BNS Solutions

QSK62P Plus
Quick-Start Kit

User's Manual

Revision 1.0

BNS Learning

www.bnssolutions.com

Table of Contents

1.0 Introduction

The QSK62P Plus Quick Start Kit (QSK) is a low-cost development environment for evaluating the M16C/60 series of microcontrollers (MCU) and Renesas software development tools.

The QSK62P Plus uses the M30620FC MCU, which belongs to the M16C/62P group of the M16C series. The M30620FC MCU has all of the on-chip peripheral functions of the M16C/30P group MCUs (plus some more), so it can be considered a “superset” suitable for evaluating the M16C/30P group as well.

The kits come with a complete software development tool chain for Renesas MCUs, including the NC30WA C-compiler, assembler and linker; an integrated “stripped down” E8 Emulator; and the High-performance Embedded Workshop (HEW), which consists of Integrated Development Environment (IDE), Graphical User Interface (GUI) and Software Debugger.

The QSK62P Plus boards feature on-board in-circuit debugging and programming support (ICD), eliminating the need for an external ICD unit. All that is required for in-circuit debugging and programming with the QSK62P Plus is a USB connection to a PC running the included development tool software. Power is provided by the USB link from your PC – eliminating the need for external power.

A real-time, source-level debug environment is implemented using the HEW4 software debug interface in conjunction with the QSK62P Plus boards. The E8 Debugger allows in-system programming of the board’s M16C/62P Flash MCU.

2.0 Contents of Product Package

This section describes the contents of the QSK62P Plus product package. When unpacking your QSK62P Plus, please check to see that all items listed below are included.

2.1 QSK62P Plus Quick Start Kit Item List

Table 2.1 lists the items included in the QSK62P Plus.

Table 2.1 QSK62P Plus Item List

Item Name	Qty	Remarks
QSK QSK62P Board	1	QSK62P QSK Board with integrated in-circuit debugging and programming support
Mini USB Cable	1	Connects QSK62P board to Host PC and provides +5v
Quick Start Guide	1	Information to get the QSK62P Plus board up and running quickly
CD-ROM	1	Auto-install program, HEW (IDE & debugger), NC30WA (C-compiler, assembler, and linker), USB drivers, Manuals, Tutorials, Sample programs

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 or Microsoft® Internet Explorer – to view the help files, and Adobe® Acrobat® Reader® to view the manuals.

Warning: To insure correct driver installation, do not connect your QSK62P Plus board until the CD software has been installed.

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, E8 Debugger, Documentation, and other QSK-related files are installed in the C:\Renesas folder. HEW is installed in the C:\Program Files\Renesas folder by default. Sample code is installed in the C:\Workspace\QSK_62P_Plus folder.

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

3.0 Limited Guarantee and Support

BNS Solutions warrants the QSK62P Plus 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. BNS Solutions 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, BNS Solutions cannot be held responsible for errors or omissions, and reserves the right to make changes without prior notice.

HEW is a trademark of Renesas Technology America, Inc. “QSK62P Plus” and the BNS Solutions and BNS Learning logos are trademarks of D’Agostino Industries Group, Inc. All other trademarks are the properties of their respective owners.

Support for the QSK62P Plus kit is provided via the following channels only:

1. Online User Forums at www.renesasuniversity.com

2. DigiKey Technical Support: www.digikey.com; click the “Contact Us” link

3. Renesas Technical Support Center: email at techsupport@renesas.com

4. BNS Solutions Technical Support center: email at techsupport@bnssolutions.com or online at www.bnssolutions.com/qsk.

4.0 System Connectivity

The following lists the hardware and software products required for using the QSK62P Plus Quick Start Kit.

