Powerful Processors – Easy to Use™

CAN Development Kit

User's Manual

RCDK8C

Rev. 1.91
July 2006

www.renesas.com

Table of Contents

1.0 Information Sources for CAN D Kit

See the kit's CD-ROM for more information on the components of the CAN Development Kit. Among the documents on the CD-ROM are User Manuals for all RSK boards with CAN-equipped MCUs that are compatible with this kit.

The table below lists all the documents that are on the CAN D Kit CD-ROM. Those documents have also been copied to your PC during installation. They can be viewed by clicking on *Start* > *(All) Programs* > *Renesas* > *RCDK8C* > *Documents*.

Table 1.1: Reference manuals

Item	Title	Description
1.	**CAN D Kit Quick Start Guide**	Document to help you get started using the CAN Demonstration Kit.
2.	**Can D Kit User's Manual**	This document.
3.	**M16C CAN API Application note**	Describes how to use a set of API (Application Programming Interface) function calls for the M16C family of microcontrollers to which the R8C/Tiny belongs.
4.	**CAN D Kit Demo & Expansion Manual**	Describes how to use other RSK boards that can run and do a demo together on a CAN bus.
5.	**RS-R8C25(23) Schematics**	Schematic for the RSK-R8C23 boards.
6.	**CAN D-Kit Package Contents**	Printed - in kit box.
7.	**R8C23 Hardware Manual**	Guide for programming the R8C/23 series MCU.
8.	**High-performance Workshop User's Manual**	This document describes installation and operation of this Integrated Development Environment for Renesas' Tools.
9.	**Assembler User's Manual**	Guide for AS30 assembler.
10	**C compiler User's Manual**	Guide for NC30WA C-compiler.
11	**E8 User's Manual**	E8 Target Debugger and Programmer user's manual.

For sample programs, updates, and evaluation tools for the CAN D Kit, please go to http://www.america.renesas.com/can

For a list of Application Notes for the R8C/23, go to http://www.renesas.com and click on `Products / MPU and MCU / M16C Family / R8C/Tiny Series`, then click on `Application Notes`.

For a list of Application Notes for the M16C/29, go to http://www.renesas.com and click on `Products / MPU and MCU / M16C Family / M16C/Tiny Series / M16C/29 Group`.

For a list of Application Notes for the M16C/6NK, go to http://www.renesas.com and click on `Products / MPU and MCU / M16C Family / M16C/60 Series / M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN Group`.

2.0 Contents of Product Package

When unpacking your CAN D Kit, please check to see that all items listed below are included.

2.1. CAN Demonstration Kit Item List

Table 2.1 CAN D Kit Item List

Item Name	Quantity	Remarks
RSK-R8C23 boards	2	Renesas Starter Kit (RSK) boards pre-programmed with demonstration software.
DC power supply	1	15W, 5V, 2.6A output, 90-264V AC input, multi-plug power supply. Each RSK board draws max 50mA with full CAN communication.
LCD	2	2-line × 8-character LCDs with KS0066 controller IC
SysTec CAN Sniffer	1	CAN Bus to PC USB Interface for monitoring CAN bus traffic.
E8 USB Debugger	1	In-system Programmer and Debugger Unit.
6′ Mini USB Cable	1	Connects E8 target debugger to Host PC.
6′ USB Cable	1	Connects SysTec CAN Sniffer to PC.
CAN bus cable assembly	1	Cable assembly to connect three (3) RSK boards and the SysTec CAN Sniffer to the CAN bus. The SysTec CAN Sniffer is connected via a 9-pin Sub-D connector; the RSK boards via 3-pin single-row ICD connectors.
DC RSK-board multi-plug supply cable	1	Parallel-connects RSK boards' DC supply inlets, to power up to five RSKs.
CD-ROM		Quick Start Guide (QSG) Auto-install program for HEW with integrated IDE, C-compiler (NC30WA), assembler, librarian, and linker. E8 USB drivers CAN D Kit manuals RSK manuals MCU manuals Schematics Sample programs SysTec CAN Sniffer software and USB driver.

2.1.1. CD-ROM

The CD-ROM contains the electronic manuals and software necessary for developing programs. Your computer must have a web browser — like Mozilla Firefox, Netscape® Browser or Microsoft® Internet Explorer — to view the help files, and Adobe® Acrobat® Reader® to view the manuals.

Insert the enclosed CD into your computer. The installer should auto-start. The installer program will create `C:\Renesas` and `C:\WorkSpace` folders on your machine. NC30WA C-Compiler, E8 Programmer, Documentation, sample code, and other CAN D Kit related files will be installed in the `C:\Renesas` folder. The High-performance Embedded Workshop (HEW) integrated development environment is installed in the `C:\Program Files\Renesas` folder by default.

If the installer program does not start automatically, browse to the CD's root folder and double-click on `RCDK8C_Installer.exe` to start the installation.

3.0 Limited Guarantee and Support

Renesas Technology America, Inc., warrants the CAN D Kit to be free from component or assembly defects for a period of 180 days from the date of purchase. Settlement is limited to repair or replacement of the product only. Renesas Technology America, Inc., does not assume any liability arising out of the application or use of any product, circuit or procedure described herein. No other liability or warranty applies, expressed or implied. Software warranty is limited to replacement of the CD only. While every attempt has been made to ensure accurate documentation, Renesas Technology America, Inc., cannot be held responsible for errors or omissions, and reserves the right to make changes without prior notice.

4.0 System Connectivity

The following hardware and software products are required to use the CAN D Kit.

- Host Computer (supplied by user)
- Two or more RSK Boards of type RSK-R8C23, RSK-M16C29, and RSK-M16C6NK
- CAN bus cable assembly, consisting of red and white CAN bus wire and serial cable with 9-pin Sub-D connector to connect the SysTec CAN Sniffer and RSK boards to the CAN bus
- RSK intra-board DC supply cable to parallel-connect up to five boards (alternatively, the E8 programmer/debugger may power up to six RSK boards)
- SysTec CAN-to-USB Interface (CAN Sniffer)
- USB cable to connect SysTec CAN Sniffer to PC
- SysTec USB-CANmodul Utility CD with PcanView software, user manual and USB driver

Required if you want to update firmware or develop code:

- E8 in-system programmer and debugger
- Mini USB cable to connect E8 to PC
- 2×5 header target cable for E8
- Renesas CD with software tools (HEW IDE, NC30 Compiler/Linker, E8 Programmer)

Figure 4.1 shows two RSK-R8C23 Boards and a CAN Sniffer connected to the CAN bus via the CAN bus cable assembly included with the kit. The E8 in-system debugger and programmer is connected to one board for code development.

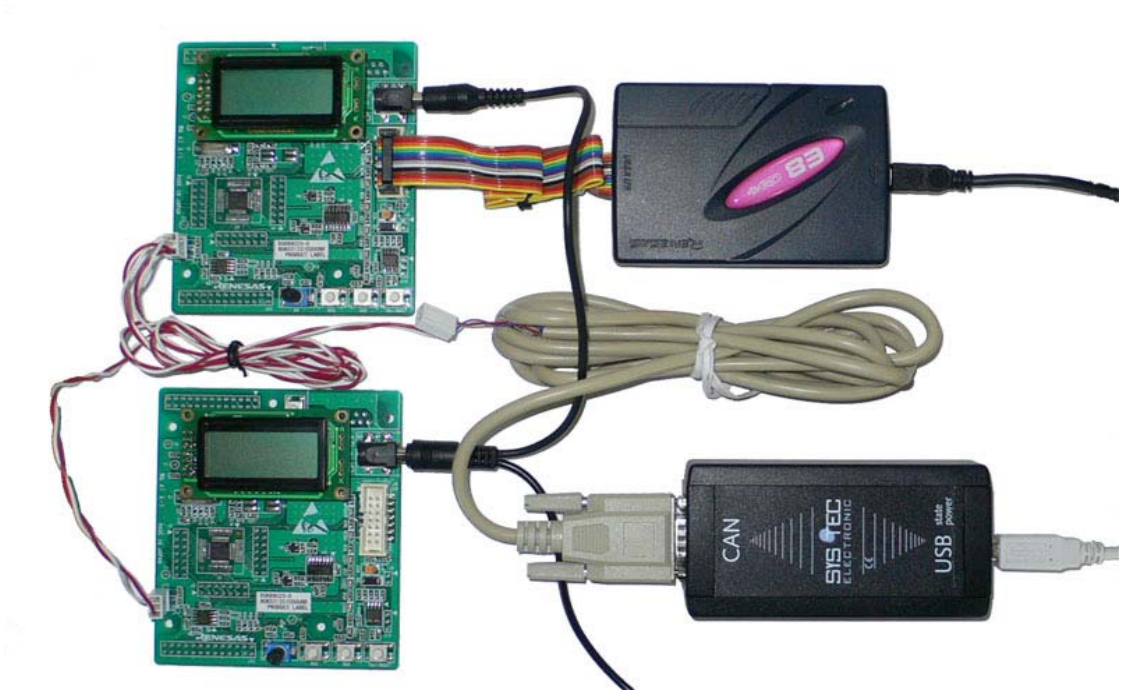


Figure 4.1: CAN Development Kit System Connectivity

4.1. Host Computer Requirements

The minimum requirement to be able to use the software that comes with the CAN D Kit is a PC with a USB port and Microsoft Windows 2000 or XP.

4.2. Compatible RSK boards and demonstration code

The two included RSK-R8C23 (RSK23) boards are pre-programmed with M16C API-based CAN demo firmware. Any combination of RSK-boards of type RSK-R8C23, RSK-M16C29 or RSK-M16C6NK can be used to run the demonstrations, and they can be used together with HEW to develop new CAN application code.

