Welcome to E-XFL.COM

Understanding Embedded - Microcontroller, Microprocessor, FPGA Modules

Embedded - Microcontroller, Microprocessor, and FPGA Modules are fundamental components in modern electronic systems, offering a wide range of functionalities and capabilities. Microcontrollers are compact integrated circuits designed to execute specific control tasks within an embedded system. They typically include a processor, memory, and input/output peripherals on a single chip. Microprocessors, on the other hand, are more powerful processing units used in complex computing tasks, often requiring external memory and peripherals. FPGAs (Field Programmable Gate Arrays) are highly flexible devices that can be configured by the user to perform specific logic functions, making them invaluable in applications requiring customization and adaptability.

Applications of Embedded - Microcontroller,

Details	
Product Status	Obsolete
Module/Board Type	MPU Core
Core Processor	Rabbit 2000
Co-Processor	-
Speed	22.1MHz
Flash Size	256KB
RAM Size	128KB
Connector Type	2 IDC Headers 2x20
Size / Dimension	2" x 3.5" (51mm x 89mm)
Operating Temperature	-40°C ~ 70°C
Purchase URL	https://www.e-xfl.com/product-detail/digi-international/101-0435

2. GETTING STARTED

This chapter describes the RCM2100 hardware in more detail, and explains how to set up and use the accompanying prototyping and development board.

NOTE: This chapter (and this manual) assume that you have the RabbitCore RCM2100 Development Kit. If you purchased an RCM2100 module by itself, you will have to adapt the information in this chapter and elsewhere to your test and development setup.

2.1 Connections

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

1. Attach the RCM2100 module to the Prototyping Board.
2. Connect the programming cable between the RCM2100 module and the workstation PC.
3. Connect the power supply to the Prototyping Board.

2.1.3 Connect Power

When all other connections have been made, you can connect power to the RCM2100 Prototyping Board.

Hook the connector from the wall transformer to header J5 on the Prototyping Board as shown in Figure 4 below. The connector may be attached either way as long as it is not offset to one side.

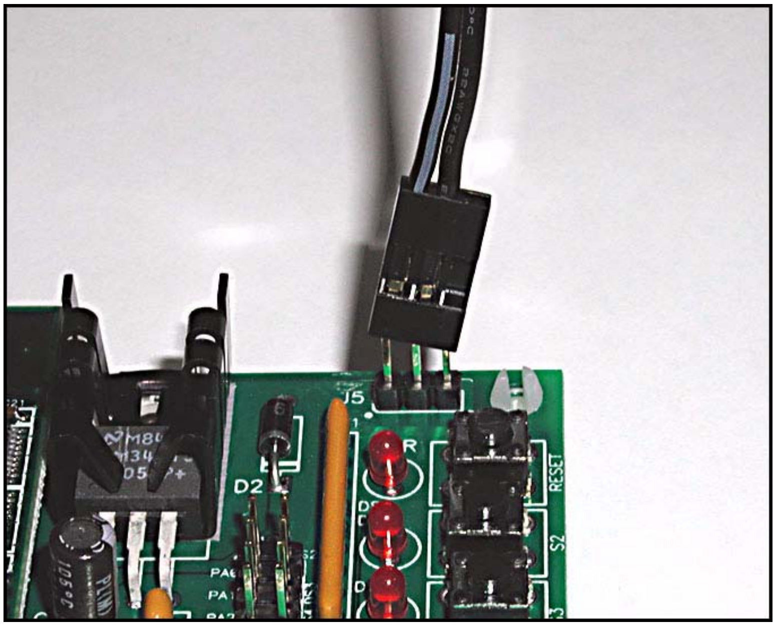


Figure 4. Power Supply Connections to Prototyping Board

Plug in the wall transformer. The power LED on the Prototyping Board should light up. The RCM2100 and the Prototyping Board are now ready to be used.

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

To power down the Prototyping Board, unplug the power connector from J5. You should disconnect power before making any circuit adjustments in the prototyping area, changing any connections to the board, or removing the RCM2100 module from the Prototyping Board.

3. RUNNING SAMPLE PROGRAMS

To develop and debug programs for the RCM2100 (and for all other Rabbit Semiconductor hardware), you must install and use Dynamic C. Dynamic C is an integrated development system for writing embedded software. It runs on an IBM-compatible PC and is designed for use with Rabbit Semiconductor single-board computers and other single-board computers based on the Rabbit microprocessor. This chapter takes you through the installation of Dynamic C, and then provides a tour of the sample programs for the RCM2100.