- Host Computer (supplied by user)
- QSK62P Plus Board
- Mini USB cable
- Software Tools (HEW IDE, NC30 Compiler/Linker, E8 Debugger)

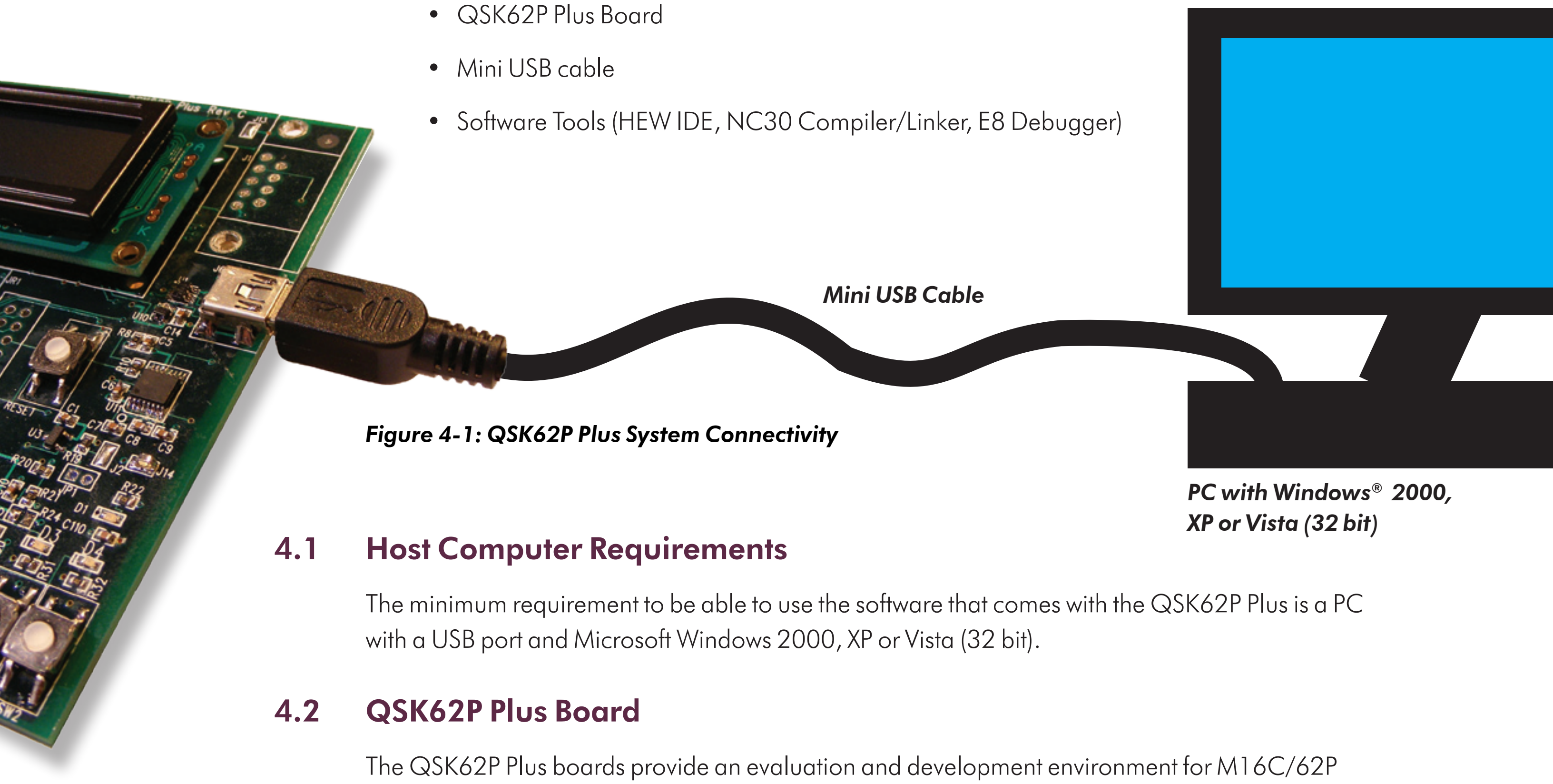


Figure 4-1: QSK62P Plus System Connectivity

4.1 Host Computer Requirements

The minimum requirement to be able to use the software that comes with the QSK62P Plus is a PC with a USB port and Microsoft Windows 2000, XP or Vista (32 bit).

