



RabbitCore RCM3000

C-Programmable Module with Ethernet

Getting Started Manual

019-0109 • 020415-A

RabbitCore RCM3000 Getting Started Manual

Part Number 019-0109 • 020415-A • Printed in U.S.A.

©2002 Z-World Inc. • All rights reserved.

Z-World reserves the right to make changes and improvements to its products without providing notice.

Trademarks

Rabbit 2000 is a trademark of Rabbit Semiconductor.

Dynamic C is a registered trademark of Z-World Inc.

Z-World, Inc.

2900 Spafford Street
Davis, California 95616-6800
USA

Telephone: (530) 757-3737
Fax: (530) 757-3792

www.zworld.com

Rabbit Semiconductor

2932 Spafford Street
Davis, California 95616-6800
USA

Telephone: (530) 757-8400
Fax: (530) 757-8402

www.rabbitsemiconductor.com

Table of Contents

Chapter 1: Introduction & Overview	1
1.1 Rabbit 3000 Microprocessor	1
1.2 RCM3000 Series RabbitCore Modules.....	2
1.2.1 Physical & Electrical Specifications	3
1.3 Development Software.....	4
1.4 How to Use This Manual	4
1.4.1 Additional Product Information	4
1.4.2 Additional Reference Information	4
1.4.3 Using Online Documentation	5
 Chapter 2: Hardware Setup	 7
2.1 Development Kit Contents.....	7
2.2 Prototyping Board	8
2.2.1 Prototyping Board Features	9
2.3 Development Hardware Connections	11
2.3.1 Attach Module to Prototyping Board	11
2.3.2 Connect Programming Cable	12
2.3.3 Connect Power	13
2.4 Run a Sample Program	14
2.5 Where Do I Go From Here?.....	14
2.5.1 Technical Support	14
 Chapter 3: Software Installation & Overview	 15
3.1 An Overview of Dynamic C	15
3.1.1 Hardware Requirements	16
3.2 Installing Dynamic C	17
3.2.1 Program & Documentation File Location	17
3.2.2 Installation Type	18
3.2.3 Select COM Port	19
3.2.4 Desktop Icons	19
3.3 Starting Dynamic C.....	20
3.3.1 Communication Error Messages	20
3.4 Sample Programs	21

Chapter 4: Using the TCP/IP Features	23
4.1 TCP/IP Connections	23
4.2 Running TCP/IP Sample Programs.....	25
4.3 IP Addresses Explained.....	27
4.4 How IP Addresses are Used	28
4.5 Dynamically Assigned Internet Addresses	29
4.6 Placing Your Device on the Internet.....	30
4.7 How to Set IP Addresses in the Sample Programs.....	31
4.8 How to Set Up your Computer's IP Address for Direct Connect	32
4.9 Run the PINGME.C Demo.....	33
4.10 Running More Demo Programs With Direct Connect	33
4.11 Where Do I Go From Here?.....	34
 Legal Notice	 35
 Index	 39
 Schematics	 41

1. INTRODUCTION & OVERVIEW

The Rabbit 3000 is a modern 8-bit microprocessor that is the central element of a complete and fully supported embedded design system that includes development tools, software libraries, core modules, sample designs, a parts store, and readily available expert, human support.

This Development Kit has the essentials that you need to design your own microprocessor-based system, and includes a complete Dynamic C software development system. This Development Kit contains a powerful RabbitCore module (the RCM3010) and Prototyping Board that will allow you to evaluate the Rabbit 3000 and to prototype circuits that interface to a Rabbit 3000 microprocessor. You will also be able to write and test software for the RCM3000 series RabbitCore modules, including TCP/IP applications.

1.1 Rabbit 3000 Microprocessor

The Rabbit 3000 microprocessor shares its instruction set and conceptual design with the successful Rabbit 2000. The instruction set is based on the Z80/Z180, but has been adapted to be C-friendly and to allow a megabyte of code space. Rabbit processors are fast with compact code.

The Rabbit 3000 has an extensive array of on-chip peripherals including 6 serial ports, 56 parallel I/O pins, motion control interfaces, a time/date clock, glueless memory and I/O interfacing, a slave interface, and in-circuit programming. Low EMI features including a clock spectrum spreader eliminate schedule-wrecking EMI problems.

Software development support is based on Z-World's Dynamic C, and includes extensive libraries for Internet connectivity.

Refer to the *Rabbit 3000 Microprocessor User's Manual* for complete information on the Rabbit 3000 microprocessor and complete specifications.

1.2 RCM3000 Series RabbitCore Modules

The RCM3000 series RabbitCore modules are designed for use on a customer-supplied motherboard that supplies power and interfaces to real-world I/O devices. Their two 34-pin connection headers provide 52 parallel user I/O lines, shared with five serial ports, along with control lines. A sixth serial port and one additional I/O line are available on the programming header.

A fully enabled slave port permits glueless master-slave interface with another Rabbit-based system. The slave port may also be used with non-Rabbit systems, although additional logic may be required.

The RCM3000 is equipped with a 10Base-T Ethernet port, 256K–512K flash memory and 128K–512K static RAM.

There are two production models in the RCM3000 series. If the standard models do not serve your needs, other variations can be specified and ordered in production quantities. Contact your Z-World or Rabbit Semiconductor sales representative for details.

Table 1 below highlights the differences between the two models in the RCM3000 family.

Table 1. RCM3000 Versions

Feature	RCM3000	RCM3010
Microprocessor	Rabbit 3000 running at 29.4 MHz	
Flash Memory	512K	256K
Static RAM	512K	128K
Serial Ports	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 clocked serial port is dedicated for programming	

NOTE: The RCM3010 is the RabbitCore module supplied with the Development Kit.

In addition, there is an RCM3100 series of RabbitCore modules that omits the RCM3000 series' Ethernet connectivity, but offer a much smaller footprint, which is one-half the size of the RCM3000 series.

The RabbitCore modules can be programed locally, remotely, or via a network.

1.2.1 Physical & Electrical Specifications

Table 2 lists the basic specifications for the RCM3000.

Table 2. RCM3000 Specifications

Specification	Data
Power Supply	3.15 – 3.45 V DC (145 mA at 29.5 MHz clock speed)
Size	1.85" x 2.65" x 0.86" (47 mm × 67 mm × 22 mm)
Environmental	–40°C to 70°C, 5–95% humidity, noncondensing

NOTE: For complete product specifications, see Appendix A in the *RabbitCore RCM3000 User's Manual*.

The RCM3000 modules have two 34-pin headers to which cables can be connected, or which can be plugged into matching sockets on a production device. The pinouts for these connectors are shown in Figure 1 below.

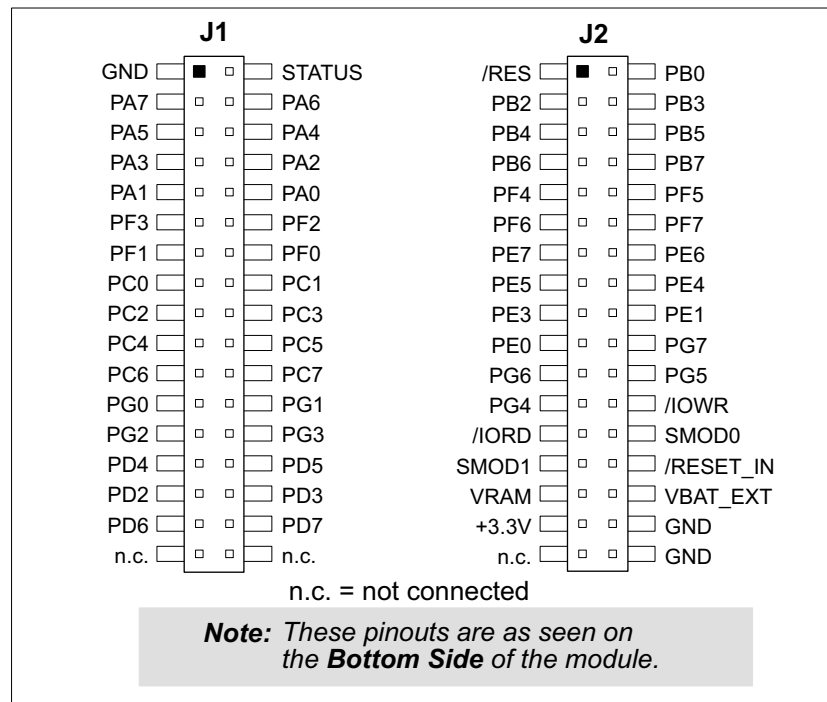


Figure 1. RCM3000 Connector Pinout

1.3 Development Software

The RCM3000 module uses the Dynamic C development environment for rapid creation and debugging of runtime applications. Dynamic C provides a complete development environment with integrated editor, compiler and source-level debugger. It interfaces directly with the target system, eliminating the need for complex and unreliable in-circuit emulators.

Dynamic C must be installed on a Windows workstation with at least one free serial USB or COM port for communication with the target system. See Chapter 3., “Software Installation & Overview.” for complete information on installing Dynamic C.

NOTE: The RCM3000 module requires Dynamic C v7.25 or later for development. A compatible version is included on the Development Kit CD-ROM.

1.4 How to Use This Manual

This *Getting Started* manual is intended to give users a quick but solid start with the RCM3000 module. It does not contain detailed information on the module hardware capabilities, the Dynamic C development environment, or the TCP/IP software support for the integrated Ethernet port. Most users will want more detailed information on some or all of these topics in order to put the RCM3000 module to effective use.

1.4.1 Additional Product Information

Detailed information about the RCM3000 will be found in the *RabbitCore RCM3000 User's Manual*, provided on the accompanying CD-ROM in both HTML and Adobe PDF format.

Some advanced users may choose to skip the rest of this introductory manual and proceed directly with the detailed hardware and software information in the *User's Manual*.

TIP: We recommend that anyone not thoroughly familiar with Z-World controllers at least read through the rest of this manual to gain the necessary familiarity to make use of the more advanced information.

1.4.2 Additional Reference Information

In addition to the product-specific information contained in the *RabbitCore RCM3000 User's Manual*, several higher level reference manuals are provided in HTML and PDF form on the accompanying CD-ROM. Advanced users will find these references valuable in developing systems based on the RCM3000 module:

- *Dynamic C Premier User's Manual*
- *An Introduction to TCP/IP*
- *Dynamic C TCP/IP User's Manual*
- *Rabbit 3000 Microprocessor User's Manual*

1.4.3 Using Online Documentation

We provide the bulk of our user and reference documentation in two electronic formats, HTML and Adobe PDF. We do this for several reasons.

We believe that providing all users with our complete library of product and reference manuals is a useful convenience. However, printed manuals are expensive to print, stock and ship. Rather than include and charge for manuals that every user may not want, or provide only product-specific manuals, we choose to provide our complete documentation and reference library in electronic form with every development kit and with our Dynamic C development environment.

NOTE: The most current version of Adobe Acrobat Reader can always be downloaded from Adobe's web site at <http://www.adobe.com>. We recommend that you use version 4.0 or later.

Providing this documentation in electronic form saves an enormous amount of paper by not printing copies of manuals that users don't need.

Finding Online Documents

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, create a new desktop icon that points to **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.

Printing Electronic Manuals

We recognize that many users prefer printed manuals for some uses. Users can easily print all or parts of those manuals provided in electronic form. The following guidelines may be helpful:

- Print from the Adobe PDF versions of the files, not the HTML versions.
- If your printer supports duplex printing, print pages double-sided.
- If you do not have a suitable printer or do not want to print the manual yourself, most retail copy shops (e.g. Kinkos, AlphaGraphics, etc.) will print the manual from the PDF file and bind it for a reasonable charge—about what we would have to charge for a printed and bound manual.