3.1 Sample Programs

To help familiarize you with the RCM2100 modules, several sample Dynamic C programs have been included. Loading, executing and studying these programs will give you a solid hands-on overview of the RCM2100's capabilities, as well as a quick start with Dynamic C as an application development tool. These programs are intended to serve as tutorials, but then can also be used as starting points or building blocks for your own applications.

> **NOTE:** It is assumed in this section that you have at least an elementary grasp of ANSI C. If you do not, see the introductory pages of the ***Dynamic C User's Manual*** for a suggested reading list.

Each sample program has comments that describe the purpose and function of the program.

Before running any of these sample programs, make sure that your RCM2100 is connected to the Prototyping Board and to your PC as described in Section 2.1, "Connections." To run a sample program, open it with the **File** menu (if it is not already open), then compile and run it by pressing **F9** or by selecting **Run** in the **Run** menu.

Sample programs are provided in the Dynamic C `SAMPLES` folder. Two folders contain sample programs that illustrate features unique to the RabbitCore RCM2100.

- `RCM2100`—Demonstrates the basic operation and the Ethernet functionality of the RabbitCore RCM2100.
- `TCPIP`—Demonstrates more advanced TCP/IP programming for Rabbit Semiconductor's Ethernet-enabled Rabbit-based boards.

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

4. Hardware Reference

Chapter 3 describes the hardware components and principal hardware subsystems of the RabbitCore RCM2100. Appendix A, "RabbitCore RCM2100 Specifications," provides complete physical and electrical specifications.

4.1 RCM2100 Digital Inputs and Outputs

Figure 5 shows the subsystems designed into the RCM2100 modules.

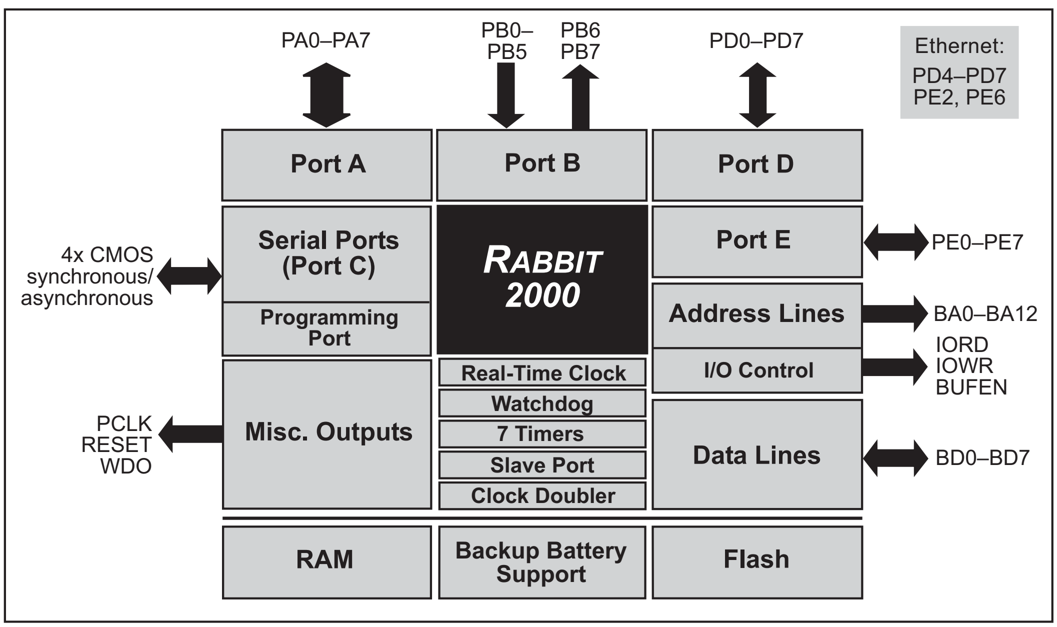


Figure 5. Rabbit Subsystems

The RCM2100 has 40 parallel I/O lines grouped in five 8-bit ports available on headers J1 and J2. The 24 bidirectional I/O lines are located on pins PA0–PA7, PD0–PD7, and PE0–PE7. The pinouts for headers J1 and J2 are shown in Figure 6.