4.2 QSK62P Plus Board

The QSK62P Plus boards provide an evaluation and development environment for M16C/62P group MCUs, respectively. See section "5.0" for more details.

4.3 Software Development Tools

The installer program installs all the development tools. For details on installation, see the Quick Start Guide or instructions in Appendix A of this manual. A brief description of all the included tools follows. Please refer to the individual tool manuals for detailed information.

4.3.1 HEW (High-performance Embedded Workshop)

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

4.3.2 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.

4.3.3 HEW Debug Interface

HEW communicates with a kernel (i.e. a ROM monitor program) on the target MCU through the debugger located on the back of the PCB (which is functionally identical to the E8 standalone debugger). This debug interface provides a highly efficient evaluation environment. Features include:

- Source-level debugging for assembly and C language
- Single-step command (unlimited breakpoints)
- Run command with 8 breakpoints* for the M16C/62P
- RAM monitor function
- C variable "watch" window

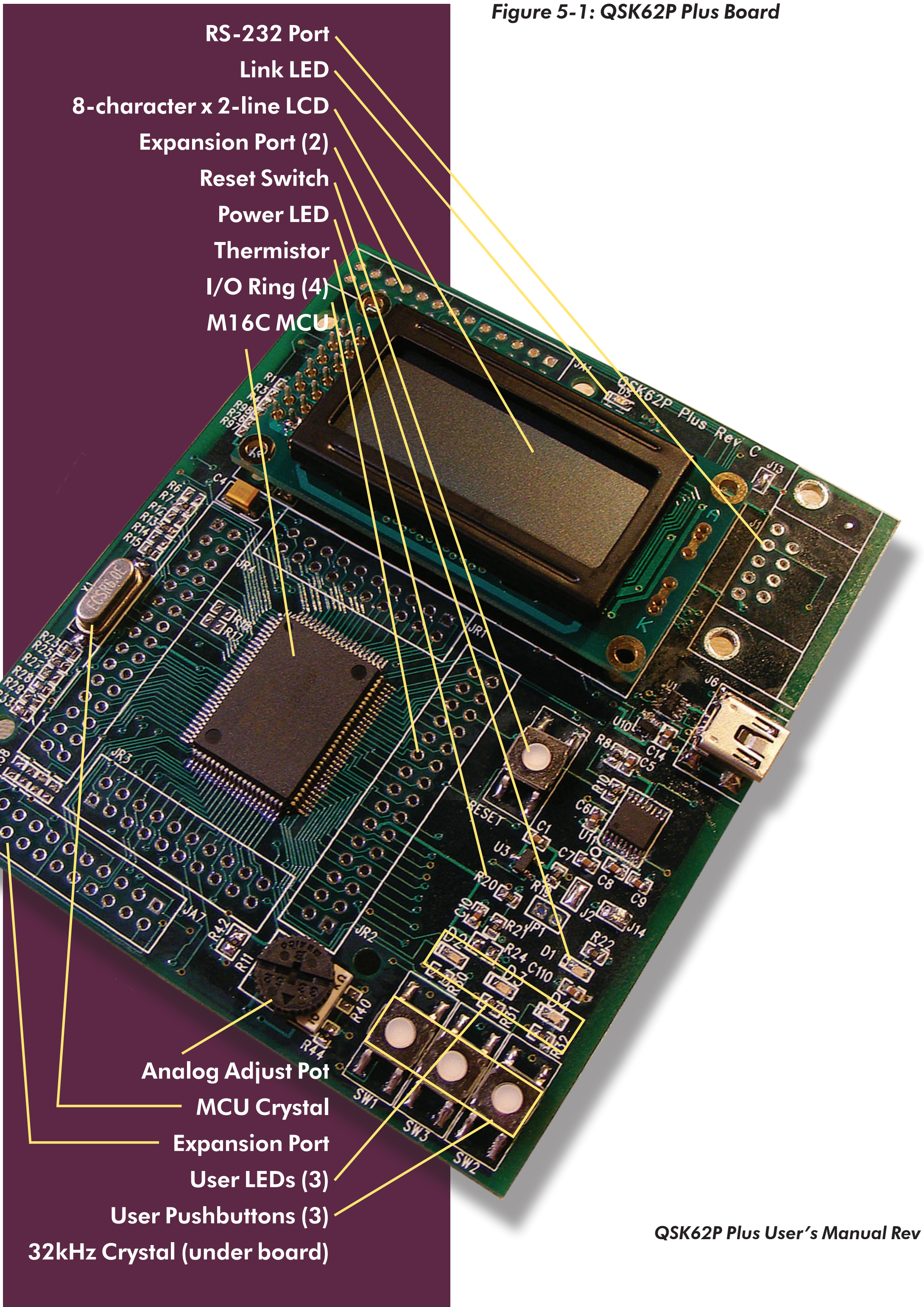
*Note: The number of breakpoints will vary depending on the M16C Flash MCU used.