2. HARDWARE SETUP

This chapter describes the RCM3000 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 RCM3000 Development Kit. If you purchased an RCM3000 module by itself, you will have to adapt the information in this chapter and elsewhere to your test and development setup.

2.1 Development Kit Contents

The RCM3000 Development Kit contains the following items:

- RCM3010 module with Ethernet port, 256K flash memory, and 128K SRAM.
- RCM3000 Prototyping Board.
- AC adapter, 9 V DC, 1 A. (Included only with Development Kits sold for the North American market. A header plug leading to bare leads is provided to allow overseas users to connect a power supply compatible with their local mains power.)
- 10-pin header to DE9 programming cable with integrated level-matching circuitry.
- *Dynamic C SE* CD-ROM, with complete product documentation on disk.
- This *Getting Started* manual.
- A bag of accessory parts for use on the Prototyping Board.
- Registration card.

2.2 Prototyping Board

The Prototyping Board included in the Development Kit makes it easy to connect an RCM3000 module to a power supply and a PC workstation for development. It also provides some basic I/O peripherals (switches and LEDs), as well as a prototyping area for more advanced hardware development.

For the most basic level of evaluation and development, the Prototyping Board can be used without modification.

As you progress to more sophisticated experimentation and hardware development, modifications and additions can be made to the board without modifying or damaging the RCM3000 module itself.

The Prototyping Board is shown below in Figure 2, with its main features identified.

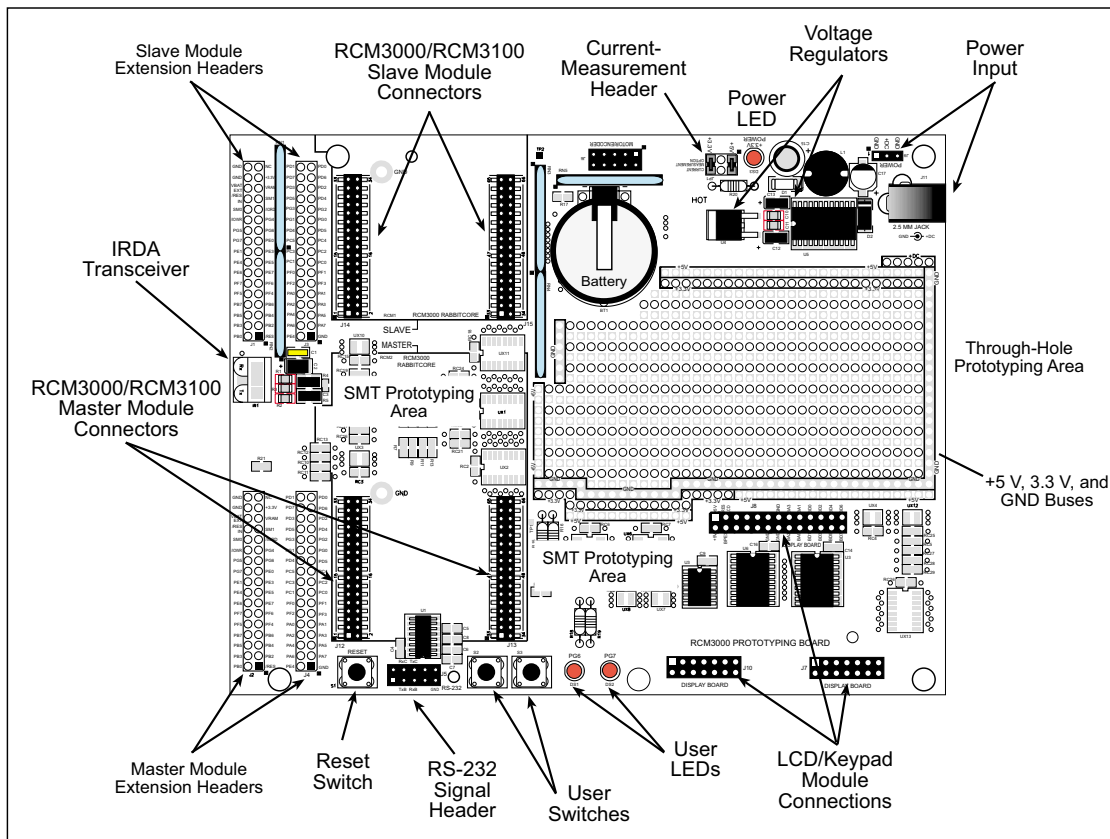


Figure 2. RCM3000/RCM3100 Prototyping Board

2.2.1 Prototyping Board Features

- **Power Connection**—A power-supply jack and a 3-pin header are provided for connection to the power supply. Note that the 3-pin header is symmetrical, with both outer pins connected to ground and the center pin connected to the raw V+ input. The cable of the AC adapter provided with the North American version of the Development Kit ends in a plug that connects to the power-supply jack. The header plug leading to bare leads provided for overseas customers can be connected to the 3-pin header in either orientation.

Users providing their own power supply should ensure that it delivers 8–24 V DC at 1 A. The voltage regulators will get warm while in use.

- **Regulated Power Supply**—The raw DC voltage provided at the POWER IN jack is routed to a 5 V switching voltage regulator, then to a separate 3.3 V linear regulator. The regulators provide stable power to the RCM3000 module and the Prototyping Board.
- **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 RCM3000's /RESET_IN pin. Pressing the switch forces a hardware reset of the system.
- **I/O Switches and LEDs**—Two momentary-contact, normally open switches are connected to the PG0 and PG1 pins of the master RCM3000 module and may be read as inputs by sample applications.

Two LEDs are connected to the PG6 and PG7 pins of the master module, and may be driven as output indicators by sample applications.

- **Prototyping Area**—A generous prototyping area has been provided for the installation of through-hole components. +3.3 V, +5 V, and Ground buses run around the edge of this area. Several areas for surface-mount devices are also available. (Note that there are SMT device pads on both top and bottom of the Prototyping Board.) Each SMT pad is connected to a hole designed to accept a 30 AWG solid wire.
- **Slave Module Connectors**—A second set of connectors is pre-wired to permit installation of a second, slave RCM3000 series or RCM3100 series module. This capability is reserved for future use, although the schematics in this manual contain all of the details an experienced developer will need to implement a master-slave system.
- **Module Extension Headers**—The complete pin sets of both the **MASTER** and **SLAVE** RabbitCore modules are duplicated at these two sets of headers. Developers can solder wires directly into the appropriate holes, or, for more flexible development, 26-pin header strips can be soldered into place. See Figure 1 for the header pinouts.
- **RS-232**—Two 3-wire or one 5-wire RS-232 serial port are available on the Prototyping Board. Refer to the Prototyping Board schematic (090-0137) for additional details.

A 10-pin 0.1-inch spacing header strip is installed at J5 to permit connection of a ribbon cable leading to a standard DE-9 serial connector.

- **Current Measurement Option**—Jumpers across pins 1–2 and 5–6 on header JP1 can be removed and replaced with an ammeter across the pins to measure the current drawn from the +5 V or the +3.3 V supplies, respectively.
- **Motor Encoder**—A motor/encoder header is provided at header J6 for future use.
- **LCD/Keypad Module**—Z-World’s LCD/keypad module (Z-World part number 101-0465) may be plugged in directly to headers J7, J8, and J10.

2.3 Development Hardware Connections

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

1. Attach the RCM3000 module to the Prototyping Board.
2. Connect the programming cable between the RCM3000 module and the workstation PC.
3. Connect the module's Ethernet port to a PC's Ethernet port, or to an Ethernet network.
4. Connect the power supply to the Prototyping Board.

2.3.1 Attach Module to Prototyping Board

Turn the RCM3000 series module so that the Ethernet connector end of the module extends off the Prototyping Board, as shown in Figure 3 below. Align the module headers J1 and J2 into sockets J21 and J13 on the Prototyping Board.

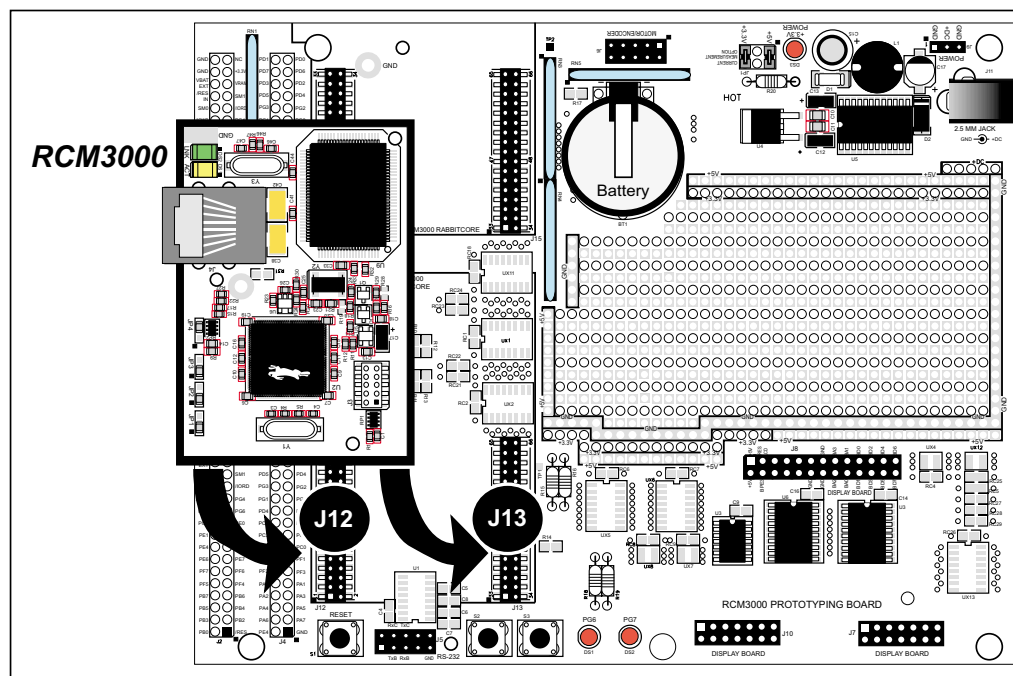


Figure 3. Installing the RCM3000 Series Module on the Prototyping Board

Although you can install a single module into either the **MASTER** or the **SLAVE** position on the Prototyping Board, all the Prototyping Board features (switches, LEDs, serial port drivers, etc.) are connected to the **MASTER** position. We recommend you install a single module in the **MASTER** position.

NOTE: It is important that you line up the pins on headers J1 and J2 of the RCM3000 series module exactly with the corresponding pins of headers J12 and J13 on the Prototyping Board. The header pins may become bent or damaged if the pin alignment is off-set, 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 firmly into the Prototyping Board headers.

2.3.2 Connect Programming Cable

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

Connect the 10-pin connector of the programming cable labeled **PROG** to header J1 on the RCM3000 series module as shown in Figure 4. 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 normal serial connection.)

NOTE: Be sure to use the programming cable supplied with this Development Kit—the programming cable has color shrink wrap around the RS-232 converter section located in the middle of the cable. Programming cables with clear shrink wrap from other Z-World or Rabbit Semiconductor kits were not designed to work with RCM3000 series modules.