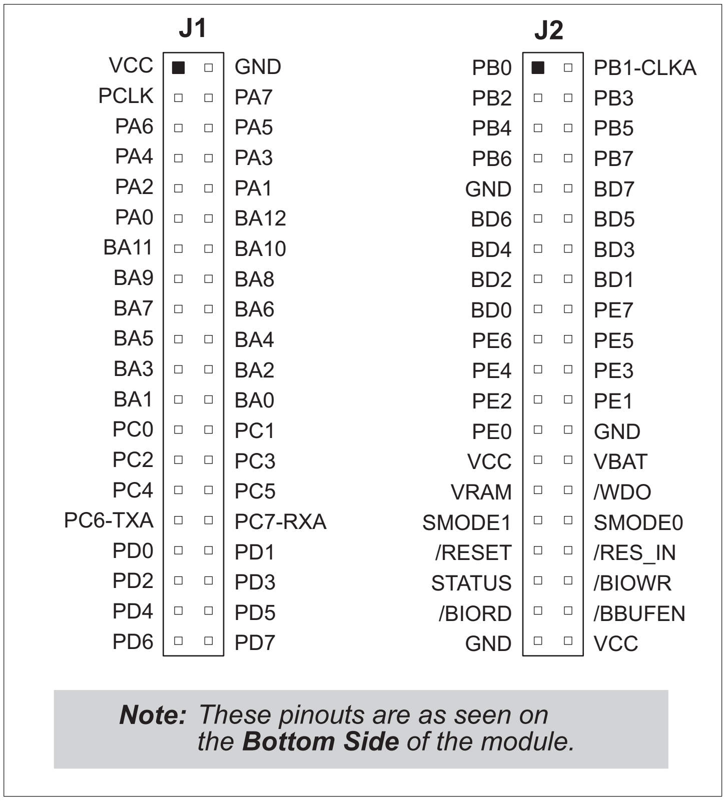


Figure 6. RCM2100 I/O Pinouts

The ports on the Rabbit 2000 microprocessor used in the RCM2100 are configurable, and so the factory defaults can be reconfigured. Table 2 lists the Rabbit 2000 factory defaults and the alternate configurations.

As shown in Table 2, pins PA0–PA7 can be used to allow the Rabbit 2000 to be a slave to another processor. PE0, PE1, PE4, and PE5 can be used as external interrupts INT0A, INT1A, INT0B, and INT1B. Pins PB0 and PB1 can be used to access the clock on Serial Port B and Serial Port A of the Rabbit microprocessor. Pins PD4 and PD6 can be programmed to be optional serial outputs for Serial Ports B and A. PD5 and PD7 can be used as alternate serial inputs by Serial Ports B and A.

The Ethernet-enabled versions of the RCM2100 do not have 0 Ω resistors (jumpers) installed at R21, R24, and R35–R38, which allows PE6, PE2, and PD4–PD7 to connect to the RealTek Ethernet chip that is stuffed on those versions.

5.2 I/O

The RCM2100 was designed to interface with other systems, and so there are no drivers written specifically for the I/O. The general Dynamic C read and write functions allow you to customize the parallel I/O to meet your specific needs. For example, use

```
WrPortI(PEDDR, &PEDDRShadow, 0x00);
```

to set all the port E bits as inputs, or use

```
WrPortI(PEDDR, &PEDDRShadow, 0xFF);
```

to set all the port E bits as outputs.

The sample programs in the Dynamic C **SAMPLES/RCM2100** directory provide further examples.

5.2.1 PCLK Output

The PCLK output is controlled by bits 7 and 6 of the Global Output Register (GOCR) on the Rabbit 2000 microprocessor, and so can be enabled or disabled in software. Starting with Dynamic C v 7.02, the PCLK output is disabled by default at compile time to minimize radiated emissions; the PCLK output is enabled in earlier versions of Dynamic C.

Use the following code to set the PCLK output as needed.

PCLK output driven with peripheral clock:

```
WrPortI(GOCR, &GOCRShadow, (GOCRShadow&~0xc0));
```

PCLK output driven with peripheral clock ÷ 2:

```
WrPortI(GOCR, &GOCRShadow, ((GOCRShadow&~0xc0)| 0x40));
```

PCLK output off (low):

```
WrPortI(GOCR, &GOCRShadow, ((GOCRShadow&~0xc0)| 0x80));
```

PCLK output on (high):

```
WrPortI(GOCR, &GOCRShadow, (GOCRShadow | 0xc0));
```

6.6 Placing Your Device on the Network

In many corporate settings, users are isolated from the Internet by a firewall and/or a proxy server. These devices attempt to secure the company from unauthorized network traffic, and usually work by disallowing traffic that did not originate from inside the network. If you want users on the Internet to communicate with your RCM2100, you have several options. You can either place the RCM2100 directly on the Internet with a real Internet address or place it behind the firewall. If you place the RCM2100 behind the firewall, you need to configure the firewall to translate and forward packets from the Internet to the RCM2100.

6.8.1 How to Set Up your Computer for Direct Connect

Follow these instructions to set up your PC or notebook. Check with your administrator if you are unable to change the settings as described here since you may need administrator privileges. The instructions are specifically for Windows 2000, but the interface is similar for other versions of Windows.

> **TIP:** If you are using a PC that is already on a network, you will disconnect the PC from that network to run these sample programs. Write down the existing settings before changing them to facilitate restoring them when you are finished with the sample programs and reconnect your PC to the network.