Demo firmware for all these boards exists in a single High-performance Embedded Workbench (HEW) project, allowing the boards to work together seamlessly on the CAN bus.

See the CAN Demonstration and Expansion Manual included on this kit's CD, and the respective user manuals of RSK-M16C29 and RSK-M16C6NK, for more details on using those boards.

See the CAN Demonstration and Expansion Manual for information on how to run the demo firmware, connect different RSK boards, and program and debug RSK-R8C23, RSK-M16C29 or RSK-M16C6NK boards.

This user manual is specific to the RSK-R8C23 (RSK23) boards only.

See the M16C family API CAN Application Note and the in-depth "CAN Application Note" for details about how to use the function calls in your application to send CAN data.

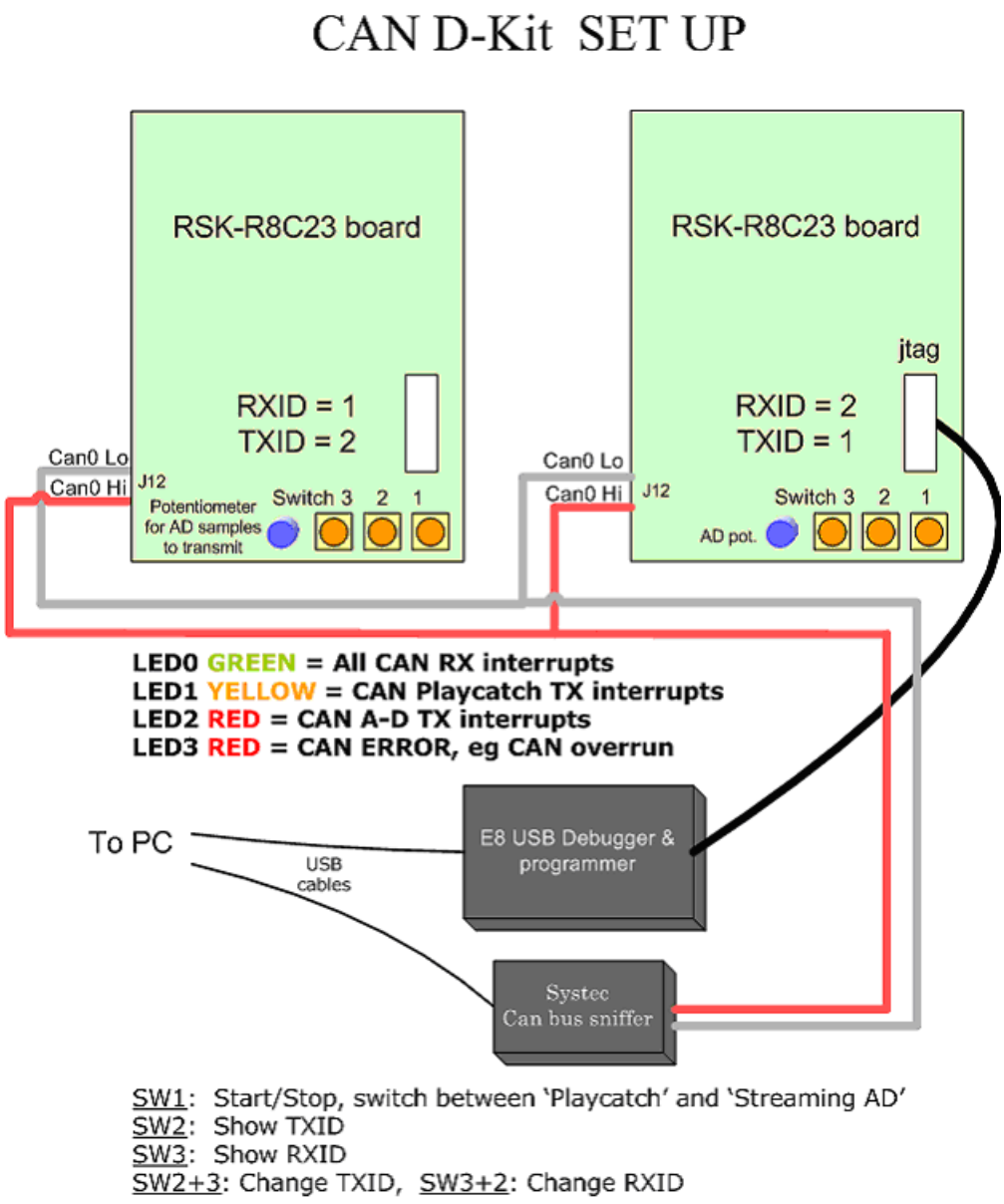


Figure 4.2: CAN Kit System Setup

4.3. Power Supply

The kit comes with a 5V DC power supply. The included “daisy chain” extension cable with additional DC power plugs allows you to power up to five RSK boards with one power supply. Each RSK board draws only about 50mA with full CAN communication.

It is possible to omit the power supply by using the E8 to supply the boards. The board connected to the E8 will have 5V in its DC supply jack that can in turn supply other boards. In order to use the E8 as a DC supply, the board to which it is connected must be used in a debug session with a downloaded X30 file. That board’s DC supply socket then may be used to distribute 5V to as many as five other RSK boards. (The E8 is rated at 300 mA.)

4.4. SysTec CAN Sniffer

Before using the SysTec CAN Sniffer, please install the USB-CANmodul USB driver and the PcanView program from the USB-CANmodul Utility CD-ROM. See Appendix H for details of the hardware and software installation process.

4.4.1. USB-CANmodul Hardware

The USB-CANmodul is a device for connecting the CAN bus to a PC using a standard USB interface. The USB-CANmodul supports all CAN-based higher-layer protocols, such as CANopen, DeviceNet or J1939. The maximum CAN bus baud rate of 1 Mbit/s is supported. The CAN messages are buffered temporarily by the USB-CANmodul, so transfer peaks are picked up.

See Appendix I for more information about the Sniffer software packages.

Figure 4.3: SysTec CAN Sniffer

The USB-CANmodul supports CAN specification 2.0A and 2.0B. Drivers, tools and demo software for Windows operating systems are included in the supply package. SysTec provides driver updates on their web site, http://www.systec-electronic.com.

4.4.2. CAN Sniffer Cable

The SysTec CAN Sniffer connects to the CAN bus via the Sub-D serial cable with one end cut open:

- Purple wire, CAN Hi: Connected to pin 8 of the serial Sub-D cable connector.
- Red wire, CAN Lo: Connected to pin 2 of the serial Sub-D cable connector.

The serial cable then connects to the CAN bus cable that goes out to the RSK boards. The wires of the CAN bus cable are red for CAN Hi, and white for CAN Lo.

4.5. E8 Debugger and Programmer

The E8 provides a plug-and-play debugging and programming interface to the RSK boards via the host computer's Universal Serial Bus (USB). The USB port also provides power to the E8 and up to six RSK boards, thereby eliminating the need for an external power supply. Use of the E8 is required only if you

need to update the firmware of the Kit's boards or if you intend to develop and debug your own software. If not powered by the E8, the RSK boards can be powered via the on-board power sockets.

The E8 in-system debugger and programmer in combination with the HEW software provides programming capability for the Kit or any Renesas target board that uses an M16C-family Flash MCU (i.e. an MCU from the groups R8C, M16C, or M32C). Please see the RTA-E8 User's Manual for more details on the E8.

See the CAN D Kit Demonstration and Expansion Manual for more details on programming and debugging the RSK23 boards with HEW and the E8 Debugger. Refer to R8C/23 Group Hardware Manual for details of programming the microcontroller without using these tools.

4.6. Software Development Tools

The installer program offers you the option to install some or all of the development tools. For details on installation, see the CAN D Kit Quick Start Guide. A brief description of all the included tools follows. Please refer to the individual tool manuals for detailed information.

4.6.1. HEW (High-performance Embedded Workshop)

HEW provides a Graphical User Interface (GUI) that integrates the software development tools and includes the C-compiler, assembler, librarian, linker, debugger, and editor.

4.6.2. HEW Debug Interface

HEW communicates with a kernel (i.e. a ROM monitor program) on the target MCU through the E8. This debug interface provides a highly efficient evaluation environment. Features include:

- Source-level debugging for assembly and C language
- Single-step command with breakpoints
- *Run* command with breakpoints for the MCU
- RAM monitor function
- C variable "watch" window

4.6.3. Debug Using Symbols

Normally when a new project is created using HEW, debugging symbols are enabled. If you are unable to view the source properly during debug, add the debug option [-g] in HEW before compiling the programs. To enable the [-g] option:

- Open the workspace and project in HEW.
- Select "Renesas M16C Standard Toolchain" from the *Options* pull-down menu.
- Click on the *Link* tab.
- Select "Output" under the *Category* list box.
- Click on the checkbox for [-g] "Outputs source debug information…"
- Click <**OK**>.

For more information, see the HEW user's manual.

4.6.4. NC30WA Evaluation Version C Compiler

The evaluation version of the M3T-NC30WA C-compiler is provided with the same functionality as the commercial version, except that link size will be restricted to 64 Kbytes after 60 days from when you begin using the compiler. Contact your local sales representative if you wish to purchase a full license.

4.6.5. Renesas AutoUpdater

The Renesas AutoUpdater utility can be configured to search the Renesas website (or your server) automatically for updates of the Renesas tools installed on your PC.

4.6.6. Documentation, Sample Projects and Project Generators

The CAN D Kit includes a full set of user documentation and sample code. After installing the CD, the sample projects can be found in the `C:\Renesas\RCDK8C\Demo_code` folder. Documentation can be browsed via the Start menu (*Start* > *(All) Programs* > *Renesas* > *RCDK8C* > *Documents*).