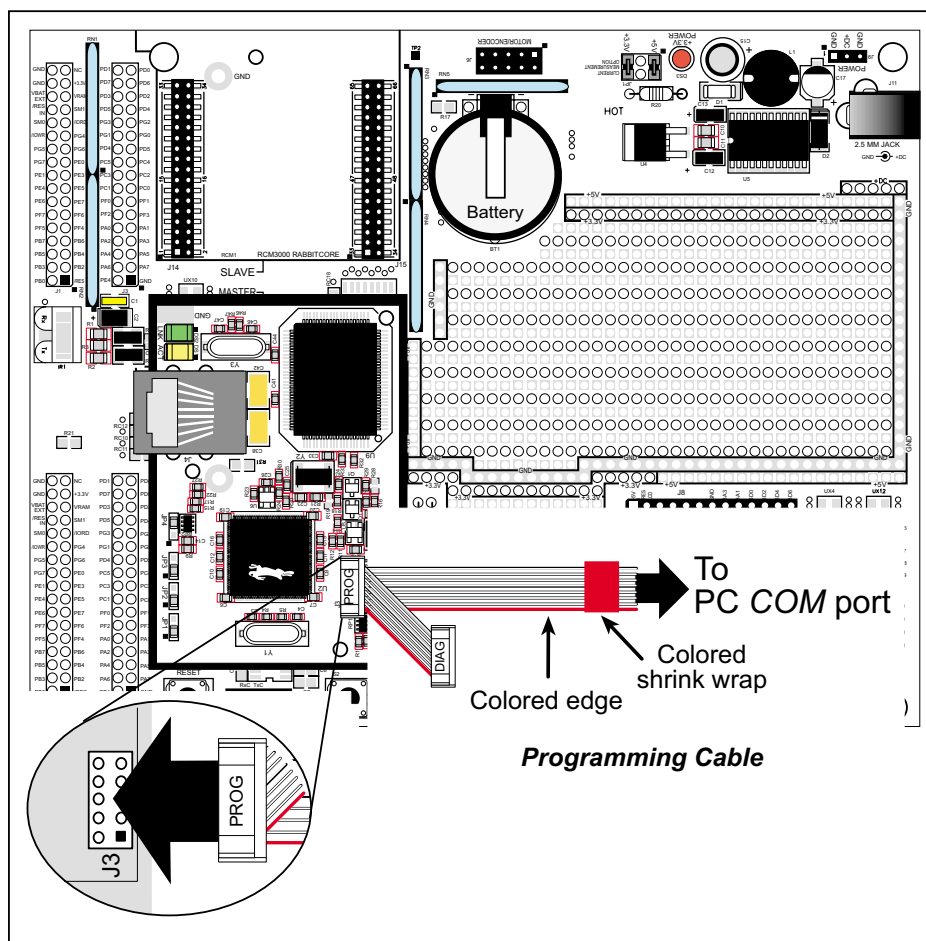


Figure 4. Connect Programming Cable to RCM3000

Connect the other end of the programming cable to a COM port on your PC.

NOTE: Some PCs now come equipped only with a USB port. It may be possible to use an RS-232/USB converter with the programming cable supplied with the RCM3000 series Development Kit. Contact Technical Support (see Section 2.5.1) for further assistance.

2.3.3 Connect Power

When all other connections have been made, you can connect power to the RCM3000 Prototyping Board.

Connect the wall transformer to jack J11 on the Prototyping Board as shown in Figure 6 below.

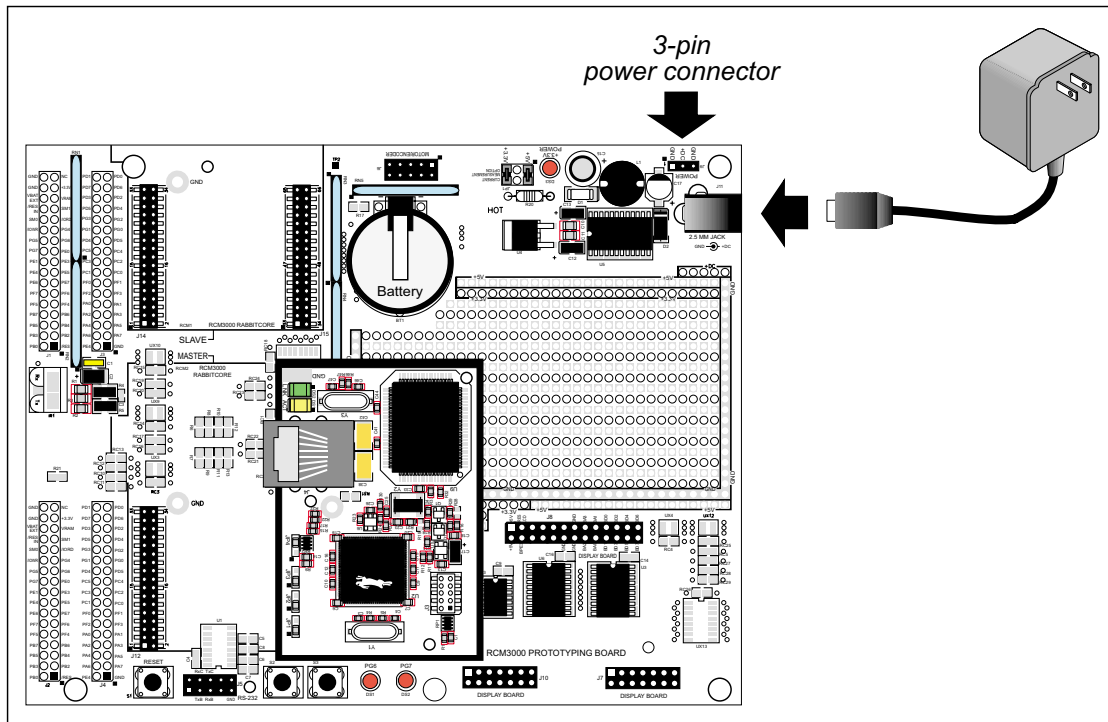


Figure 6. Power Supply Connections

Plug in the wall transformer. The power LED on the Prototyping Board should light up. The RCM3000 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 J11. You should disconnect power before making any circuit adjustments in the prototyping area, changing any connections to the board, or removing the RCM3000 from the Prototyping Board.

2.3.3.1 Overseas Development Kits

Development kits sold outside North America include a header connector that may be connected to 3-pin header J9 on the Prototyping Board. The connector may be attached either way as long as it is not offset to one side. The red and black wires from the connector can then be connected to the positive and negative connections on your power supply. The power supply should deliver 8 V–24 V DC at 1 A.

2.4 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.

Find the file **PONG.C**, which is in the Dynamic C **SAMPLES** folder. To run the program, open it with the **File** menu (if it is not still open), compile it using the **Compile** menu, and then run it by selecting **Run** in the **Run** menu. The **STDIO** window will open and will display a small square bouncing around in a box.

This program shows that the CPU is working. The sample program described in Section 4.9, “Run the PINGME.C Demo,” tests the TCP/IP portion of the board.

2.5 Where Do I Go From Here?

We recommend that you proceed to the next chapter and install Dynamic C (if you do not already have it installed), then run the **PONG.C** sample program to verify that the RCM3000 module and the Prototyping Board are set up and functioning correctly.

If everything appears to be working, we recommend the following sequence of action:

1. Run all of the sample programs described in Section 3.4 to get a basic familiarity with Dynamic C and the RCM3000 module’s capabilities.
2. For further development, refer to the *RabbitCore RCM3000 User’s Manual* for details of the module’s hardware and software components.

A documentation icon should have been installed on your workstation’s desktop; click on it to reach the documentation menu. You can create a new desktop icon that points to **default.htm** in the **docs** folder in the Dynamic C installation folder.

3. For advanced development topics, refer to the *Dynamic C Premier User’s Manual* and the *Dynamic C TCP/IP User’s Manual*, also in the online documentation set.

2.5.1 Technical Support

NOTE: If you purchased your RCM3000 through a distributor or through a Z-World or Rabbit Semiconductor partner, contact the distributor or partner first for technical support.

If there are any problems at this point:

- Check the Z-World/Rabbit Semiconductor Technical Bulletin Board at www.zworld.com/support/bb/.
- Use the Technical Support e-mail form at www.zworld.com/support/support_submit.html.

3. SOFTWARE INSTALLATION & OVERVIEW

To develop and debug programs for the RCM3000 (and for all other Z-World and Rabbit Semiconductor hardware), you must install and use Dynamic C. This chapter takes you through the installation of Dynamic C, and then provides a tour of its major features with respect to the RCM3000.

3.1 An Overview of Dynamic C

Dynamic C integrates the following development functions into one program:

- Editing
- Compiling
- Linking
- Loading
- In-Circuit Debugging

In fact, compiling, linking and loading are one function. Dynamic C does not use an In-Circuit Emulator; programs being developed are downloaded to and executed from the “target” system via an enhanced serial-port connection. Program development and debugging take place seamlessly across this connection, greatly speeding system development.

Other features of Dynamic C include:

- Dynamic C has an easy-to-use built-in text editor. Programs can be executed and debugged interactively at the source-code or machine-code level. Pull-down menus and keyboard shortcuts for most commands make Dynamic C easy to use.
- Dynamic C also supports assembly language programming. It is not necessary to leave C or the development system to write assembly language code. C and assembly language may be mixed together.
- Debugging under Dynamic C includes the ability to use `printf` commands, watch expressions, breakpoints and other advanced debugging features. Watch expressions can be used to compute C expressions involving the target’s program variables or functions. Watch expressions can be evaluated while stopped at a breakpoint or while the target is running its program.

- Dynamic C provides extensions to the C language (such as shared and protected variables, costatements and cofunctions) that support real-world embedded system development. Interrupt service routines may be written in C. Dynamic C supports cooperative and preemptive multi-tasking.
- Dynamic C comes with many function libraries, all in source code. These libraries support real-time programming, machine level I/O, and provide standard string and math functions.
- Dynamic C compiles directly to memory. Functions and libraries are compiled and linked and downloaded on-the-fly. On a fast PC, Dynamic C can load 30,000 bytes of code in 5 seconds at a baud rate of 115,200 bps.

3.1 Hardware Requirements

To install and run Dynamic C, your system must be running one of the following operating systems:

- Windows 95
- Windows 98
- Windows NT
- Windows Me
- Windows 2000
- Windows XP

3.1.1 Hardware Requirements

The PC on which you install Dynamic C for development of RCM3000-based systems should have the following hardware:

- A Pentium or later microprocessor
- 32 MB of RAM
- At least one free COM (serial) port for communication with the target systems
- A 10Base-T Ethernet network interface port
(optional if you will not be using the RCM3000's Ethernet port)
- A CD-ROM drive (for software installation)

3.2 Installing Dynamic C

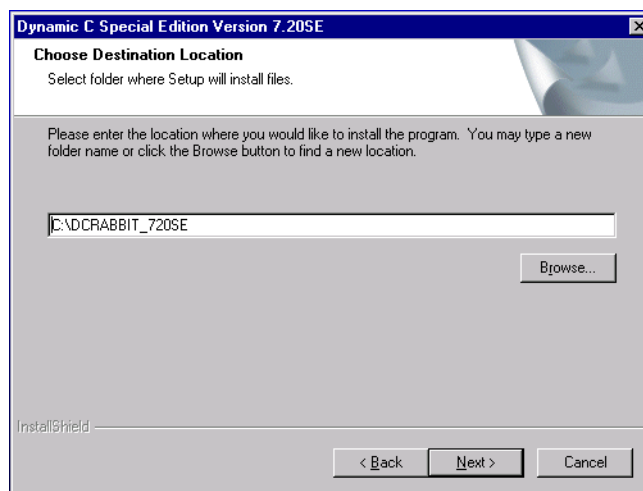
Insert the Dynamic C CD-ROM in the drive on your PC. If autorun is enabled, the CD installation will begin automatically.

