD and RxC together on the RS-232 header at J4, and you will also tie RxD and TxC together using the jumpers supplied in the Development Kit as shown in the diagram.

- **`SIMPLE5WIRE.C`**—This program demonstrates 5-wire RS-232 serial communication with flow control on Serial Port D and data flow on Serial Port C.

  To set up the Prototyping Board, you will need to tie TxD and RxD together on the RS-232 header at J4, and you will also tie TxC and RxC together using the jumpers supplied in the Development Kit as shown in the diagram.

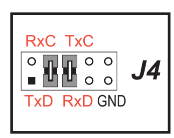
  

  Once you have compiled and run this program, you can test flow control by disconnecting the TxD jumper from RxD while the program is running. Characters will no longer appear in the **STDIO** window, and will display again once TxD is connected back to RxD.

  If you have two Prototyping Boards with modules, run this sample program on the sending board, then disconnect the programming cable and reset the sending board so that the module is operating in the Run mode. Connect TxC, TxD, and GND on the sending board to RxC, RxD, and GND on the other board, then, with the programming cable attached to the other module, run the sample program. Once you have compiled and run this program, you can test flow control by disconnecting TxD from RxD as before while the program is running. Since the J4 header locations on the two Prototyping Boards are connected with wires, there are no slip-on jumpers at J4 on either Prototyping Board.

- **`SWITCHCHAR.C`**—This program demonstrates transmitting and then receiving an ASCII string on Serial Ports C and D. It also displays the serial data received from both ports in the **STDIO** window.

  To set up the Prototyping Board, you will need to tie TxD and RxC together on the RS-232 header at J4, and you will also tie RxD and TxC together using the jumpers supplied in the Development Kit as shown in the diagram.

  Once you have compiled and run this program, press and release switches S2 and S3 on the Prototyping Board. The data sent between the serial ports will be displayed in the **STDIO** window.