5.0 Hardware

5.1. RSK-R8C23 (RSK23) Board

The Kit comes with CAN demo code for the RSK-R8C23 board that works seamlessly with the demo code for RSK-M16C29 and RSK-M16C6NK boards on the same bus.

The RSK23 boards are marked *RSKR8C23*. Details of the board can be seen in the layout (see Appendix B) and schematics included on the CD.

Figure 5.1 shows the RSK23 Board with major components identified.

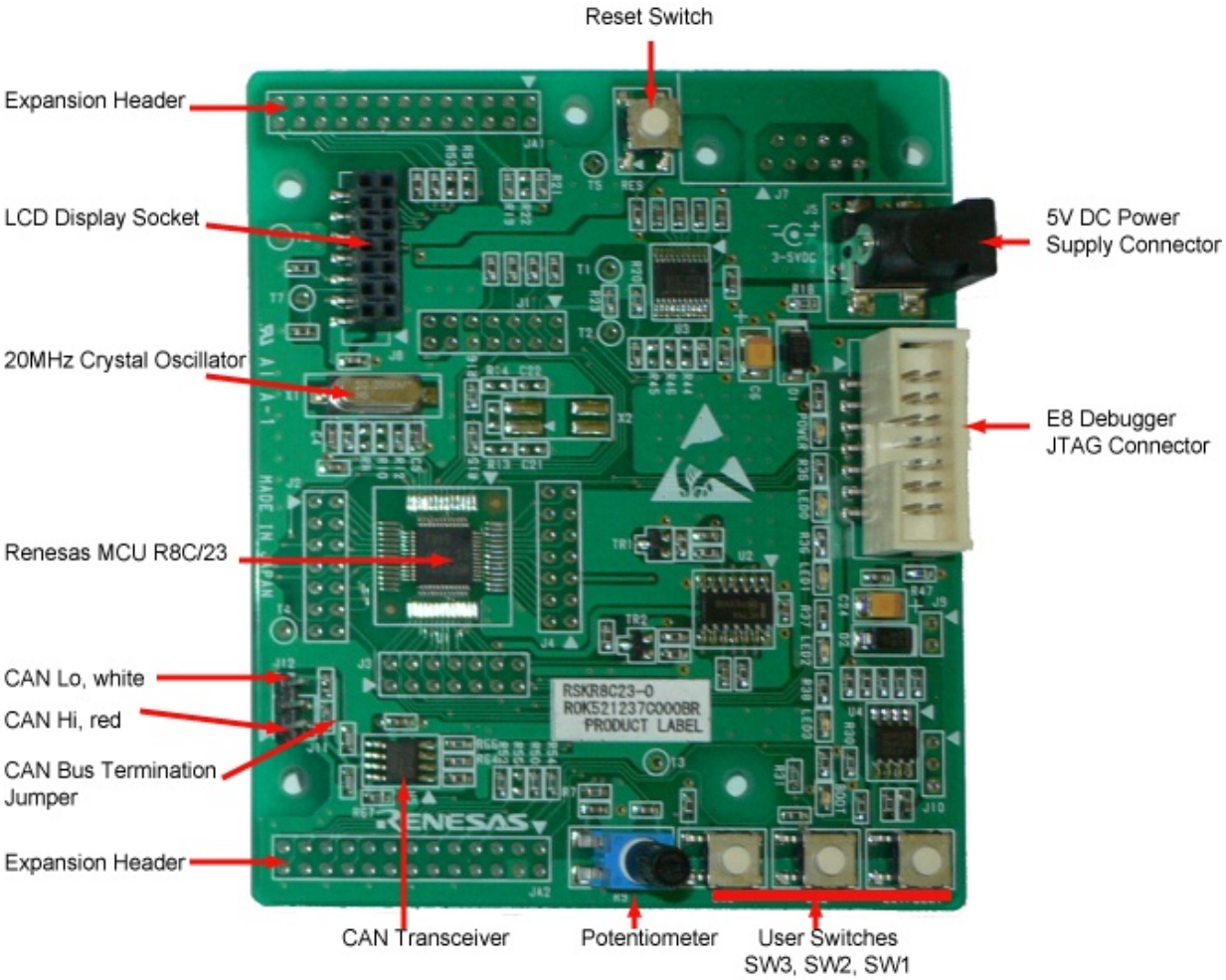


Figure 5.1: RSK-R8C23 Board

5.1.1. The R8C/23 MCU

The RSK-R8C23 board incorporates an R8C/23 R5F21237 MCU from the M16C/Tiny (or R8C) group of microcontrollers.

The R8C/23 is a 16-bit single-chip Flash microcontroller of the M16C series CPU core. The hardware and software manuals for this group of microcontrollers can be found in the `C:\Renesas\RCDK8C\Docs` folder on your PC, or by using the Start menu (*Start* > *(All) Programs* > *Renesas* > *RCDK8C* >*Documents*) after CD software installation.

The R8C/23 is based on the R8C CPU core and has 1MB of memory space. Maximum operating frequency is 20MHz. Internal Flash Memory is programmable on a single power source.

Key features of the R5F21237 MCU:
- 8-bit Multifunction Timer with 8-bit prescaler (Timer RA and RB): 2 channels
- Input Capture/Output Compare Timer (Timer RD): 16-bit × 2 channels
- Timer with compare match function (Timer RE): 1 channel
- CAN Module (2.0B): 1 channel, 16 slots
- UART + Clock Synchronous Serial Interface: 1 channel
- I²C-bus™ Interface (IIC)/Chip-select Clock Synchronous Serial Interface (SSU): 1 channel
- LIN Module: 1 channel (Timer RA, UART0)
- 10-bit A/D Converter: 12 channels
- Watchdog Timer
- Clock Generating Circuits: XIN Clock Generation Circuit, On-chip Oscillator (High/Low Speed)
- Oscillation Stop Detection Function
- Voltage Detection Circuit
- I/O Ports: 41
- Interrupts: 14 internal factors, 6 external factors, 4 software factors
- 2.5 Kbytes RAM
- 48 Kbytes Program Flash
- Data Flash: 2 Kbytes

For more information, visit http://www.renesas.com and click on `Products / MPU and MCU / M16C Family / R8C/Tiny Series.`

5.2. Switches

There are four switches located on the RSK. The function of each switch and its connection are shown in Table 5.1. Refer to the schematics for detailed connectivity information.

Table 5.1: RSK23 Switch Functions

Switch	Function	Microcontroller Pin
RES	Reset Switch. When pressed, the MCU is reset.	RESET Pin
SW1/Boot	Connects to an IRQ input for user controls. The switch also is used in conjunction with the RES switch to place the device (MCU) in BOOT mode when not using the E8 debugger.	INT0 Pin25 (Port 4, pin 5)
SW2	Connects to an IRQ Interrupt input line for user controls	INT1 Pin20 (Port 1, pin 7)
SW3	Connects to a Key-In Interrupt input line for user controls	KI3 Pin24 (Port 1, pin 3)

5.3. LEDs

The RSK23 board has six LEDs. The green “POWER” LED lights when the board is powered. The yellow “BOOT” LED indicates the MCU is in Boot Mode when lit. The four user LEDs are connected to an I/O port, and will light when their corresponding port pin is set low.

Table 5.2 below shows the LED pin references and their corresponding microcontroller port pin connections.

Table 5.2: RSK23 User LED Ports

LED (as shown on silkscreen)	Color	Microcontroller Port Pin	Package Pin Number
LED0	Green	Port2_4	15
LED1	yellow	Port2_5	14
LED2	Red	Port2_6	13
LED3	Red	Port2_7	12

5.4. RSK23 Jumper Configuration

A Termination Resistor jumper is used to connect the termination resistor to the CAN bus. Use one termination resistor at each extreme end of the bus. The termination resistors are connected by default at shipment, even without a jumper in place. For the jumper to have any effect, the 0Ω resistor R62 must be removed.

On the RSK23, the jumper for CAN bus termination is JP11.

5.5. Potentiometer

A single-turn potentiometer is connected to pin AN8 (P1.0) of the microcontroller. This may be used to vary the input analog voltage value to this pin between VREF and Ground.

5.6. Serial Ports

The RSK23 board’s connector J7 has not been fitted, and provides room for an optional 9-pin Sub-D serial connector. The MCU’s UART1 interface may be connected to the on-board RS232 transceiver — which in turn connects to J7 — by fitting 0Ω resistors in locations R45 and R46. However, there are restrictions on using UART1 with the E8 debugger, see chapter 7.2 for details.

Table 5.3: Serial Port Settings

Serial Signal	Function	Resistor to be fitted to enable RS232
TxD1	MCU serial port1 transmit	R45
RxD1	MCU serial port1 receive	R46

The MCU’s other serial port, UART0, is connected to the application header JA2 pins 6 (SCIaTx) and 8 (SCIaRx). UART0 also is shared with the LIN module (J10).

5.7. Liquid Crystal Display Module

The LCD module is a 2-line by 8-character display with a KS0066 controller IC. It connects to J8. The LCD should be fitted in such a way that it lies over J1. Please make sure that the LCD module's pins are inserted correctly into J8.

Even though the RSK board can operate from 3V to 5V DC, the LCD module only supports 5V operation.

The LCD module uses a 4-bit interface to reduce the pin allocation. No contrast control is provided; the value of resistor R11 on the supplied display module sets the contrast.