If autorun is disabled or the installation otherwise 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 and not covered in this section. Selected steps that may be confusing to some users are outlined below. (Some of the installation utility screens may vary slightly from those shown.)

3.2.1 Program & Documentation File Location

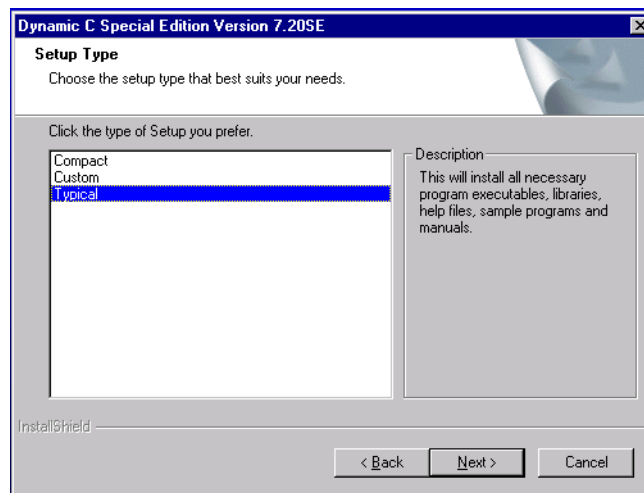
Dynamic C's application, library and documentation files can be installed in any convenient location on your workstation's hard drives.



The default location, as shown in the example above, is in a folder named for the version of Dynamic C, placed in the root folder of the C: drive. If this location is not suitable, enter a different root path before clicking **Next >**. Files are placed in the specified folder, so do not set this location to a drive's root directory.

3.2.2 Installation Type

Dynamic C has two components that can be installed together or separately. One component is Dynamic C itself, with the development environment, support files and libraries. The other component is the documentation library in HTML and PDF formats, which may be left uninstalled to save hard drive space or installed elsewhere (on a separate or network drive, for example).

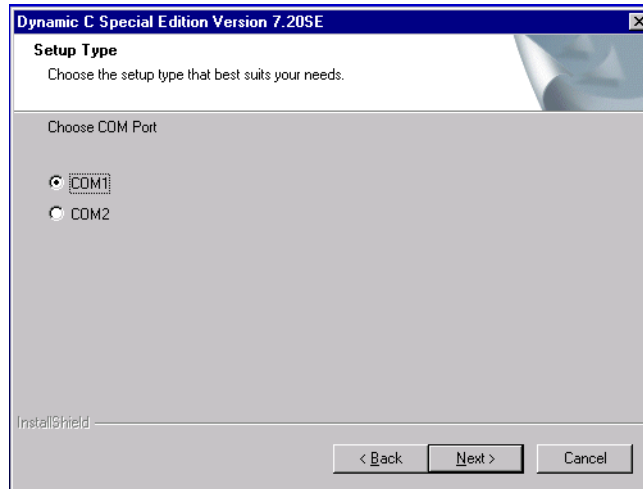


The installation type is selected in the installation menu shown above. The options are:

- **Typical Installation** — Both Dynamic C and the documentation library will be installed in the specified folder (default).
- **Compact Installation** — Only Dynamic C will be installed.
- **Custom Installation** — You will be allowed to choose which components are installed. This choice is useful to install or reinstall just the documentation.

3.2.3 Select COM Port

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, as shown in the example above, 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.

3.2.4 Desktop Icons

Once your installation is complete, you will have up to three icons on your PC desktop, as shown below.



One icon is for Dynamic C, one opens the documentation menu, and the third is for the Rabbit Field Utility, a tool used to download precompiled software to a target system.

3.3 Starting Dynamic C

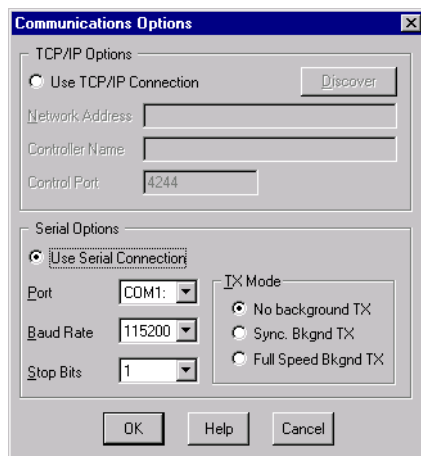
Once the RCM3000 is set up and connected as described in Chapter 2 and Dynamic C has been installed, start Dynamic C by double-clicking on the Dynamic C icon. Dynamic C should start, then look for the target system on the COM port you specified during installation (by default, COM1). Once detected, Dynamic C should go through a sequence of steps to cold-boot the module and compile the BIOS.

If you receive the message beginning **"BIOS successfully compiled"** you are ready to continue with the sample programs.

3.3.1 Communication Error Messages

If you receive the message **"No Rabbit Processor Detected,"** the programming cable may be connected to a different 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 and that the jumper across pins 5–6 of header JP1 on the Prototyping Board is installed. If the LED is lit, check both ends of the programming cable to ensure that it is firmly plugged into the PC and the RCM3000 series module's programming port. If you are using the Prototyping Board, ensure that the module is firmly and correctly installed in its connectors.

If there are no faults with the hardware, select a different COM port within Dynamic C. From the Options menu, select Communications. The dialog shown should appear.



Select another COM port from the list, then click OK. Press **<Ctrl-Y>** to force Dynamic C to recompile the BIOS. If Dynamic C still reports it is unable to locate the target system, repeat the above steps until you locate the active COM port.

If Dynamic C appears to compile the BIOS successfully, but you then receive a communication error message, it is possible that your PC cannot handle the 115,200 bps baud rate. Try changing the baud rate to 57,600 bps as follows.

- Locate the **Serial Options** dialog in the Dynamic C **Options > Communications** menu. Change the baud rate to 57,600 bps.

3.4 Sample Programs

To help familiarize you with the RCM3000 modules, Dynamic C includes several sample programs. Loading, executing and studying these programs will give you a solid hands-on overview of the RabbitCore's capabilities, as well as a quick start with Dynamic C as an application development tool.

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

Of the many sample programs included with Dynamic C, several are specific to the RCM3000. These programs will be found in the **samples/RCM3000** folder.

We suggest that you examine the following five of these sample programs in order to get a complete tour of the capabilities of the RCM3000 modules.

- **FLASHLED1.c**
- **FLASHLED2.c**
- **TOGGLESWITCH.c**
- **BROWSELED.c**
- **PINGLED.c**

Each of these programs is fully commented within the source code. Refer to these comments for the details of how each program works.

To run a program, open it with the **File** menu (if the sample program is not already open), compile it using the **Compile** menu, and then run it by selecting **Run** in the **Run** menu.

Once you have loaded and executed these five programs and have an understanding of how Dynamic C and the RCM3000 modules interact, you can move on and try the other sample programs, or begin building your own.

4. USING THE TCP/IP FEATURES

4.1 TCP/IP Connections

Programming and development can be done with the RCM3000 series RabbitCore modules without connecting the Ethernet port to a network. However, if you will be running the sample programs that use the Ethernet capability or will be doing Ethernet-enabled development, you should connect the RCM3000 module's Ethernet port at this time.

Before proceeding you will need to have the following items.

- If you don't have Ethernet access, you will need at least a 10Base-T Ethernet card (available from your favorite computer supplier) installed in a PC.
- Two RJ-45 straight through Ethernet cables and a hub, or an RJ-45 crossover Ethernet cable.

The Ethernet cables and Ethernet hub are available from Z-World in a TCP/IP tool kit. More information is available at www.zworld.com.

1. Connect the AC adapter and the programming cable as shown in Chapter 2, "Hardware Setup."
2. Ethernet Connections

There are four options for connecting the RCM3000 series module to a network for development and runtime purposes. The first two options permit total freedom of action in selecting network addresses and use of the "network," as no action can interfere with other users. We recommend one of these options for initial development.

- **No LAN** — The simplest alternative for desktop development. Connect the RCM3000's Ethernet port directly to the PC's network interface card using an RJ-45 *crossover cable*. A crossover cable is a special cable that flips some connections between the two connectors and permits direct connection of two client systems. A standard RJ-45 network cable will not work for this purpose.
- **Micro-LAN** — Another simple alternative for desktop development. Use a small Ethernet 10Base-T hub and connect both the PC's network interface card and the RCM3000's Ethernet port to it, using standard network cables.

The following options require more care in address selection and testing actions, as conflicts with other users, servers and systems can occur:

- **LAN** — Connect the RCM3000's Ethernet port to an existing LAN, preferably one to which the development PC is already connected. You will need to obtain IP addressing information from your network administrator.
- **WAN** — The RCM3000 is capable of direct connection to the Internet and other Wide Area Networks, but exceptional care should be used with IP address settings and all network-related programming and development. We recommend that development and debugging be done on a local network before connecting a RabbitCore system to the Internet.

TIP: Checking and debugging the initial setup on a micro-LAN is recommended before connecting the system to a LAN or WAN.

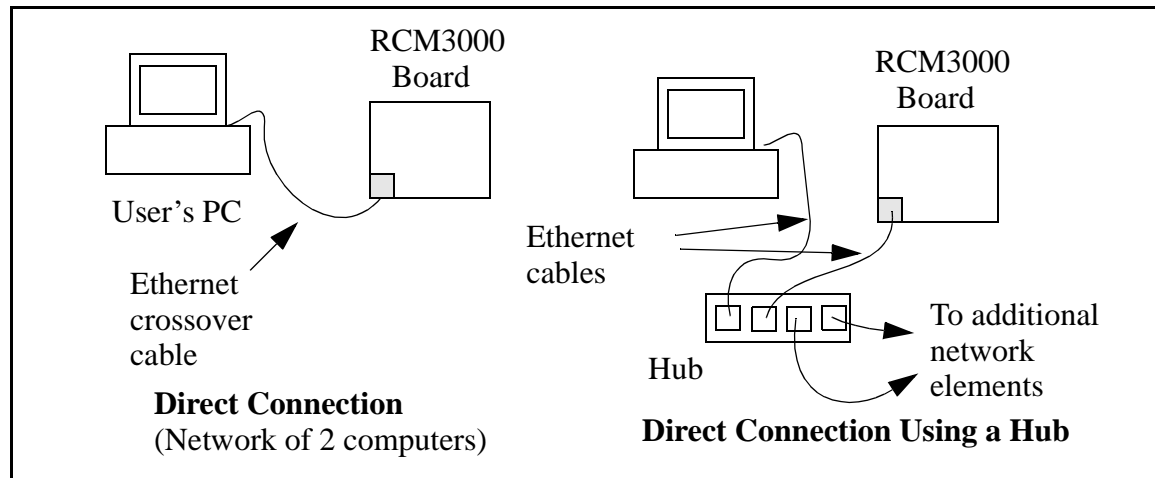
The PC running Dynamic C through the serial port on the RCM3000 does not need to be the PC with the Ethernet card.

3. Apply Power

Plug in the AC adapter. The RCM3000 series module is now ready to be used.