5.0 Hardware

5.1 QSK62P Plus Board

Figure 5-1 shows the QSK62P Plus Board with major components identified.

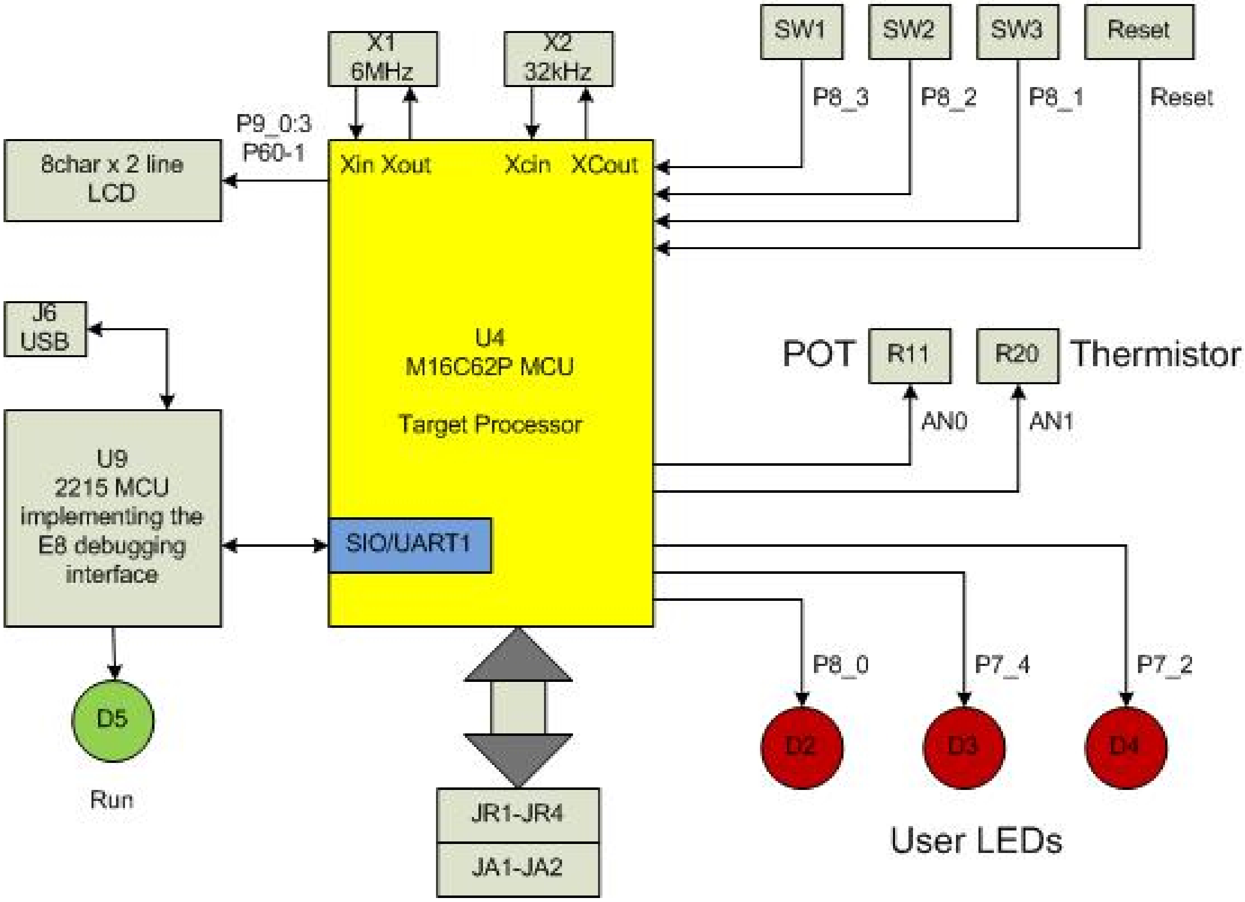
Figure 5-1: QSK62P Plus Board



5.2 QSK62P Board Block Diagram

The QSK62P Plus board incorporates an M30620FC MCU (100-pin QFP) from the M16C/62P group of microcontrollers, designated as U4. Figure 5-2 shows the QSK62P Plus block diagram.

Figure 5-2: QSK62P Plus Block Diagram



5.3 M16C/62P and M16C/30P Groups of MCUs

The QSK62P kits use an M30620FC MCU from the M16C/62P group of microcontrollers. The M16C/62P group is a functional superset of the M16C/30P group. The M16C/62P group features an integrated clock PLL circuit, which is not available on M16C/30P group MCUs. The QSK62P Plus makes use of the integrated PLL.

The M16C/62P group of 16-bit single-chip Flash microcontrollers is part of the M16C/60 family and utilizes an M16C/60 series CPU core. The hardware and software manuals for the M16C/62P group of microcontrollers can be found in the C:\Renesas\QSK62P\Docs folder on your PC or from the Start menu (Start > Programs > Renesas > QSK62P Plus > Document Descriptions) after QSK62P Plus software installation.

5.4 LCD (Liquid Crystal Display)

The LCD is a 2-line by 8-character display with a KS0066 controller IC.

6.0 System Operation & Limitations

The QSK62P Plus provides sophisticated on-board debugging features at a low cost. The QSK62P Plus' in-circuit debugging and programming circuitry (ICD) cannot be disconnected for use with other M16C-based boards.

The QSK62P Plus does have some limitations when used with the HEW software debugger and the board's ICD circuit. Section 6.1 introduces the kernel (ROM monitor) program and its purpose. The limitations when this kernel is running with the user program are listed in Table 6.1