Table 5.4: LCD Module Connections

J8 Pin	Signal Name	MCU pin	J8 Pin	Signal Name	MCU pin
1	Ground	-	2	5V DC	-
3	No Connection	-	4	LCD_RS	29
5	R/W select (hard wired to write)	-	6	LCD_E	28
7	No Connection	-	8	No Connection	-
9	No Connection	-	10	No Connection	-
11	LCD_D4	47	12	LCD_D5	46
13	LCD_D6	46	14	LCD_D7	44

5.8. RSK23 Board Option Links

Functionality can be added to, or removed from, the RSK23 board by fitting or removing 0Ω resistors. This is referred to as Option Links. Appendix E describes the function of the Option Links for this RSK board.

5.9. Oscillator Source

A 20MHz crystal oscillator in HC/49U package is fitted on the RSK23 board and is used to supply the main clock input to the Renesas microcontroller.

5.10. RSK23 Reset Circuit

The RSK23 Board includes a simple latch circuit that combines mode selection and reset function for the MCU. It provides an easy method for switching the microcontroller between Boot Mode and Single Chip Mode. This circuit is not required on your own board designs, as it is intended for providing easy evaluation of the operating modes of the device on the RSK23. Please refer to the R8C/23 hardware manual for more information on the requirements of the reset circuit.

5.11. CAN Bus Cable

The accompanying CAN bus cable has a red wire for CAN Hi and a white wire for CAN Lo. CAN Hi is marked by an arrow on the silkscreen for the RSK board's 3-pin CAN connector. The board's CAN connector center pin is for ground connection between boards, which is necessary when connecting measurement equipment, such as oscilloscopes, between several different boards. See also chapter 4.4.2 CAN Sniffer Cable.

5.12. CAN Transceiver

The RSK boards use the TJA1041 High-speed CAN transceiver. Detailed specifications can be found on the CAN D Kit CD.

5.13. CAN Bus Error Codes

If CAN bus communication errors cause a board to enter a CAN bus error state, the RSK board's LCD will display an error code. The following error codes are defined:

Bus: 03 Board is in CAN 'bus error passive' mode. This condition may happen if a board is not connected to the CAN bus correctly and can therefore not receive recessive bits.

Bus: 05 Board is in CAN 'bus off' mode. This condition occurs if a board is not able to transmit on the CAN bus.

Note: The demo code automatically exits the 'bus off' state after a while, which you should not do in a real world application. If a CAN node cannot transmit, it is usually faulty and should stay off the bus until the fault has been corrected.

6.0 Operating Modes

The RSK23 Board supports both Boot Mode and Single Chip Mode. In Boot Mode, the microcontroller executes code from its Boot memory area, waiting for code to be downloaded to its Flash program memory. In Single Chip Mode, the microcontroller executes code programmed into its Flash program memory.

6.1. Boot Mode

Holding the MCU's Mode pin low while at the same time applying and releasing Reset will switch the MCU into Boot Mode.

The software supplied with this Kit supports programming of the microcontroller's Flash program memory in Boot mode using the E8 in-circuit debugger/programmer and HEW software only.

However, hardware exists on the board to enter boot mode manually. Do **not** connect the E8 if you want to enter Boot Mode manually. Instead, press and hold the SW1/BOOT switch, which will hold the Mode pin low, then press and release Reset. Finally, release the Boot switch. The "BOOT" LED will be illuminated to indicate that the microcontroller is in Boot Mode.

When the E8 is not connected and the board is placed in Boot Mode as described above, the Mode pin is pulled high by a 4.7kΩ resistor.

When an E8 is used, the Mode pin is controlled by the E8.

6.2. Single Chip Mode

Holding the MCU's Mode pin high while applying and releasing Reset will switch the MCU into Single Chip Mode.

The RSK23 board always will boot in Single Chip Mode when the E8 is not connected and the Boot switch is not depressed, because a 4.7kΩ resistor on the board pulls the MCU Mode pin high.

Refer to R8C/23Group Hardware Manual for details of Single Chip Mode.

7.0 System Limitations

The RSK-R8C23 provides sophisticated on-board debugging features at a low cost. The RSK-R8C23 does have some limitations when used with the HEW software and E8 Debugger. Section 7.1 introduces the kernel (ROM monitor) program and its purpose. The limitations when this kernel is running with the user program are listed in Table 7.1. For details on debugging R8C/23 MCUs, see the "E8 User's Manual" available from the Start menu (*Start* > *(All) Programs* > *Renesas* > *RCDK8C* > *Documents*).

Table 7.1: System Limitations when Debugging

Item	Please Refer To
User Limitations	7.2 Pin and Peripheral Limitations
	7.3 Memory Map
	7.4 Limitations on Interrupts
Debugger Limitations	7.5 Instruction Limitations
	7.6 User Program's Real-Time Capability

7.1. Kernel (ROM Monitor) Introduction

During debug, a small program called a kernel is uploaded to the R8C/23 MCU. The kernel communicates with HEW through the E8 target debugger regarding MCU status during user code debugging operations.

There are no special steps required in the user program to make use of the E8 (you only need to select the *Download Emulator firmware* option when HEW attempts to connect to the MCU). After downloading the kernel, the MCU is ready to download user code. The operation of the kernel is transparent to the user, but there are some limitations. These are discussed from section 7.2 onward.

Connecting the E8 without starting HEW will not affect the signal lines connected between the E8 and the MCU; the E8 keeps the signal lines in a high-impedance state. The E8 only drives the pins after HEW or the E8 Programmer software attempts to connect.

After completing program debug and verification with HEW, you can create an image of your code in Intel (.hex) or Motorola (.mot) file formats. The .mot image can be programmed into the MCU using the E8 Programmer. This procedure erases the kernel and leaves only the user program.

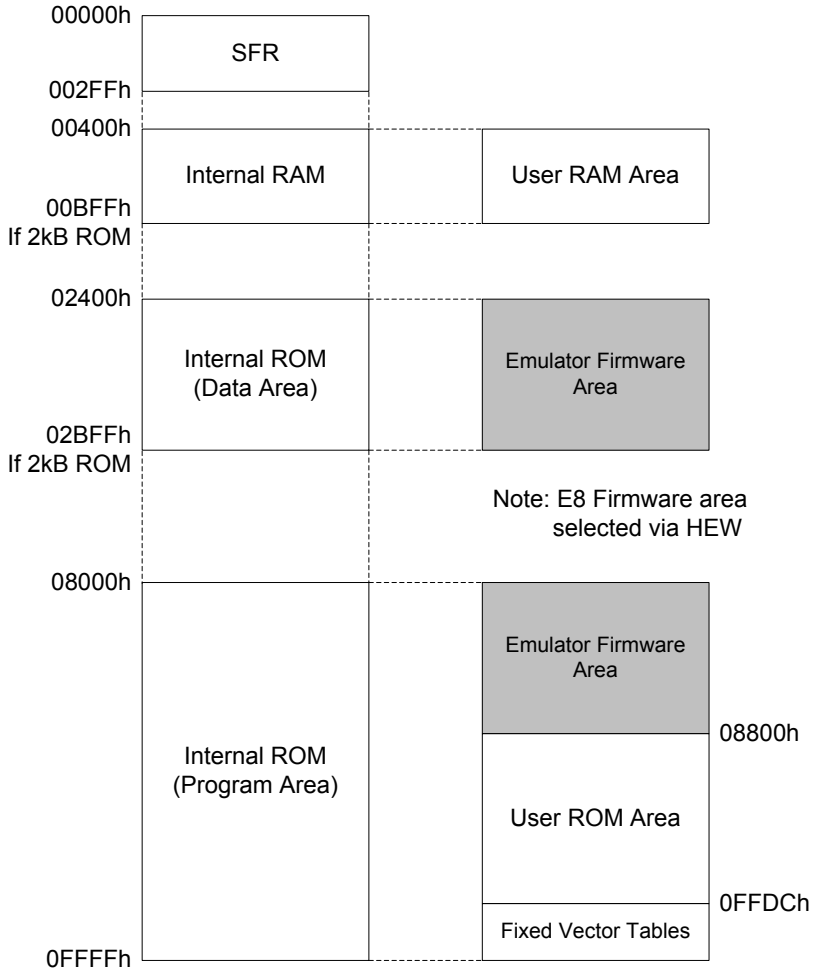
7.2. Pin and Peripheral Limitations

Although the MCU can use the on-chip oscillator while debugging, a clock signal must be connected to the OSC1 pins when the E8 Debugger is connected (i.e. do **not** remove X1). The on-chip oscillator is not recommended for use with CAN.

Do not access UART1-related registers, or stopping and stepping will not work. If you application code uses UART1, press Reset-> Go in HEW if you did a single step or program execution stop.

7.3. Memory Map

The amount and location of memory used by the kernel on the RSK23 board's MCU is shown below. .
The Special Function Register (SFR) area for CAN is allocated at addresses 01300h to 0147Fh.



7.4. Limitations on Interrupts

Do not set the Address Match Interrupt (the AIER, RMAD0, RMAD1 registers and the fixed vector tables) in a user system.

7.5. Instruction Limitations

Do not use the BRK instruction in a user system.

The stack pointer with up to 8 bytes is used during the USER program break. Therefore, save up to 8 bytes of space for the stack area.

7.6. User Program's Real-Time Capability

Please be aware that while the kernel is in a "STOP" state, the hardware peripherals will continue to run. Therefore, interrupts will not be serviced by their processor-run routines.

While the kernel is in a "RUN" state, there is no overhead on the application code unless a RAM monitor window is open. This window requires periodic communication with the MCU. This communication suspends normal application operation while servicing the request (approximately 2000 BCLK cycles for

each 16 bytes of data displayed in the window are used per window update). The user must determine whether this behavior is acceptable.