4.2 Running TCP/IP Sample Programs

We have provided a number of sample programs demonstrating various uses of TCP/IP for networking embedded systems. These programs require that the user connect his PC and the RCM3000 board together on the same network. This network can be a local private network (preferred for initial experimentation and debugging), or a connection via the Internet.



Obtaining IP addresses to interact over an existing, operating, network can involve a number of complications, and must usually be done with cooperation from your ISP and/or network systems administrator. For this reason, it is suggested that the user begin instead by using a direct connection between a PC and the RCM3000 board using an Ethernet crossover cable or a simple arrangement with a hub. (A crossover cable should not be confused with regular straight through cables.)

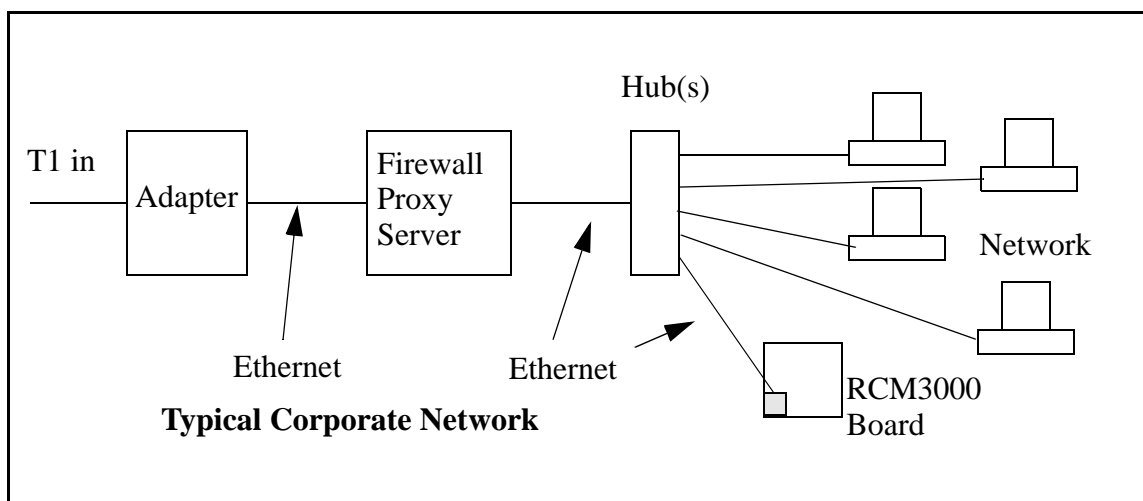
In order to set up this direct connection, the user will have to use a virgin PC (right out of the box), or disconnect a PC from the corporate network, or install a second Ethernet adapter and set up a separate private network attached to the second Ethernet adapter. Disconnecting your PC from the corporate network may be easy or nearly impossible, depending on how it is set up. If your PC boots from the network or is dependent on the network for some or all of its disks, then it probably should not be disconnected. If a second Ethernet adapter is used, be aware that Windows TCP/IP will send messages to one adapter or the other, depending on the IP address and the binding order in Microsoft products. Thus you should have different ranges of IP addresses on your private network from those used on the corporate network. If both networks service the same IP address, then Windows may send a packet intended for your private network to the corporate network. A similar situation will take place if you use a dial-up line to send a packet to the Internet. Windows may try to send it via the local Ethernet network if it is also valid for that network.

The following IP addresses are set aside for local networks and are not allowed on the Internet: 10.0.0.0 to 10.255.255.255, 172.16.0.0 to 172.31.255.255, and 192.168.0.0 to 192.168.255.255.

The RCM3000 board uses a 10Base-T type of Ethernet connection, which is the most common scheme. The RJ-45 connectors are similar to U.S. style telephone connectors, are except larger and have 8 contacts.

An alternative to the direct connection using a crossover cable is a direct connection using a hub. The hub relays packets received on any port to all of the ports on the hub. Hubs are low in cost and are readily available. The RCM3000 board uses 10 Mbps Ethernet, so the hub or Ethernet adapter must be either a 10 Mbps unit or a 10/100 unit that adapts to either 10 or 100 Mbps.

In a corporate setting where the Internet is brought in via a high-speed line, there are typically machines between the outside Internet and the internal network. These machines include a combination of proxy servers and firewalls that filter and multiplex Internet traffic. In the configuration below, the RCM3000 board could be given a fixed address so any of the computers on the local network would be able to contact it. It may be possible to configure the firewall or proxy server to allow hosts on the Internet to directly contact the controller, but it would probably be easier to place the controller directly on the external network outside of the firewall. This avoids some of the configuration complications by sacrificing some security.



If your system administrator can give you an Ethernet cable along with its IP address, the netmask and the gateway address, then you may be able to run the sample programs without having to setup a direct connection between your computer and the RCM3000 board. You will also need the IP address of the nameserver, the name or IP address of your mail server, and your domain name for some of the sample programs.

4.3 IP Addresses Explained

IP (Internet Protocol) addresses are expressed as 4 decimal numbers separated by periods, for example:

216.103.126.155

10.1.1.6

Each decimal number must be between 0 and 255. The total IP address is a 32-bit number consisting of the 4 bytes expressed as shown above. A local network uses a group of adjacent IP addresses. There are always 2^N IP addresses in a local network. The netmask (also called subnet mask) determines how many IP addresses belong to the local network. The netmask is also a 32-bit address expressed in the same form as the IP address. An example netmask is:

255.255.255.0

This netmask has 8 zero bits in the least significant portion, and this means that 2^8 addresses are a part of the local network. Applied to the IP address above (216.103.126.155), this netmask would indicate that the following IP addresses belong to the local network:

216.103.126.0

216.103.126.1

216.103.126.2

etc.

216.103.126.254

216.103.126.255

The lowest and highest address are reserved for special purposes. The lowest address (216.102.126.0) is used to identify the local network. The highest address (216.102.126.255) is used as a broadcast address. Usually one other address is used for the address of the gateway out of the network. This leaves $256 - 3 = 253$ available IP addresses for the example given.

4.4 How IP Addresses are Used

The actual hardware connection via an Ethernet uses Ethernet adapter addresses (also called MAC addresses). These are 48-bit addresses and are unique for every Ethernet adapter manufactured. In order to send a packet to another computer, given the IP address of the other computer, it is first determined if the packet needs to be sent directly to the other computer or to the gateway. In either case, there is an IP address on the local network to which the packet must be sent. A table is maintained to allow the protocol driver to determine the MAC address corresponding to a particular IP address. If the table is empty, the MAC address is determined by sending an Ethernet broadcast packet to all devices on the local network asking the device with the desired IP address to answer with its MAC address. In this way, the table entry can be filled in. If no device answers, then the device is nonexistent or inoperative, and the packet cannot be sent.

IP addresses are arbitrary and can be allocated as desired provided that they don't conflict with other IP addresses. However, if they are to be used with the Internet, then they must be numbers that are assigned to your connection by proper authorities, generally by delegation via your service provider.

Each RCM3000 series RabbitCore module has its own MAC address, which consists of the prefix 0090C2 followed by the code that appears on the label affixed to the RCM3000 series module. For example, a MAC address might be 0090C2C002C0.

TIP: You can always verify the MAC address on your board by running the sample program `DISPLAY_MAC.C` from the `SAMPLES\TCPIP` folder.

4.5 Dynamically Assigned Internet Addresses

In many instances, there are no fixed IP addresses. This is the case when, for example, you are assigned an IP address dynamically by your dial-up Internet service provider (ISP) or when you have a device that provides your IP addresses using the Dynamic Host Configuration Protocol (DHCP). The RCM3000 series RabbitCore modules can use such IP addresses to send and receive packets on the Internet, but you must take into account that this IP address may only be valid for the duration of the call or for a period of time, and could be a private IP address that is not directly accessible to others on the Internet. These private address can be used to perform some Internet tasks such as sending e-mail or browsing the Web, but usually cannot be used to participate in conversations that originate elsewhere on the Internet. If you want to find out this dynamically assigned IP address, under Windows 98 you can run the `wini`**pcfg** program while you are connected and look at the interface used to connect to the Internet.

Many networks use private IP addresses that are assigned using DHCP. When your computer comes up, and periodically after that, it requests its networking information from a DHCP server. The DHCP server may try to give you the same address each time, but a fixed IP address is usually not guaranteed.

If you are not concerned about accessing the RCM3000 from the Internet, you can place the RCM3000 on the internal network using a private address assigned either statically or through DHCP.

4.6 Placing Your Device on the Internet

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 RCM3000, you have several options. You can either place the RCM3000 directly on the Internet with a real Internet address or place it behind the firewall. If you place the RCM3000 behind the firewall, you need to configure the firewall to translate and forward packets from the Internet to the RCM3000.

4.7 How to Set IP Addresses in the Sample Programs

Most of the sample programs such as shown in the example below use macros to define the IP address assigned to the board and the IP address of the gateway, if there is a gateway.

```
#define MY_IP_ADDRESS "216.112.116.155"  
#define MY_NETMASK "255.255.255.248"  
#define MY_GATEWAY "216.112.116.153"
```

In order to do a direct connection, the following IP addresses can be used for the RCM3000:

```
#define MY_IP_ADDRESS "10.1.1.2"  
#define MY_NETMASK "255.255.255.248"  
// #define MY_GATEWAY "216.112.116.153"
```

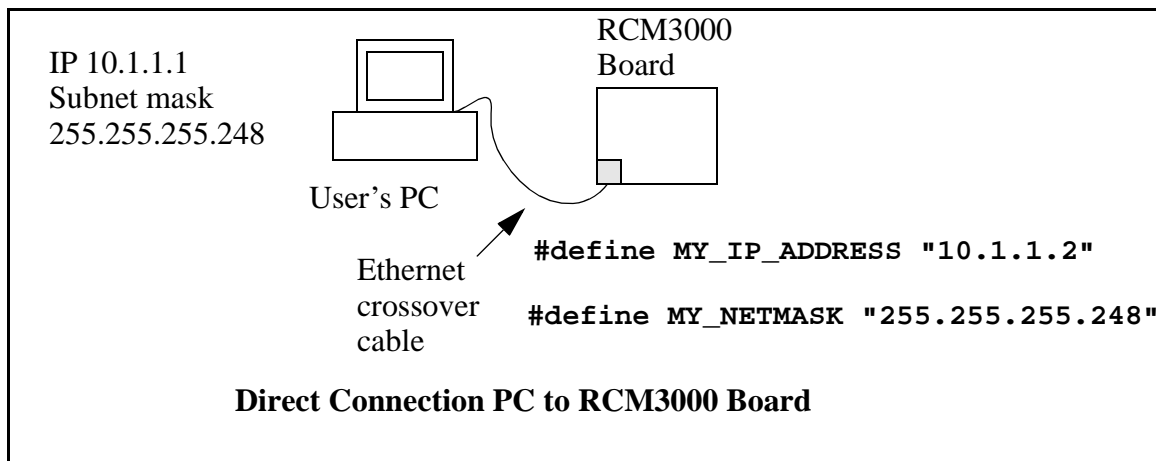
In this case, the gateway is not used and is commented out. The IP address of the board is defined to be 10.1.1.2. The IP address of you PC can be defined as 10.1.1.1.

4.8 How to Set Up your Computer's IP Address for Direct Connect

When your computer is connected directly to the RCM3000 board via an Ethernet connection, you need to assign an IP address to your computer. To assign the PC the address 10.1.1.1 with the subnetmask 255.255.255.248 under Windows 98, do the following.