1. Go to the control panel (**Start > Settings > Control Panel**), and then double-click the Network icon.
2. Select the network interface card used for the Ethernet interface you intend to use (e.g., **TCP/IP Xircom Credit Card Network Adapter**) and click on the “Properties” button. Depending on which version of Windows your PC is running, you may have to select the “Local Area Connection” first, and then click on the “Properties” button to bring up the Ethernet interface dialog. Then “Configure” your interface card for a “10Base-T Half-Duplex” or an “Auto-Negotiation” connection on the “Advanced” tab.

   > **NOTE:** Your network interface card will likely have a different name.

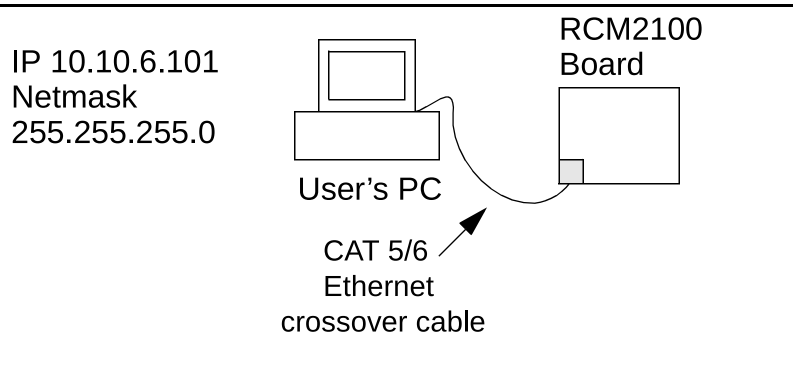
3. Now select the **IP Address** tab, and check **Specify an IP Address**, or select TCP/IP and click on “Properties” to assign an IP address to your computer (this will disable “obtain an IP address automatically”):

   IP Address : 10.10.6.101

   Netmask : 255.255.255.0

   Default gateway : 10.10.6.1

4. Click **<OK>** or **<Close>** to exit the various dialog boxes.



Direct Connection PC to RCM2100 Board

6.10.2 Sample Program: ETHCORE1.C

The RCM2100 modules with Ethernet ports can act as micro Web page servers, with dynamic interaction between the controller and the web pages. This sample program demonstrates how a Web page can be used to both monitor and control an RCM2100 module.

Compile & Run Program

Open the sample program **`ETHCORE1.C`**. Press **F9** to compile and run the program.

> **TIP:** This program will be more interesting to observe if LEDs DS4 and DS5 are installed on the Prototyping Board.

When the program starts, LEDs DS2, DS3 and DS5 will be lit, and DS4 will be dark. Open a web browser and enter the IP address you defined for the RCM2100 module in the program in the address window. A page like that shown in Figure 11 should appear.

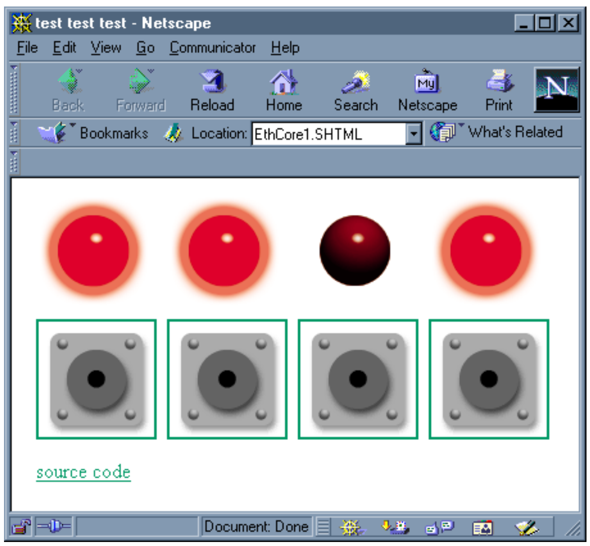


***Figure 11. Browser screen for Sample Program* `ETHCORE1.C`**

Clicking on each of the button images in the browser window will toggle the state of the associated LED image, and will toggle the state of the corresponding LED on the Prototyping Board. Since the web page is generated by the RabbitCore module (using Dynamic HTML), the LED image and the corresponding LED's real state will always be in step.

Program Description

This program begins to show the range of applications for an Ethernet-enabled embedded system controller, so let's look closely at its operation.

As with **`PINGLED.C`**, several network addresses must be defined before this application can work. The **`TCPCONFIG 1`** macro in the sample program provides default settings for **`MY_IP_ADDRESS`**, which is the address of the RCM2100 module, **`MY_NETMASK`**, and **`MY_GATEWAY`** (which needs to be defined if you wish to reach the system from outside the local network). These TCP/IP settings can be changed as needed in the **`TCP_CONFIG.LIB`** library.