7.7. Watchdog Timer

When the E8 Debugger is activated, the watchdog timer is disabled. The user program should not enable the watchdog function. The watchdog timer will not be serviced if the debug kernel is in a “STOP” state, and likely will time out if active.

8.0 RSK23 Board Specifications

Table 8.1 provides an overview of the RSK-R8C23 hardware.

Table 8.1: CAN D Kit RSK-R8C23 Board Specifications

Item	Specification
MCU	R5F21237J
Clocks	Main Clock: crystal 20 MHz
Connectors	[J1-J4]: Four 2×7-pin measurement test points connected to the MCU pins. Can also be used to connect your own expansion boards via 2×7 headers. [J6]: In-Circuit Debug connector for E8 debugger [J12]: CAN Connector
Jumpers	[J11]: For connecting/disconnecting CAN bus termination resistor. See chapter 6.2 Jumper Configuration.
Switches	[SW1]: pushbutton (connected to MCU pin 25, Port 4_5) [SW2]: pushbutton (connected to MCU pin 20, Port 1_7) [SW3]: pushbutton (connected to MCU pin 25, Port 1_3) [RES]: pushbutton, MCU Reset
LEDs	LED0 Green MCU pin 15, Port 2_4 LED1 Yellow MCU pin 14, Port 2_5 LED2 Red MCU pin 13, Port 2_6 LED3 Red MCU pin 12, Port 2_7
LCD	2-line × 8-character LCD with KS0066 controller IC
Data Memory	2.5 Kbytes RAM, 2Kbytes Data Flash
Program Memory	48 Kbytes Flash

8.1. Power Supply Requirements

The RSK23 board operates from a 3V to 5V power supply. A diode provides reverse-polarity protection only if a current-limiting power supply is used. Typical board power consumption is about 50mA.

The RSK23 board has a positive-center supply connector using a 2.1mm barrel power jack.

The E8 in-circuit debugger also can provide power to a RSK23 board via the E8 JTAG connector. The E8 is able to provide up to 300mA in current. The board connected to the E8 will have 5V at its DC supply jack that can in turn supply other boards. In order to use the E8 as a DC supply, the board to which it is connected must be used in a debug session with a downloaded X30 file. That board's DC supply socket then may be used to distribute 5V to as many as five other RSK boards.

WARNING

The RSK23 has no under- or over-voltage protection. Use a regulated 5VDC, center-positive supply for this board.

8.2. Operating Environment

Table 8.2 lists the environmental conditions for using and storing the RSK23 boards. Store the boards in a conductive bag inside the original factory packaging.

Table 8.2: Operating and Storage Environments

Environmental Condition	Ambient Temperature	Ambient Humidity
Operating	0 to 55°C (No corrosive gas allowed)	30 to 80% (non-condensing)
Storage	-30 to 75°C (No corrosive gas allowed)	30 to 80% (non-condensing)

9.0 CAN Demonstration Firmware

Two demos are incorporated into the Kit: “Streaming A-D” and “PlayCatch”. Upon Power-up or when all boards are reset, they are in INIT state and the LCD displays **Renesas CANDKit**.

In INIT state, **all** boards connected to the CAN bus will enter “PlayCatch” mode if **any** board’s Switch 1 (SW1) is pressed. Any subsequent press of a board’s SW1 will switch that particular board only between “PlayCatch” and “Streaming A-D” modes.

9.1. Running the Firmware

See the Quick Start Guide for instructions on setting up and running the demos for the two RSK23 boards that come with your Kit. See the CAN D Kit Demonstration & Expansion Manual for more information on the demo code and on programming and debugging that code for all RSK boards with CAN (RSK23, RSK29, and RSK6NK).

10.0 CAN Baud Rate

See the “M16C CAN API Application Note” with its worksheet for help on changing the CAN data speed. There are additional comments in “CAN Application Note”. You also will need the hardware manual for your MCU.

Appendix A. Troubleshooting Guide

This section discusses possible problems you may encounter while installing the development tool software and USB drivers, or running the HEW debugger and E8 Programmer applications. This section also discusses the countermeasures and solutions to resolve these problems.

For troubleshooting information on the SysTec CAN Sniffer interface and SysTec CAN Sniffer hardware, see the SysTec CD.

If, for any reason, you cannot resolve the problem, please contact your Renesas representative for assistance.

A.1 USB Driver Problems

This part discusses how to fix common problems that may occur with USB driver installation. The most common problem is that Windows did not properly install the USB drivers, so the E8 Debugger is not recognized. In this case, Windows Device Manager may indicate that the “Renesas Emulator” device is not working properly.

Before trying the following steps, try restarting your PC to see if that resolves the problem. You can check the USB Driver status using the Windows Device Manager (*Start* > *Control Panel* > *System Properties* > *Hardware* > *Device Manager* > *Universal Serial Bus controllers*). Expand the “Renesas Emulator” entry, and if the “Renesas E-Series USB Driver” appears with **no** red X or yellow exclamation point, the driver was installed properly.

***NOTE: You will need Administrator privileges to be able to install the drivers**.*

For cases where “Renesas E-Series USB Driver” appears with a red X or yellow exclamation point in the Windows Device Manager, please try the following:

1. Open the Windows Device Manager (*Start* > *Control Panel* > *System Properties* > *Hardware* > *Device Manager* > *Universal Serial Bus controllers*).
2. Click on “Renesas Emulator” and double-click on “Renesas E-Series USB Driver”; a properties dialog box appears.
3. Click on the [Driver] tab and click the <**Update Driver**> button.
4. Select “Display a list…” and click on the <**Have Disk**> button.
5. Browse to the `C:\Windows\systems32\drivers` directory and install the `E1usb.sys` driver.

A.2 Debugging Problems

This section discusses the causes of common problems, and countermeasures to resolve them. The common problems encountered with debugging are:

- Erratic debug behavior
- Can’t connect to target
- Issues that may come up during debug operations

A.2.1 Erratic Debug Behavior

Although multiple instances of HEW can be launched, erratic behavior can result if more than one instance of HEW is open during a debug session. Running the E8 Programmer software at the same time as HEW also can result in erratic debug behavior. Having more than one E8 target debugger installed can cause erratic programs or cause HEW to crash.

A.2.2 Can't Connect to Target

If the message "Can't connect with the target" is displayed when attempting to connect, there are several possible reasons for the message to have appeared. Each cause and its corresponding countermeasure is discussed below.

Table A.1: Connection Problems

Problem	Possible Cause and Solution
The CAN D Kit board or the E8 target debugger are not connected correctly.	• Unplug the E8 from the USB cable (first connect the RSK target board to the E8 target debugger via the supplied 2×5-header ribbon cable) then re-connect the E8 to the USB cable.
USB was not selected in the HEW Init dialog box.	• Select "USB" from the Init dialog box that is displayed right after you start a debug session.
The E8 Debugger has no power (ACT LED of E8 is off).	• Ensure that the Mode switch under the cover on the E8 is in the "1" position. • The E8's power is supplied via USB. Check that your USB cable is not broken. Check that your PC's USB port is working correctly. If you connect the E8 via an USB hub, check the connectivity between the hub and your PC. If you use a self-powered hub, check the power supply of your hub.
The selected Device (MCU) when connecting and the actual target MCU do not match.	• Close the error message by clicking on the <**OK**> button, then click on the <**Cancel**> button of the Init window. Make sure you select the correct Device (MCU) when connecting (e.g. "R5F21237" for the RSK-R8C23). If the selected MCU's debugger file loaded on the E8 target debugger is different, HEW will re-program the E8 to match it. • Check that you actually are connecting to the correct RSK, as you won't get a warning telling you that the Device and board are incompatible — just connection failures.
The target MCU is damaged.	• Try a different target board and see if HEW will connect. You may have a damaged board or MCU.

A.2.3 Issues that May Arise During Debug Operations

Table A.2: Debug Operations Problems

Problem	Possible Cause and Solution
Seemingly erratic stepping behavior	• Remember that peripherals are not stopped when the MCU is stopped in the debugger. Peripheral registers may change value "behind your back" since they keep running.
HEW locks up (cannot stop program) or Communication error message is displayed.	• Press <**Disconnect**>. Wait. Unplug the E8 from the USB cable, then re-connect the E8 to USB. Press <**Connect**>.
Download problems	• Make sure that filenames or directory names do not contain spaces or special characters. • HEW project was not properly set up (startup files missing or out of order, files added to wrong member, etc.). Try creating a new project and adding your source files to it. For details, please see the HEW User's Manual.