Click on **Start > Settings > Control Panel** to bring up the Control Panel, and then double-click the Network icon. In the window find the line of the form **TCP/IP > Ethernet adapter name**. Double-click on this line to bring up the TCP/IP properties dialog box. You can edit the IP address directly and the subnet mask. (Disable "obtain an IP address automatically.") You may want to write down the existing values in case you have to restore them later. It is not necessary to edit the gateway address since the gateway is not used with direct connect.

The method of setting the IP address may differ for different versions of Windows, such as 95, NT, or 2000.



4.9 Run the PINGME.C Demo

In order to run this program, edit the IP address and netmask in the **PINGME.C** program (**SAMPLES\TCPIP\ICMP**) to the values 10.1.1.2 and 255.255.255.248 given above for a direct connection. Compile the program and start it running under Dynamic C. The crossover cable is connected from your computer's Ethernet adapter to the RCM3000 board's RJ-45 Ethernet connector. When the program starts running, the green **LNK** light on the RCM3000 board should be on to indicate an Ethernet connection is made. (Note: If the **LNK** light does not light, you may not have a crossover cable, or if you are using a hub perhaps the power is off on the hub.)

The next step is to ping the board from your PC. This can be done by bringing up the MS-DOS window and running the pingme program:

```
ping 10.1.1.2
```

or by **Start > Run**

and typing the entry

```
ping 10.1.1.2
```

Notice that the red **ACT** light flashes on the RCM3000 board while the ping is taking place, and indicates the transfer of data. The ping routine will ping the board four times and write a summary message on the screen describing the operation.

4.10 Running More Demo Programs With Direct Connect

The program **STATIC.C** (**SAMPLES\TCPIP\HTTP**) demonstrates how to make the RCM3000 board be a Web server. In order to run these sample programs, edit the IP address as for the pingme program, compile the program and start it executing. Then bring up your Web browser and enter the following server address: <http://10.1.1.2>.

This should bring up the Web page served by the sample program.

The sample program **RXSAMPLE.C** (**SAMPLES\TELNET**) allows you to communicate with the RCM3000 board using the Telnet protocol. To run this program, edit the IP address, compile the program, and start it running. Run the Telnet program on your PC (**Start > Run telnet 10.1.1.2**). 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.

4.11 Where Do I Go From Here?

NOTE: If you purchased your RCM3000 through a distributor or through a Z-World or Rabbit Semiconductor partner, contact the distributor or Z-World partner first for technical support.

If there are any problems at this point:

- Check the Z-World/Rabbit Semiconductor Technical Bulletin Board at www.zworld.com/support/bb/.
- Use the Technical Support e-mail form at www.zworld.com/support/support_submit.html.

If the sample programs ran fine, you are now ready to go on.

Additional sample programs are described in the *Dynamic C TCP/IP User's Manual*.

Please refer to the *Dynamic C TCP/IP User's Manual* to develop your own applications. *An Introduction to TCP/IP* provides background information on TCP/IP, and is available on the CD and on [Z-World's Web site](#).



LEGAL NOTICE

ZWORLD PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE-SUPPORT DEVICES OR SYSTEMS UNLESS A SPECIFIC WRITTEN AGREEMENT REGARDING SUCH INTENDED USE IS ENTERED INTO BETWEEN THE CUSTOMER AND Z-WORLD PRIOR TO USE. Life-support devices or systems are devices or systems intended for surgical implantation into the body or to sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling and user's manual, can be reasonably expected to result in significant injury.

No complex software or hardware system is perfect. Bugs are always present in a system of any size. In order to prevent danger to life or property, it is the responsibility of the system designer to incorporate redundant protective mechanisms appropriate to the risk involved.

All Z-World products are 100 percent functionally tested. Additional testing may include visual quality control inspections or mechanical defects analyzer inspections. Specifications are based on characterization of tested sample units rather than testing over temperature and voltage of each unit. Z-World products may qualify components to operate within a range of parameters that is different from the manufacturer's recommended range. This strategy is believed to be more economical and effective. Additional testing or burn-in of an individual unit is available by special arrangement.

INDEX

A

additional information
 online documentation 5
 references 4

C

C language 15, 16

D

Development Kit 7
Dynamic C 4, 15, 16
 assembly language 15
 debugger 15
 debugging 15
 desktop icons 19
 editor 15
 features 15
 handling communication error
 messages 20
 hardware requirements 16
 installing 17, 18, 19
 interrupt service routines .. 16
 sample programs 21
 starting 20
 watch expressions 15

E

Ethernet cables 23
Ethernet connections 23, 26
 10Base-T 26
 10Base-T Ethernet card 23
 additional resources 34
 Ethernet cables 26
 Ethernet hub 23
 IP addresses 25, 27
 steps 23

F

features
 Prototyping Board 8, 9
 RCM3000 2

H

hardware connections 11
 install RCM3000 on Prototyp-
 ing Board 11
 power supply 13
 programming cable 12
hardware reset 13

I

IP addresses 27, 32
 how to set 31
 how to set PC IP address ... 32

M

MAC addresses 28
models
 factory versions 2

P

pinout
 RCM3000 3
power supply
 connections 13
programming cable
 RCM3000 connections 12
Prototyping Board 8
 expansion area 9
 features 8, 9
 mounting RCM3000 11

R

RCM3000
 mounting on Prototyping
 Board 11
reset 13
running TCP/IP sample pro-
 grams 25

S

sample programs 21
 getting to know the RCM3000
 BROWSELED.C 21
 FLASHLED1.C 21
 FLASHLED2.C 21
 PINGLED.C 21
 TOGGLESWITCH.C 21
 how to set IP address 31
 PONG.C 14
 TCP/IP 25
 DISPLAY_MAC.C 28
 PINGME.C 33
 running TCP/IP sample pro-
 grams 25
 RXSAMPLE.C 33
 STATIC.C 33
software
 sample programs 21
specifications
 physical and electrical 3

T

technical support 14



SCHEMATICS

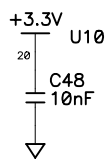
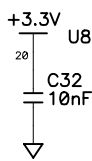
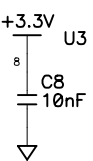
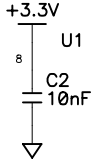
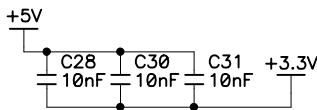
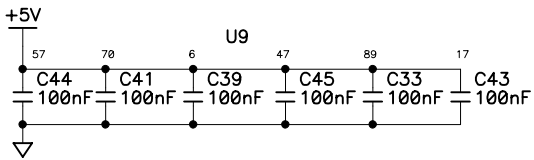
090-0136 RCM3000 Schematic

090-0137 RCM3000/RCM3100 Prototyping Board Schematic

090-0128 Programming Cable Schematic

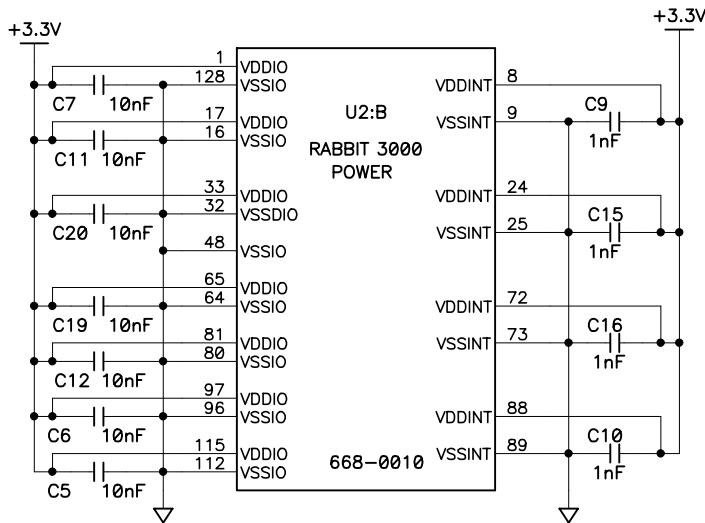
POWER TABLE

Ref Des	Device	GND	+3.3V	+5V	VRAM
U1	FLASH	24	8		
U3	FLASH	24	8		
U4	SRAM	24			8
U8	74LCX245	10	20		
U9	RTL8019AS	86,83,52,44,28,14		89,70,57,47,17,6	
U10	74LCX245	10	20		



STUFFING TABLE

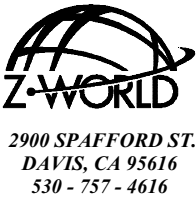
CIRCUIT	PART	RCM3000	RCM3010
SRAM	U4	512K	128K
SRAM select	JP4	2-3	1-2
FLASH 1	U1	256K	256K
FLASH 1 select	JP2	1-2	1-2
FLASH 2	U3	256K	NOT INSTALLED
FLASH 2 select	JP3	1-2	
BANK SELECT	JP1	1-2	1-2
EXT DATA BUS	JP5	2-3	2-3
POWER TO VRAM	R19	NOT INSTALLED	NOT INSTALLED
+5V_ OUT	R29	NOT INSTALLED	NOT INSTALLED
+5V_ IN	R28	NOT INSTALLED	NOT INSTALLED
PORT D OPTION	R15, R17, R22, R27	INSTALLED	INSTALLED
FUTURE OPTION	R38, R39, R40, R41, R42, R43	NOT INSTALLED	NOT INSTALLED
CHARGE PUMP	U7 C27, C36	INSTALLED	INSTALLED
INTERRUPT OPTION	R35	NOT INSTALLED	NOT INSTALLED