Generally, the other defined values may be left at their default settings. If you are operating the system behind a firewall or proxy and need to specify a host port for redirection, you should comment out the line reading:

```
#define REDIRECTHOST MY_IP_ADDRESS
```

Then uncomment the next line, which defines a specific redirection host and port:

```
#define REDIRECTHOST "my host.com:8080"
```

Be sure to enter the host port where indicated by **`"my host.com:8080"`**.

This application creates dynamic HTML web pages on the fly. For simplicity, all of the Web page components—shell HTML, image GIFs, etc.—are imported into flash memory using the **`#ximport`** statements. It is also possible to read these files from other locations, including the onboard flash file system, but this application keeps things simple by loading all the components into working memory.

The program then defines four instances of an LED toggling function, which are basic CGI functions that swap the values "ledon.gif" and "ledoff.gif" as the contents of the **`led`*`n`*** strings, and then force a reload of the web page to change the associated LED image. The physical LEDs on the Prototyping Board are turned on or off to match the **`led`*`n`*** strings displayed on the Web page.

6.10.3 Additional Sample Programs

- **`ETHCORE2.C`**—This program takes anything that comes in on a port and sends it out Serial Port C. It uses SW2 as a signal that the connection should be closed, and PA0 as an indication that there is an open connection. You may change SW2 and PA0 to suit your application needs.

  Follow the instructions included with the sample program. Run the Telnet program on your PC (**Start > Run** **`telnet 10.10.6.100`**). As long as you have not modified the **`TCPCONFIG 1`** macro in the sample program, the IP address is 10.10.6.100 as shown; otherwise use the TCP/IP settings you entered in the **`TCP_CONFIG.LIB`** library. Each character you type will be printed in Dynamic C's **STDIO** window, indicating that the board is receiving the characters typed via TCP/IP.

- **`LEDCONSOLE.C`**—Demonstrates the features of **`ZCONSOLE.LIB`** command-oriented console library to control two LEDs on the Prototyping Board.

6.10.4 More Information

Refer to the ***Dynamic C TCP/IP User's Manual*** for complete details on the Dynamic C implementation of TCP/IP protocols.

APPENDIX A. RABBITCORE RCM2100 SPECIFICATIONS

Appendix A provides the specifications for the RCM2100, and describes the conformal coating.

A.1.1 Headers

The RCM2100 uses headers at J1, J2, and J3 for physical connection to other boards. J1 and J2 are 2 × 20 SMT headers with a 2 mm pin spacing. J3 is a 2 × 4 header with a 2 mm pin spacing.

Figure A-3 shows the layout of another board for the RCM2100 to be plugged in to. These reference design values are relative to the mounting hole or to the header connectors.