Appendix B. RSK23 Board Layout

The following diagram shows the top layer component layout of the board

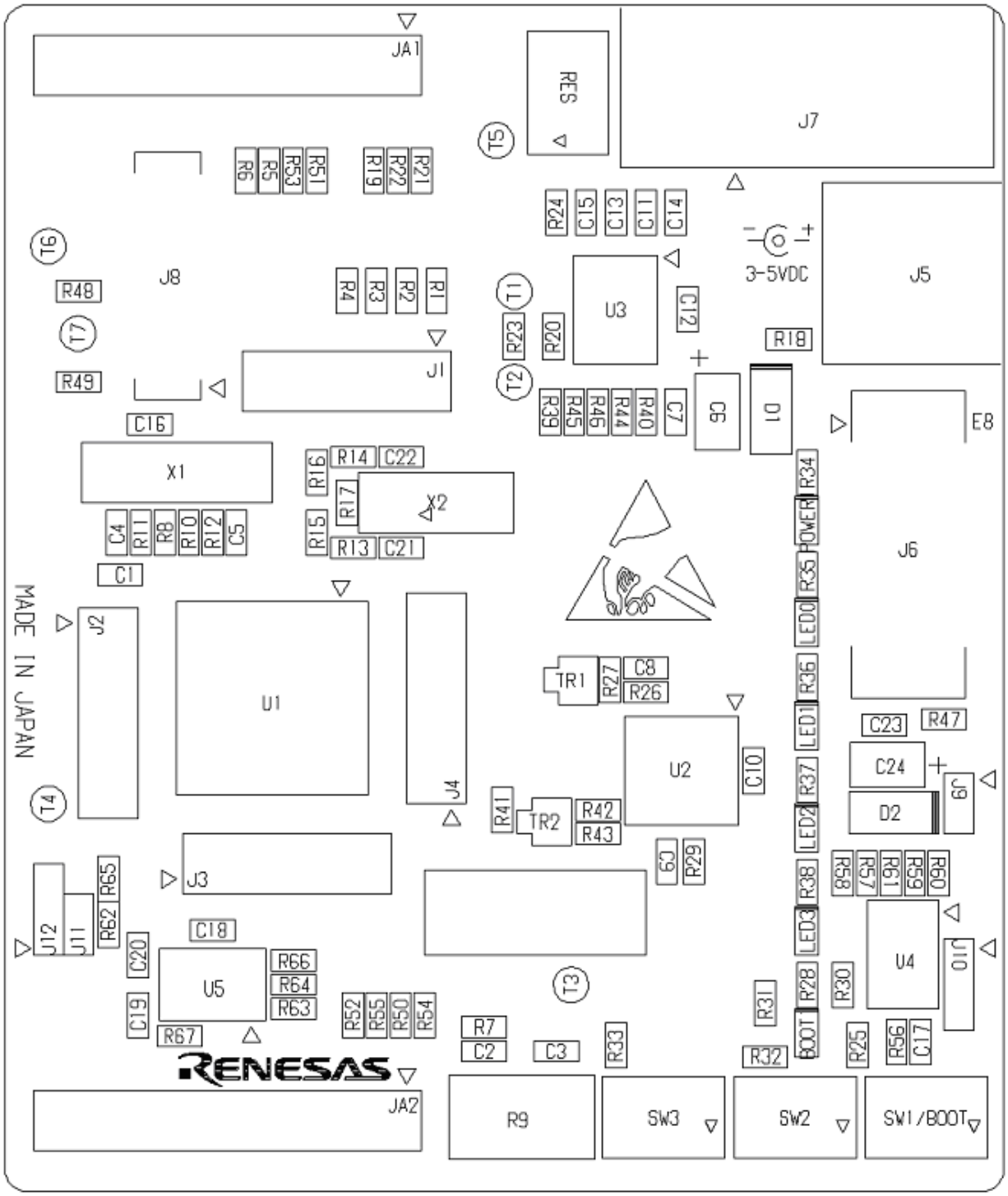


Figure B.1: RSK23 Component Layout

Appendix C. RSK23 Board Dimensions

The following diagram gives the board dimensions and connector positions. All through-hole connectors are on a common 0.1” grid for easy interfacing.

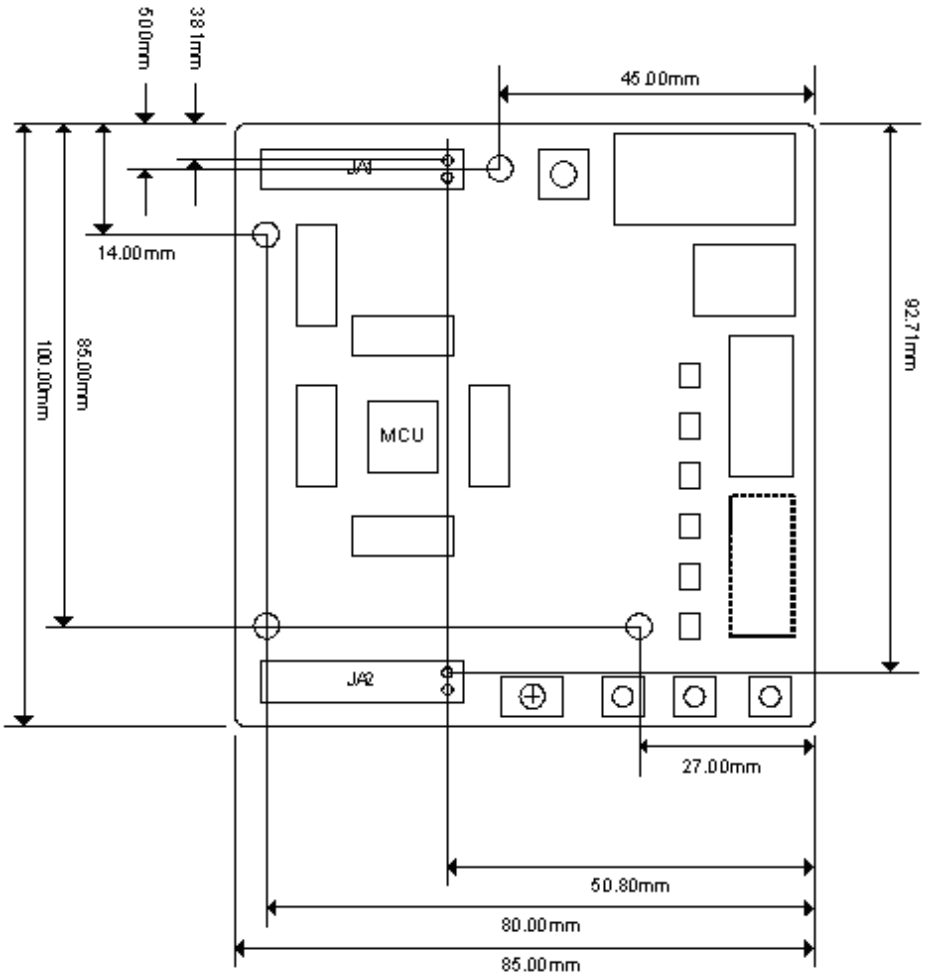


Figure C.1: RSK23 Board Dimensions

Appendix D. RSK23 Board Schematics

The circuit board schematics are available as a separate PDF document. It can be viewed via *Start* > *(All) Programs* > *Renesas* > *RCDK8C* > *Documents* >*R8C23-related*, or by browsing to the folder `C:\Renesas\RCDK8C\Docs\R8C23-related` and opening the file `RSK-R8C25-23_Schematics_v0r01.pdf`

Appendix E. RSK23 Board Option Link Settings

The default configuration settings are shown in **bold**.

Table E.1: RSK23 Option Links

Resistor Reference	Function	Fitted	Removed	Related To
R7	Reference Voltage	**Connects Reference Voltage to microcontroller**	Reference Voltage disconnected from microcontroller	R19
R8	Oscillator (Main clock)	Connects External Microcontroller header pins to microcontroller	**Disconnects sensitive microcontroller signals from external pins**	R10, R11, R12
R10	Oscillator (Main clock)	Connects External Microcontroller header pins to microcontroller	**Disconnects sensitive microcontroller signals from external pins**	R8, R11, R12
R11	Oscillator (Main clock)	**Connects main clock (X1) to microcontroller**	Main clock disconnected from microcontroller	R8, R10, R12
R12	Oscillator (Main clock)	**Connects main clock (X1) to microcontroller**	Main clock disconnected from microcontroller	R8, R10, R11
R13	Oscillator (Sub clock)	**Connects sub clock (X2) to microcontroller**	Sub clock disconnected from microcontroller	R14, R15, R16, R17
R14	Oscillator (Sub clock)	**Connects sub clock (X2) to microcontroller**	Sub clock disconnected from microcontroller	R13, R15, R16, R17
R15	Oscillator (Sub clock)	Connects External Microcontroller header pins to microcontroller	**Disconnects sensitive microcontroller signals from external pins**	R13, R14, R16
R16	Oscillator (Sub clock)	Connects External Microcontroller header pins to microcontroller	**Disconnects sensitive microcontroller signals from external pins**	R13, R14, R15
R17	Oscillator (Sub clock)	Parallel resistor for sub clock (X2)	**Not fitted**	R13, R14
R18	Board VCC	**Supply to board from DC Power Jack (J5)**	Disconnected	R20
R19	Reference Voltage	**Connects Board_VCC supply to Reference Voltage supply**	Reference Voltage MUST be provided from external interface	R7
R20	Board VCC	**Connects Board_VCC supply to board voltage line**	Board_VCC disconnected from board voltage line	R18, R19, R21, R22, R23
R21	Board VCC	**Connects External 5V (CON_5V) to Board_VCC**	External 5V disconnected from Board_VCC	R20, R22
R22	Board VCC	Connects External 3V3 (CON_3V3) to Board_VCC	**External 3V3 disconnected from Board_VCC**	R20, R21
R23	Microcontroller VCC	**Supply to microcontroller**	Fit Low ohm resistor to measure current	
R30	User I/O Power Supply	**Connects Board_VCC supply to SW2, 3 and LED0-3**	Board_VCC disconnected from SW2, 3 and LED0-3	
R31	SW1	**Connects SW1 to INT0 Input**	Disconnected	