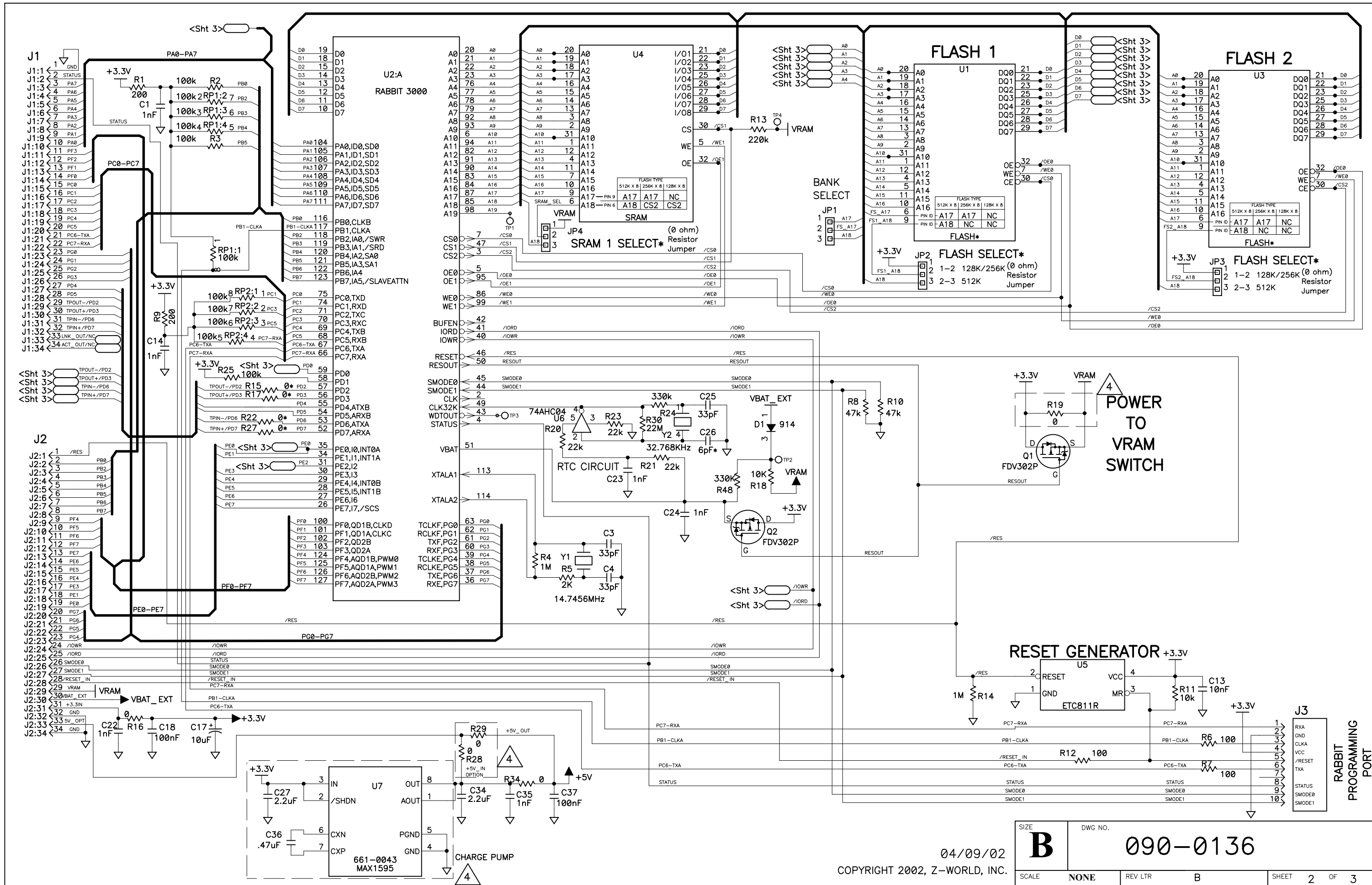


- NOTES: UNLESS OTHERWISE SPECIFIED;
- 1. ALL RESISTOR VALUES ARE IN OHMS, 1/10W, 5%
 - 2. ALL CAPACITORS ARE 50VDC OR HIGHER.
 - 3. THE ORIGINATION SOURCE OF A VOLTAGE IS REPRESENTED BY (V_{CC}), AND ALL REFERENCES TO THAT VOLTAGE ARE REPRESENTED BY (V_{CC}).
 - 4. OUTLINED CIRCUIT MAY NOT BE STUFFED DEPENDING ON MODEL, SEE STUFFING CHART FOR CLARIFICATION.
 - 5. COMPONENT VALUES SHOWN WITH AN ASTERISK (*) FOLLOWING THE VALUE, MAY HAVE DIFFERENT VALUES, OR MAY NOT BE STUFFED DEPENDING ON MODEL. SEE STUFFING CHART FOR CLARIFICATION.

COPYRIGHT 2002, Z-WORLD, INC.

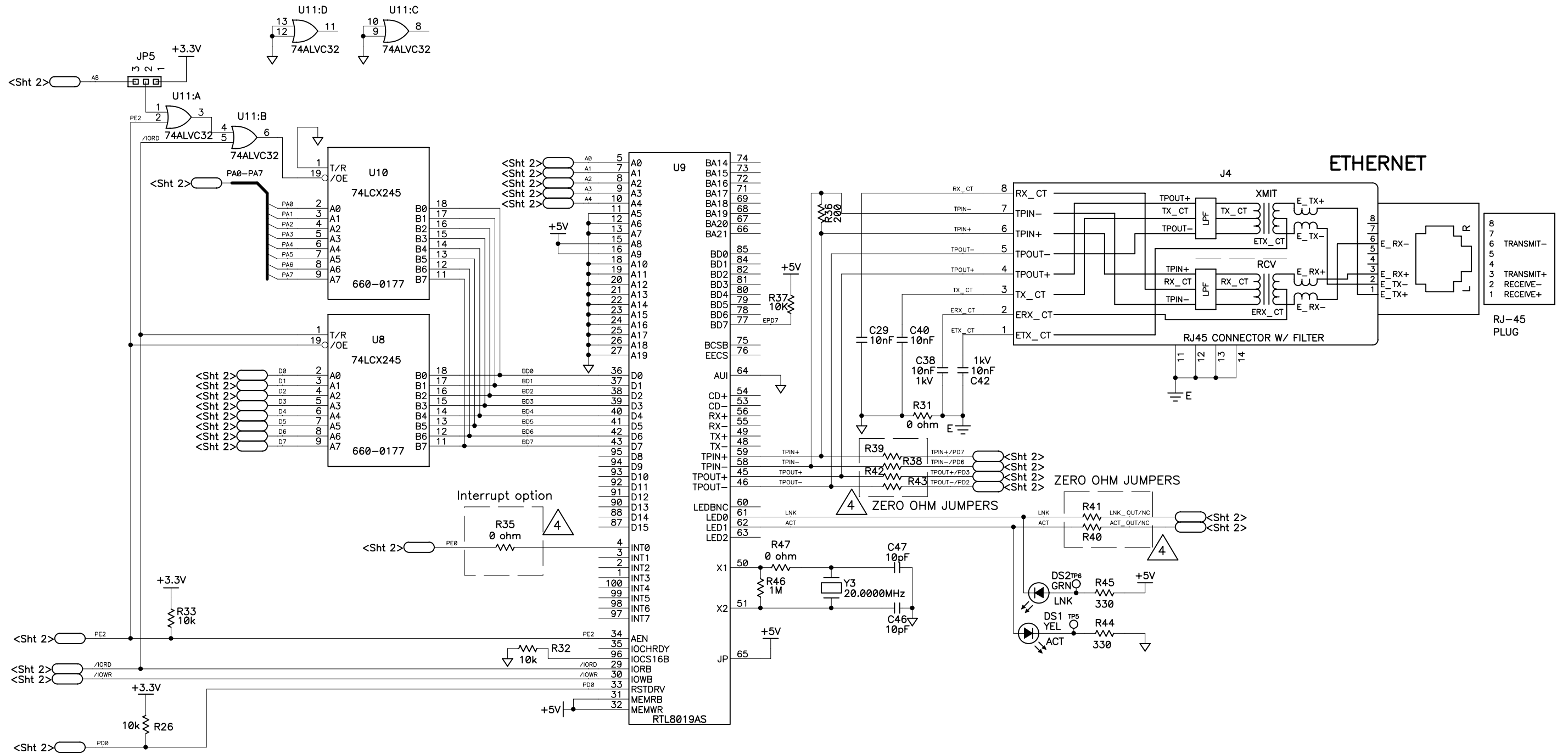
APPEND THE FOLLOWING DOCUMENTS WHEN CHANGING THIS DOCUMENT:		DRAWING CONTENT:		TITLE	
		DRAWN BY: (INITIAL RELEASE)		SCHEMATIC DIAGRAM RCM3000 SERIES RABBITCORE	
		DARREN MUSGROVE	03/15/02		
		REVISED BY:			
		KIS / D.MUSGROVE	04/09/02		
		APPROVALS: INITIAL RELEASE		SIZE B	
		PROJECT ENGINEER:			
		D.MUSGROVE	3/19/02		
		ENGINEERING MANAGER:			
		R.MATTHEWS	3/19/02	DWG NO. 090-0136	
		SIGNATURES	DATE		
		SCALE	NONE	RELEASE DATE	3/19/02
				SHEET	1 OF 3





04/09/02
COPYRIGHT 2002, Z-WORLD, INC.

SIZE	DWG NO.	REV	LTR	SHEET	OF
B	090-0136	NONE	B	2	3

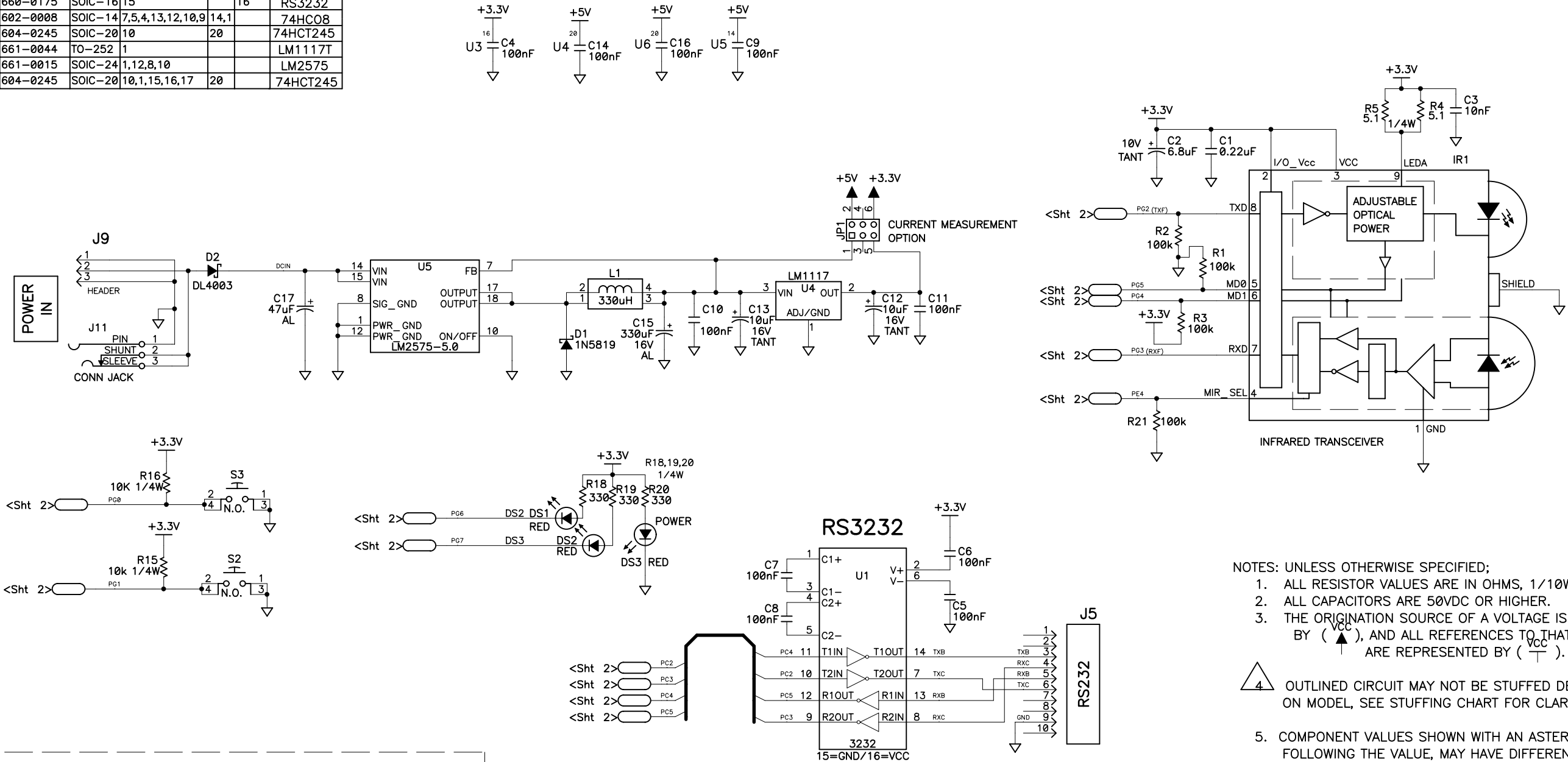


COPYRIGHT 2002, Z-WORLD, INC.
04/09/02

SIZE B	DWG NO. 090-0136
SCALE NONE	REV LTR B
SHEET 3	OF 3

Power Table						
Ref Des	Device(Type)	Package	GND	+5V	+3.3V	Part
U1	660-0175	SOIC-16	15		16	RS3232
U2	602-0008	SOIC-14	7,5,4,13,12,10,9	14,1		74HC08
U3	604-0245	SOIC-20	10	20		74HCT245
U4	661-0044	TO-252	1			LM1117T
U5	661-0015	SOIC-24	1,12,8,10			LM2575
U6	604-0245	SOIC-20	10,1,15,16,17	20		74HCT245