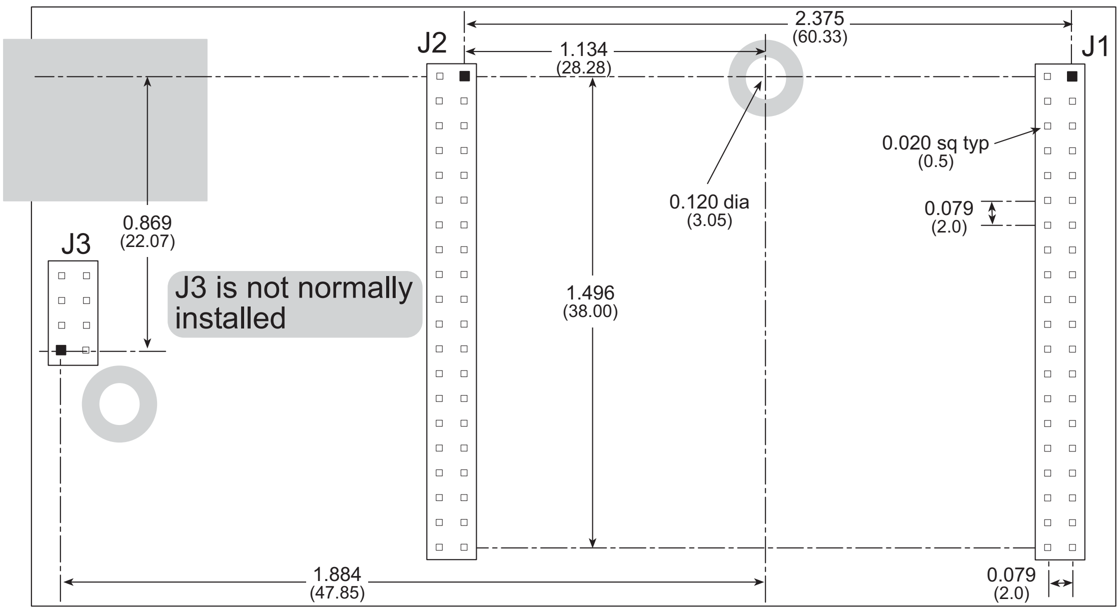


Figure A-3. User Board Footprint for the RCM2100

A.1.2 Physical Mounting

A 9/32” (7 mm) standoff with a 4-40 screw is recommended to attach the RCM2100 to a user board at the hole position shown in Figure A-3. A standoff with a screw may also be used at the hole position close to the RJ-45 Ethernet connector for a second anchor, or you may opt to have a nut and bolt with a wire at this hole position if you removed resistor R5 and elected to ground the RJ-45 Ethernet connector to the chassis.

A.4 I/O Buffer Sourcing and Sinking Limit

Unless otherwise specified, the Rabbit I/O buffers are capable of sourcing and sinking 8 mA of current per pin at full AC switching speed. Full AC switching assumes a 22.1 MHz CPU clock and capacitive loading on address and data lines of less than 100 pF per pin. Pins A0–A12 and D0–D7 are each rated at 8 mA. The absolute maximum operating voltage on all I/O is V_{DD} + 0.5 V, or 5.5 V.

Table A-5 shows the AC and DC output drive limits of the parallel I/O buffers when the Rabbit 2000 is used in the RCM2100.

Table A-5. I/O Buffer Sourcing and Sinking Capability

Pin Name	Output Drive Sourcing*/Sinking† Limits (mA)	
Output Port Name	Full AC Switching SRC/SNK	Maximum‡ DC Output Drive SRC/SNK
PA [7:0]	8/8	12/12
PB [7, 1, 0]	8/8	12/12
PC [6, 4, 2, 0]	8/8	12/12
PD [7:4]	8/8	12/12
PD [3:0]**	16/16	25/25
PE [7:0]	8/8	12/12

* The maximum DC sourcing current for I/O buffers between V_{DD} pins is 112 mA.

† The maximum DC sinking current for I/O buffers between V_{SS} pins is 150 mA.

‡ The maximum DC output drive on I/O buffers must be adjusted to take into consideration the current demands made my AC switching outputs, capacitive loading on switching outputs, and switching voltage.

The current drawn by all switching and nonswitching I/O must not exceed the limits specified in the first two footnotes.

** The combined sourcing from Port D [7:0] may need to be adjusted so as not to exceed the 112 mA sourcing limit requirement specified in the first footnote.

B.1.1 Prototyping Board Features

Power Connection. A 3-pin header is provided for connection of a power supply. Note that it is symmetrical, with both outer pins connected to ground and the center pin connected to the raw V+ input. The cable of the wall transformer provided with the North American version of the Development Kit ends in a connector that is correctly connected in either orientation.

Users providing their own power supply should ensure that it delivers 9–24 V DC at not less than 500 mA. The voltage regulator will get warm in use, but lower supply voltages will reduce thermal dissipation from the device.

Regulated Power Supply. The raw DC voltage provided at the POWER IN jack is routed to a 5 V linear voltage regulator, which provides stable power to the RCM2100 module and the Prototyping Board. A Shottky diode protects the power supply against damage from reversed raw power connections.

Power LED. The power LED lights whenever power is connected to the Prototyping Board.

Reset Switch. A momentary-contact, normally open switch is connected directly to the RCM2100's **/RES_IN** pin. Pressing the switch forces a hardware reset of the system.

I/O Switches & LEDs. Two momentary-contact, normally open switches are connected to the PB2 and PB3 pins of the RCM2100 module, and may be read as inputs by sample applications.

Two LEDs are connected to the PA0 and PA1 pins of the module, and may be driven as output indicators by sample applications. (Two more LEDs, driven by PA2 and PA3, may be added to the Prototyping Board for additional outputs.)

All the LEDs are connected through JP1, which has traces shorting adjacent pads together. These traces may be cut to disconnect the LEDs, and an 8-pin header soldered into JP1 to permit their selective reconnection with jumpers. See Figure B-4 for details.

Expansion Areas. The Prototyping Board is provided with several unpopulated areas for expansion of I/O and interfacing capabilities. See the next section for details.

Prototyping Area. A generous prototyping area has been provided for the installation of through-hole components. Vcc (5 V DC) and Ground buses run around the edge of this area. An area for surface-mount devices is provided to the right of the through-hole area. (Note that there are SMT device pads on both top and bottom of the Prototyping Board.)