Resistor Reference	Function	Fitted	Removed	Related To
R44	RS232 Transceiver	Disables RS232 Serial Transceiver	**Enables RS232 Serial Transceiver**	R45, R46
R45	Programming Serial Port	Connects RS232 port to Programming SCI port	**Disconnected**	R44, R46
R46	Programming Serial Port	Connects RS232 port to Programming SCI port	**Disconnected**	R44, R45
R47	E8	**Enables E8 Connection**	Do not connect an option resistor	
R50	Microcontroller pin function select	**Connects microcontroller pin 28 to IRQ1**	MUST be removed if R51 is fitted	R51
R51	Microcontroller pin function select	Connects microcontroller pin 28 to IO_6	**Should be removed if R50 is fitted**	R50
R52	Microcontroller pin function select	**Connects microcontroller pin 29 to IRQ2**	MUST be removed if R53 is fitted	R53
R53	Microcontroller pin function select	Connects microcontroller pin 28 to	**Should be removed if R52 is fitted**	R52
R54	Microcontroller pin function select	**Connects microcontroller pin 27 to IRQ0** (SW1)	MUST be removed if R55 is fitted	R55
R55	Microcontroller pin function select	Connects microcontroller pin 27 to TRIGa (SW1)	**Should be removed if R54 is fitted**	R54
R56	LIN	**For Master Mode**	For Slave Mode	R59, R60, R61
R59	LIN	**Connects microcontroller pin 22 to LIN-NSLP**	Disconnected	
R60	LIN	**Connects microcontroller pin 23 to LIN-RXD0**	Disconnected	
R61	LIN	**Connects microcontroller pin 23 to LIN-TXD0**	Disconnected	
R62	**CAN**	**Connects 120 ohm termination resistor to bus by default.** This resistor shorts (disables) the jumper.	Enables Jumper J11 to have effect.	
R64	**CAN**	**Connects transceiver TX port to MCU CAN0OUT port.**	Disconnects transceiver TX port to MCU	
R66	**CAN**	**Connects transceiver RX port to MCU CAN0IN port.**	Disconnects transceiver RX port to MCU	

Appendix F. RSK23 Headers

F.1 Microcontroller Headers

The RSK23 Board “MCU-ring” pin headers J1-J4 provide direct access to the microcontroller pins.

J1 Pin	MCU Pin	Board Net Name		J1 Pin	MCU Pin	Board Net Name
1	-	IIC_SCL		2	2	TRISTn
3	3	IIC_SDA		4	4	MODE_E8B
5	5	P4_3	**J1**	6	6	P4_4
7	7	RESet		8	8	CON_XOUT
9	9	VSS		10	10	CON_XIN
11	11	VCC		12	12	MO_Wn
13	-	No Connection		14	-	No Connection

J2 Pin	MCU Pin	Board Net Name		J2 Pin	MCU Pin	Board Net Name
1	13	MO_Vn		2	14	MO_Wp
3	15	MO_Vp		4	16	MO_Un
5	17	TMR0	**J2**	6	18	MO_Up
7	19	P2_0		8	20	TRIGb (SW2)
9	21	SCIaCK		10	22	SCIaRX
11	23	SCIaTX		12	24	IRQ3 – SW3
13	-	No Connection		14	-	No Connection

J3 Pin	MCU Pin	Board Net Name		J3 Pin	MCU Pin	Board Net Name
1	25	IRQ0/TRIGa* (SW1)		2	26	IRQ1/IO_6*
3	27	IRQ2/IO_7*		4	28	LCD_E
5	29	LCD_RS	**J3**	6	30	AD_POT
7	31	P3_1		8	32	TMR1
9	33	IO_5		10	34	IO_4
11	35	IO_3		12	36	AD0
13	-	No Connection		14	-	No Connection

J4 Pin	MCU Pin	Board Net Name		J4 Pin	MCU Pin	Board Net Name
1	-	No Connection		2	37	AD1
3	38	AD2		4	39	AD3
5	40	P4_2/VREF	**J4**	6	41	IO_0
7	42	IO_2		8	43	IO_1
9	44	LCD_D7		10	45	LCD_D6
11	46	LCD_D5		12	47	LCD_D4
13	48	MO_UD		14	-	No Connection

** For these signals there is no direct connection from the header to the MCU pin. Use of these signals requires the placement of a 0Ω resistor on the PCB and the removal of certain components on the board. See the schematics for details.*

F.1.1 Application Headers

The tables below show the standard application header connections

JA1							
Pin	**Header Name**	**RSK Signal Name**	**MCU Pin**	**Pin**	**Header Name**	**RSK Signal Name**	**MCU Pin**
1	Regulated Supply 1	CON_5V	-	2	Regulated Supply 1	GROUND	-
3	Regulated Supply 2	CON_3V3	-	4	Regulated Supply 2	GROUND	-
5	Analog Supply	NC	-	6	Analog Supply	NC	-
7	Analog Reference	CON_VREF	40	8	ADTRG	NC	-
9	ADC0	AD0	36	10	ADC1	AD1	37
11	ADC2	AD2	38	12	ADC3	AD3	39
13	DAC0	NC	-	14	DAC1	NC	-
15	IOPort0	IO_0	41	16	IOPort1	IO_1	43
17	IOPort2	IO_2	42	18	IOPort3	IO_3	35
19	IOPort4	IO_4	34	20	IOPort5	IO_5	33
21	IOPort6	IO_6*(Uart1 Tx)	26	22	IOPort7	IO_7*(Uart1 Rx)	27
23	IRQ3	IRQ3 – SW3	24	24	I²C Bus (3rd pin)	NC	-
25	I²C Bus	uIIC_SDA	3	26	I²C Bus	uIIC_SCL	1

JA2							
Pin	**Header Name**	**RSK Signal Name**	**MCU Pin**	**Pin**	**Header Name**	**RSK Signal Name**	**MCU Pin**
1	Reset	RESn	7	2	External Clock Input	CON_XIN	10
3	Interrupt	NC	-	4	Regulated Supply 1	GND	-
5	SPARE	NC	-	6	Serial Port	SCIaTX	23
7	Interrupt	IRQ0* - SW1	25	8	Serial Port	SCIaRX	22
9	Interrupt	IRQ1*	26	10	Serial Port	SCIaCK	21
11	Motor up/down	MO_UD*	48	12	Serial Port Handshake	NC	-
13	Motor control	MO_Up	18	14	Motor control	MO_Un	16
15	Motor control	MO_Vp	15	16	Motor control	MO_Vn	13
17	Motor control	MO_Wp	14	18	Motor control	MO_Wn	12
19	Timer Output	TMR0	17	20	Timer Output	TMR1	32
21	Timer Input	TRIGa* (SW1)	25	22	Timer Input	TRIGb (SW2)	20
23	Interrupt	IRQ2*	27	24	Tri-state Control	TRISTn	2
25	SPARE	P2_0	19	26	SPARE	P3_1	31

* *For these signals there is no direct connection from the header to the MCU pin. Use of these signals requires the placement of a 0Ω resistor on the PCB and the removal of certain components on the board. See the schematics for details.*

F.1.2 LIN Headers

J9		
Pin	**Function**	**Signal Name**
1	Power Supply (for LIN module)	VBAT
2	GROUND	GND

J10		
Pin	**Function**	**Signal Name**
1	Power Supply (for LIN module)	VBAT
2	LIN Bus Line	LIN
3	GROUND	GND

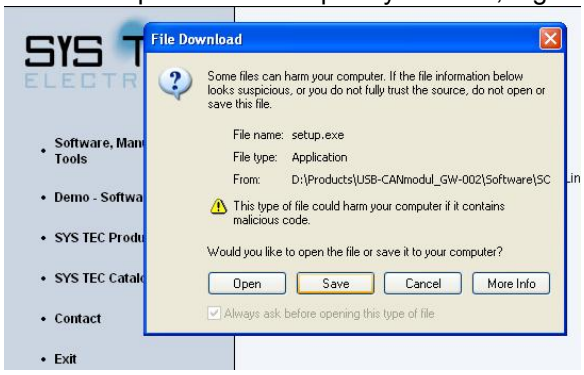
Appendix G. CAN D Kit Bill of Materials

Renesas	**Renesas Part Number**		**Comments**	**Quantity/ kit**
RSKR8C23	**R0K521237C000BR**		RSK-R8C23 Board	2
CAN D Kit Quick Start Guide			CAN D Kit QSG Document	1
RTA-E8	**R0E000080KCE00**		E8 for starter kits, with cables	1
CD			CAN D Kit CD with installer and documentation	
Digikey	**Digikey Part Number**	**Manufacturer: Part Number**	**Comments**	**Quantity/ kit**
Conn housing 3 pos. 100 hi press	**WM2602-ND**	Molex/Waldom: 10-11-2033	White CAN conn. Receptor	3
Conn term female 22-30 AWG tin	**WM1114-ND**	Molex/Waldom: 08-50-0114	"Tin pin" connectors	6
Wire, red, one drum 100ft.	**A3049R-100-ND**	Alpha: 3049RD005	Hook up wire AWG-26 stranded, Red	1 roll / 30 kits
Wire, white, one drum 100ft.	**A3049W-100-ND**	Alpha: 3049WH005	Hook up wire AWG-26 stranded, White	1 roll / 30 kits
CAN 90 deg. jumper pins '0.1″ KK breakaway headers'	**WM6303-ND**	Molex: 22-28-8033	RSK jumper pins - soldered into J12	2 (2 × 3pins)
Heat shrink KYNAR 3/8″ × 4′. Clear	**EPS3038K-ND**	Kynar	Heat Shrink Tube. Hold sniffer+CAN wire together at connector	1 tube / 30 kits
DC supply related material:				
15W 5V 2.6A Multi plug, 90-264 VAC	**T371-P5P-ND**	CUI: DMS050260-P5P-SZ	DC-supply 5V.	1
				2
PowerSuppliesOnline	http://www.powersuppliesonline.co.uk/product/Multi-2-1mm-DC-Plug-Extension-Lead/JR94C/default.htm	JR94C	Multi 2.1mm DC Plug Extension Lead	1
Conn power jack 2.1×5.5mm hi cur	**CP-002AHPJCT-ND (R)**	CUI: PJ-002AH-SMT	Receptacle for DC power	2
CAN sniffer		**Manufacturer: Part Number**	**Comments**	**Quantity/ kit**
USB CANmodul (CAN sniffer) Systec, Germany	-	**SysTec: GW-002**	SysTec CAN analyzer, Renesas N.A. has in stock	1
Serial Connection Cable, sniffer to RSK	-	**Assman: AK152-2-R**	DSUB 9-pin, female <=> DSUB 9-pin, female	1/2 (1 cable p.2 kits)

Not part of kit
TOOL CRIMP 14-24AWG UNIVERSAL, **WM9999-ND,** Molex/Waldom:63811-1000, Crimp tool to be used for kit manufacturing of CAN cabling.