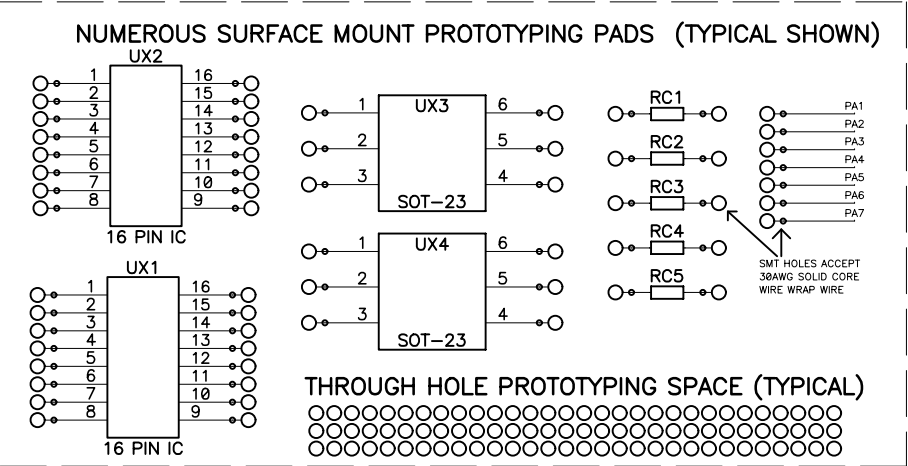
REVISION HISTORY			REVISION APPROVAL			
REV	ECO	DESCRIPTION	PROJECT ENGINEER	APPROVAL DATE	DOCUMENT CONTROL	APPROVAL DATE
A	E11864	INITIAL RELEASE				




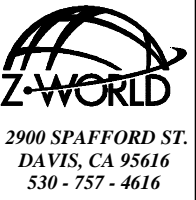
NOTES: UNLESS OTHERWISE SPECIFIED;

1. ALL RESISTOR VALUES ARE IN OHMS, 1/10W, 5%.
2. ALL CAPACITORS ARE 50VDC OR HIGHER.
3. THE ORIGIN SOURCE OF A VOLTAGE IS REPRESENTED BY (\uparrow), AND ALL REFERENCES TO THAT VOLTAGE ARE REPRESENTED BY (\uparrow).
4. OUTLINED CIRCUIT MAY NOT BE STUFFED DEPENDING ON MODEL, SEE STUFFING CHART FOR CLARIFICATION.
5. COMPONENT VALUES SHOWN WITH AN ASTERISK (*) FOLLOWING THE VALUE, MAY HAVE DIFFERENT VALUES, OR MAY NOT BE STUFFED DEPENDING ON MODEL.

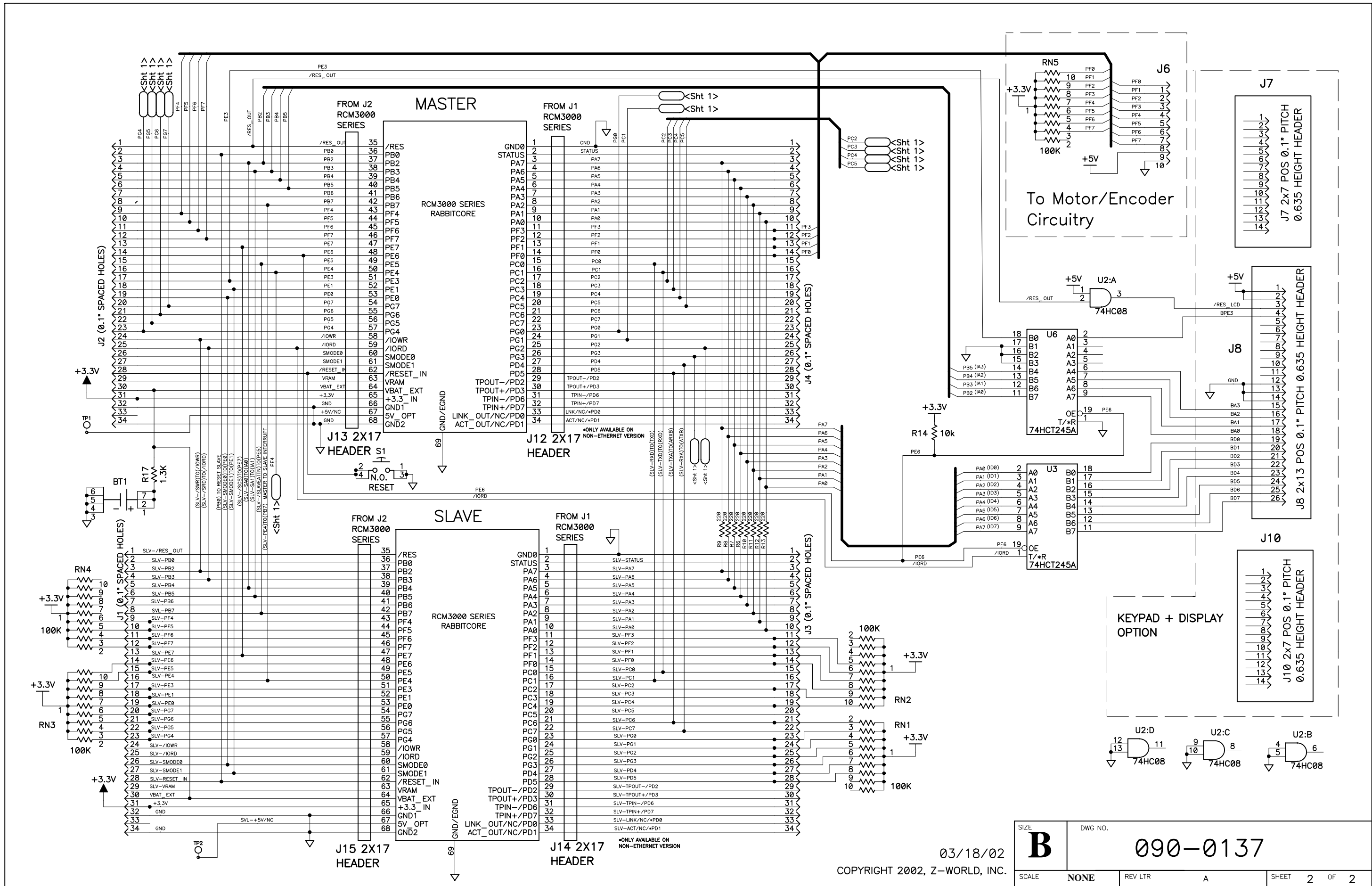
COPYRIGHT 2002, Z-WORLD, INC.



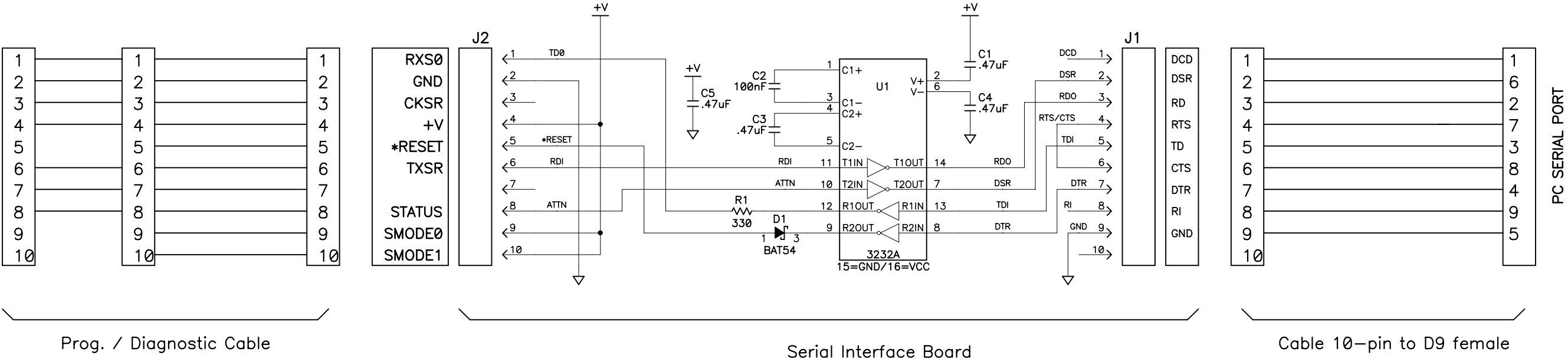
APPEND THE FOLLOWING DOCUMENTS WHEN CHANGING THIS DOCUMENT:		DRAWING CONTENT:		TITLE SCHEMATIC DIAGRAM RCM3000 PROTO BOARD		 2900 SPAFFORD ST. DAVIS, CA 95616 530 - 757 - 4616								
		DRAWN BY: (INITIAL RELEASE) D.MUSGROVE	03/15/02											
		REVISED BY: K.SCHALLER	03/18/02	APPROVALS: INITIAL RELEASE										
		PROJECT ENGINEER:		ENGINEERING MANAGER:										
		SIGNATURES		DATE		SCALE		NONE	RELEASE DATE	TBD	SHEET	1	OF	2




SIZE **B** DWG NO. **090-0137**



REVISION HISTORY			REVISION APPROVAL			
REV	ECO	DESCRIPTION	PROJECT ENGINEER	APPROVAL DATE	DOCUMENT CONTROL	APPROVAL DATE
A	E11523	INITIAL RELEASE OF SCHEMATIC	EP	5/14/01	KIS	5/14/01
B	E11691	CORRECT DE9 PINOUT	EP	10/5/01	KIS	10/4/01
C	E11816	ADD 3.3V CAPABILITY AND RED TUBING	DM	1/24/02	KLS	1/24/02



NOTES: UNLESS OTHERWISE SPECIFIED;
1. ALL RESISTOR VALUES ARE IN OHMS, 1/10W, 5%
2. ALL CAPACITORS ARE 50VDC OR HIGHER.
3. THE ORIGATION SOURCE OF A VOLTAGE IS REPRESENTED BY (VCC), AND ALL REFERENCES TO THAT VOLTAGE ARE REPRESENTED BY (VCC).

APPEND THE FOLLOWING DOCUMENTS WHEN CHANGING THIS DOCUMENT:		DRAWING CONTENT:		TITLE		<div> 2900 SPAFFORD ST. DAVIS, CA 95616 530 - 757 - 4616</div>	
		DRAWN BY: (INITIAL RELEASE)					
		E. PEAK	14MAY01				
		REVISED BY:					
		D MUSGROVE	01/21/02				
		APPROVALS: INITIAL RELEASE					
		PROJECT ENGINEER:		SIZE		DWG NO.	
		ERIC PEAK	5/14/01				
		ENGINEERING MANAGER:		B		090-0128	
		R MATTHEWS	5/14/01				
		SIGNATURES	DATE	SCALE		NONE	
				RELEASE DATE		SHEET 1 OF 1	