To maximize the availability of RCM2100 resources, the demonstration hardware (LEDs and switches) on the Prototyping Board may be disconnected. This is done by cutting the traces below the silk-screen outline of header JP1 on the bottom side of the Prototyping Board. Figure B-4 shows the four places where cuts should be made. An exacto knife would work nicely to cut the traces. Alternatively, a small standard screwdriver may be carefully and forcefully used to wipe through the PCB traces.

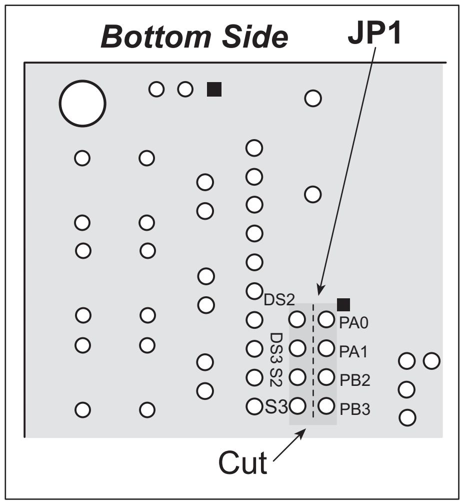


Figure B-4. Where to Cut Traces to Permanently Disable Demonstration Hardware on Prototyping Board

The power LED (PWR) and the RESET switch remain connected. Jumpers across the appropriate pins on header JP1 can be used to reconnect specific demonstration hardware later if needed.

Table B-2. Prototyping Board Jumper Settings

Header JP2	
Pins	**Description**
1–2	PA0 to LED DS2
3–4	PA1 to LED DS3
5–6	PB2 to Switch S2
7–8	PB3 to Switch S3

Note that the pinout at location JP1 on the bottom side of the Prototyping Board (shown in Figure B-4) is a mirror image of the top side pinout.

The Prototyping Board provides the user with RCM2100 connection points brought out conveniently to labeled points at headers J2 and J4 on the Prototyping Board. Small to medium circuits can be prototyped using point-to-point wiring with 20 to 30 AWG wire between the prototyping area and the holes at locations J2 and J4. The holes are spaced at 0.1" (2.5 mm),

B.4.1 Adding Other Components

There is room on the Prototyping Board for a user-supplied RS-232 transceiver chip at location U2 and a 10-pin header for serial interfacing to external devices at location J6. A Maxim MAX232 transceiver is recommended. When adding the MAX232 transceiver at position U2, you must also add 100 nF charge storage capacitors at positions C3–C6 as shown in Figure B-7.

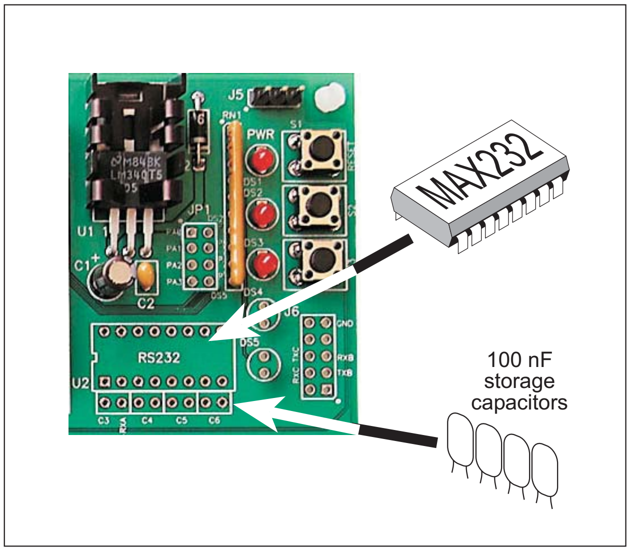


Figure B-7. Location for User-Supplied RS-232 Transceiver and Charge Storage Capacitors

There are two sets of pads that can be used for surface mount prototyping SOIC devices. The silk screen layout separates the rows into six 16-pin devices (three on each side). However, there are pads between the silk screen layouts giving the user two 52-pin (2×26) SOIC layouts with 50 mil pin spacing. There are six sets of pads that can be used for 3- to 6-pin SOT23 packages. There are also 60 sets of pads that can be used for SMT resistors and capacitors in an 0805 SMT package. Each component has every one of its pin pads connected to a hole in which a 30 AWG wire can be soldered (standard wire wrap wire can be soldered in for point-to-point wiring on the Prototyping Board). Because the traces are very thin, carefully determine which set of holes is connected to which surface-mount pad.

VRAM is also available on pin 29 of header J2 to facilitate battery backup of the external circuit. Note that the recommended maximum external current draw from VRAM is 100 µA, and new battery-life calculations should be done to take external loading into account.