Appendix H. Installing ‘USB-CANmodul’ Sniffer

1. Insert the SysTec CD that comes with your Sniffer hardware. When the automatic install screen asks you what to install, select “Install USB-CANmodul for Win2000 / WinXP”.
2. Save `setup.exe` to a temporary location, e.g. `C:\temp`.

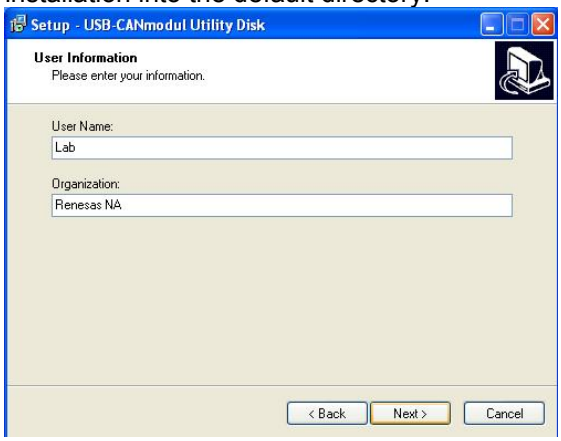


3. Open that temporary folder (e.g. `C:\temp`) and run `setup.exe`.

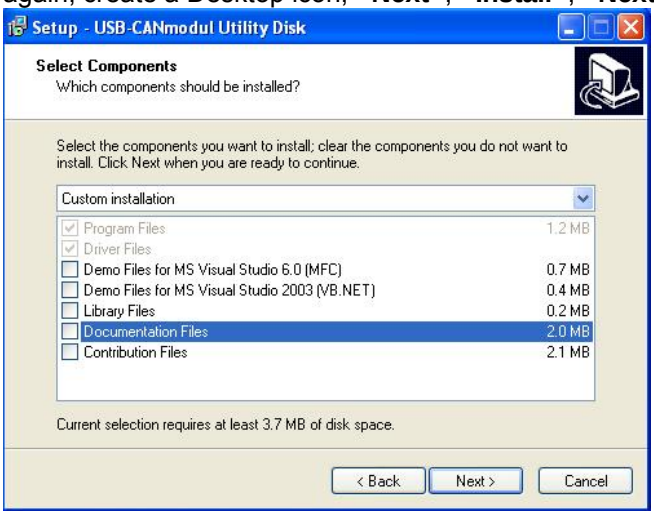
4. Make sure that the Sniffer is disconnected, then click <**OK**>.

5. Click <**Next**>, accept the license agreement, click <**Next**> and <**Next**> again.

6. Enter “Lab” in the *User Name* field, and “Renesas” in the *Organization* field, then click <**Next**> for installation into the default directory.



7. You only need to install the default content to run the lab, as shown below. Click <**Next**>, <**Next**> again, create a Desktop icon, <**Next**>, <**Install**>, <**Next**>, and <**Finish**>.

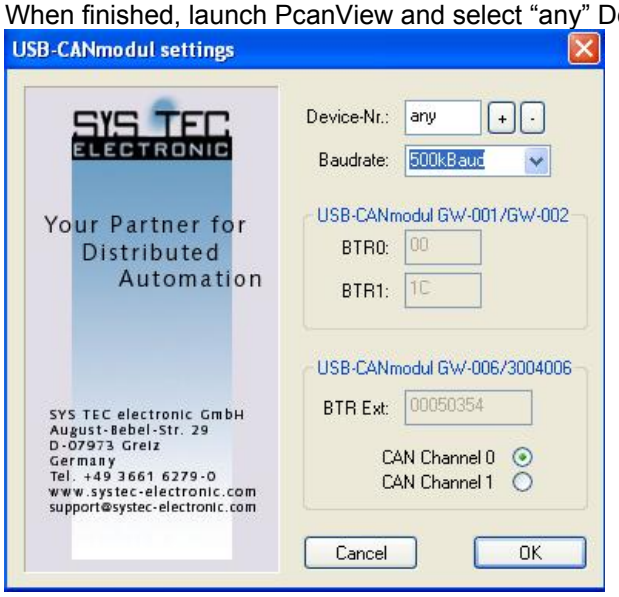


8. Connect the SysTec USB connector cable to the PC. After a while, you should see the "Found New Hardware Wizard".

9. The SysTec CD still should be in your PC's CD-ROM drive. Click <**Next**>. Click <**Yes**> for the "Not digitally signed" window, then click <**Finish**>.

10. If you get another "Found New Hardware Wizard", install the next driver.

11. When finished, launch PcanView and select “any” Device, “500kBaud” and *CAN Channel 0*.

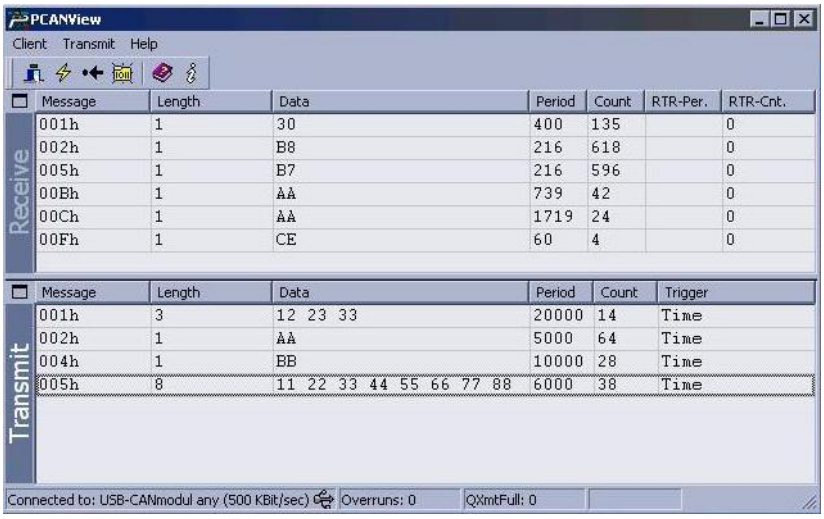


12. In the next window, select *Standard ID mode* and click <**OK**>. The Sniffer should come up with two sub-windows: “Receive” and “Transmit”. Installation and setup now is complete.

Appendix I. SysTec CAN Bus Sniffer Software

I.1 Sniffer Software, PcanView

The PcanView PC software is included on the SysTec CD and allows you to see transmitted and received data frames in real time on the bus. PcanView monitors the CAN bus and shows the latest data frame content of each CAN ID it encounters. To add functionality to the CAN Sniffer software beyond the basic PcanView, the programs PCAN-Explorer or CAN-REport can be used.



I.2 PCAN-Explorer Software

This software is not included with the Kit, but it may be used with the Sniffer.

PCAN-Explorer is a universal monitor for watching data traffic on a CAN network. In order to obtain a simple and clear assignment of the individual messages, they can be provided with their own references (symbols). The newly integrated Visual Basic scripting support makes it possible to create your own tools for the CAN bus easily. Data traffic on the bus can be acquired and stored using the integrated data logger. PCAN-Explorer 3 is realized as an automation server. The COM objects can be addressed using so-called dual interfaces; in other words, they provide access via the Dispatch interface or via a COM interface.

More information on PCAN-Explorer can be found at http://www.systec-electronic.com.

I.3 CAN-REport Software

This software is not included with the Kit, but it may be used with the Sniffer.

CAN-REport is a CAN-bus monitor and analyzing tool that puts you in the position to observe, record and evaluate the CAN communication on a logical level. Thanks to its programmability and extensibility of the user interface, CAN-REport can be adapted flexibly to customer requirements. Even in the standard version, CAN-REport is a powerful and functional package. CAN telegrams can be observed online in different modes and stored in a file for later processing.

You also have the capability of sending CAN telegrams manually, cyclically or sequentially using CAN-REport. The number of transmission channels available is freely configurable. The Record function in CAN-REport can be triggered by CAN telegrams. In addition, pre- and post-triggers can be defined. This makes it possible to acquire relevant information with precision timing. When this is done, the CAN messages can be interpreted by CAN-REport and its extensions during recording. A postponed processing of previously stored logs is also possible and allows shifting the time-intensive processing of huge amounts of data after recording.

CAN-REport comes with an extendable visualization interface. Thus, CAN-REport functionality can be extended by means of additional software modules. An example of this is the service- or protocol-dependent representation of CAN telegrams. Accessory modules are available for the CANopen and DeviceNet protocols which implement the service-dependent representation of the CAN telegrams. Detailed CANopen messages are displayed separately according to the SDO, PDO, NMT, EMCY or Flying Master services in separate windows.

CAN-REport can be downloaded and evaluated for 30 days by registering on the SysTec website http://www.systec-electronic.com.