Table 6.1: System Limitations when Debugging

Item	Please Refer To
User Limitations	6.2 Pin and Peripheral Limitations 6.3 Memory Map 6.4 Status After Reset 6.5 Register Operation Limitations 6.6 Limitations on Interrupts – Vectors that Reside in the Hardware Vector Table
Debugger Limitations	6.7 Stop or Wait Mode Limitations The kernel cannot be run in STOP or WAIT modes. Do not use these modes when debugging your program 6.8 User Program's Real-Time Capability

6.1 Kernel (ROM Monitor) Introduction

During debug, a small program called a kernel is uploaded to the M16C/62P (M30620FC) MCU. The kernel communicates with HEW through the ICD portion of the QSK62P Plus board regarding MCU status during user code debugging operations (the ICD is a stripped down version of the E8 debugger).

There are no special steps required in the user program to make use of the ICD portion. The operation of the kernel is transparent to the user, but there are some limitations. These are discussed from section 6.2 onward.

After starting a HEW debug session, the ICD uploads the kernel to the M16C/62P if it does not already exist (e.g. on a blank device or a device that was programmed for stand-alone operation). After downloading the kernel, the M16C/62P is ready to download user code.

Connecting the QSK30P/62P board without starting HEW will not affect the signal lines connected between the E8 MCU (H8/2215) and the M16C/62P; the ICD MCU keeps the signal lines in high-impedance state. The ICD MCU only drives the pins after HEW or the E8 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. This image can be programmed into the M16C/62P using the E8 in program mode. This procedure erases the kernel and leaves only the user program.