C.1.2 Power to VRAM Switch

The VRAM switch, shown in Figure C-3, allows a customer-supplied external battery to provide power when the external power goes off. The switch provides an isolation between Vcc and the battery when Vcc goes low. This prevents the Vcc line from draining the battery.

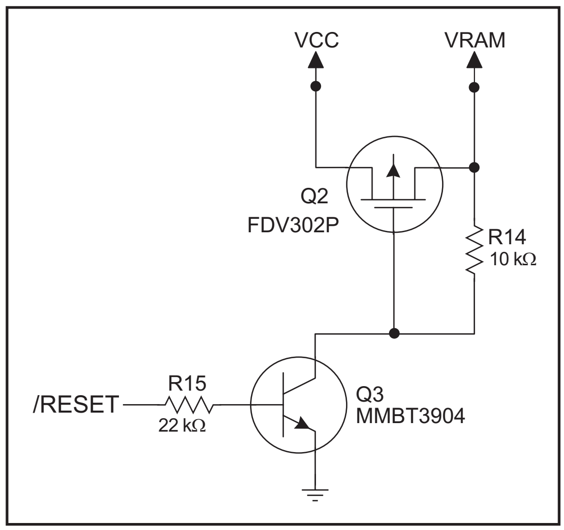


Figure C-3. VRAM Switch

Transistor Q2 is needed to provide a very small voltage drop between Vcc and VRAM (<100 mV, typically 10 mV) so that the processor lines powered by Vcc will not have a significantly different voltage than VRAM.

When the RCM2100 is not resetting (pin 2 on U3 is high), the /RESET line will be high. This turns on Q3, causing its collector to go low. This turns on Q2, allowing VRAM to nearly equal Vcc.

When the RCM2100 is resetting, the /RESET line will go low. This turns off Q2 and Q3, providing an isolation between Vcc and VRAM.

The battery-backup circuit keeps VRAM from dropping below 2 V.

C.1.3 Reset Generator

The RCM2100 uses a reset generator, U2, to reset the Rabbit 2000 microprocessor when the voltage drops below the voltage necessary for reliable operation. The reset occurs between 4.50 V and 4.75 V, typically 4.63 V. The RCM2100 has a reset output, pin 33 on header J2. The reset generator has a reset input, pin 34 on header J2, that can be used to force the RCM2100 to reset.

Appendix D. Sample Circuits

This appendix details several basic sample circuits that can be used with the RCM2100 modules.

- RS-232/RS-485 Serial Communication
- Keypad and LCD Connections
- External Memory
- D/A Converter

D.4 D/A Converter

The output will initially be 0 V to -10.05 V after the first inverting op-amp, and 0 V to +10.05 V after the second inverting op-amp. All lows produce 0 V out, FF produces 10 V out. The output can be scaled by changing the feedback resistors on the op-amps. For example, changing 5.11 kΩ to 2.5 kΩ will produce an output from 0 V to -5 V. Op-amps with a very low input offset voltage are recommended.

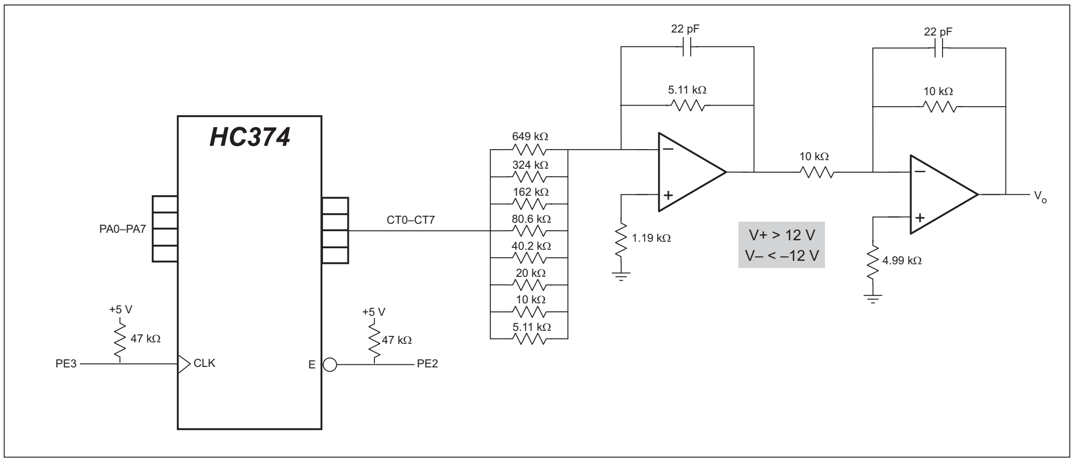


Figure D-5. Sample D/A Converter Connections