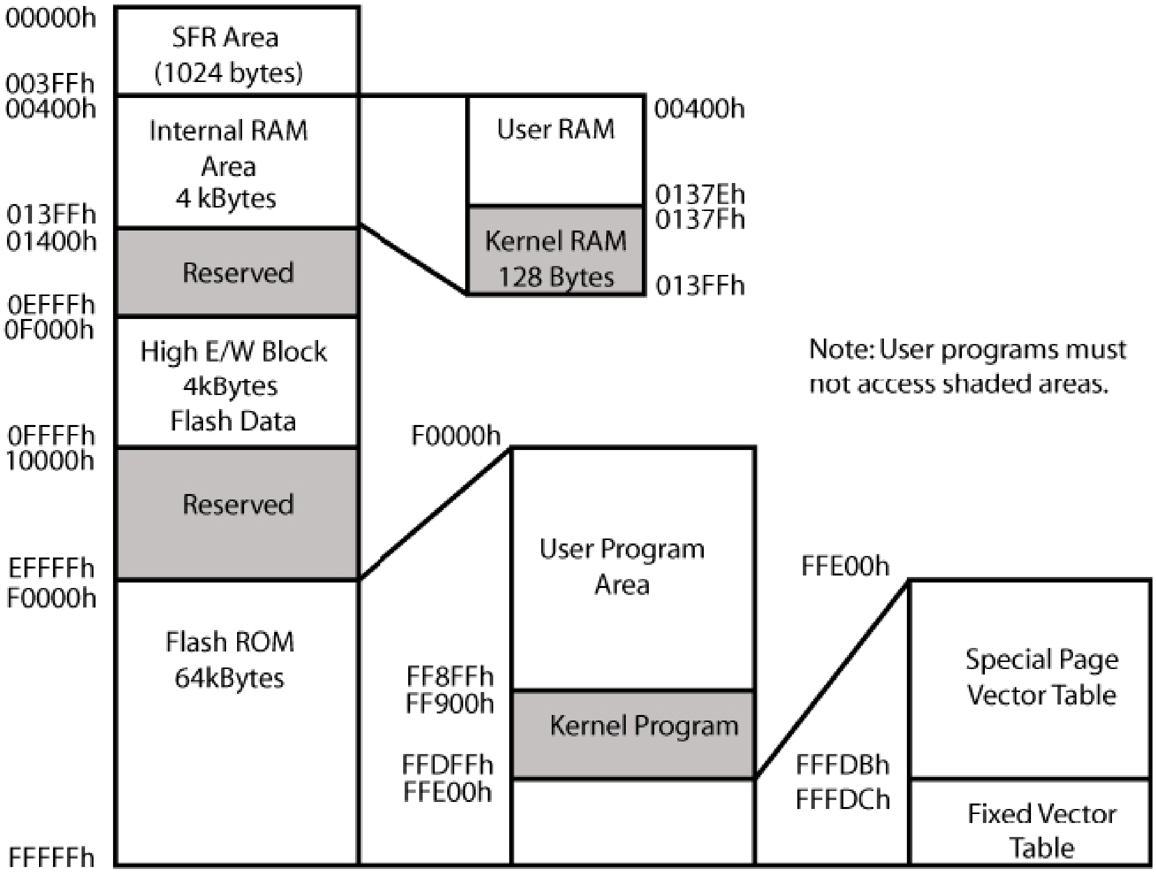
6.2 Pin and Peripheral Limitations

SIO/UART1 pins are used for communication between the M16C/62P kernel on the QSK62P Plus board and the board's ICD MCU. Do not connect these pins to any other circuit, as SIO/UART1 cannot be used in the user program while using the Debugger.

6.3 Memory Map

The amount and locations of memory used by the kernel on the QSK62P Plus board's M16C/62P MCU are shown in Figure 6-1.

Figure 6-1: M30622F8 Memory Map with the Kernel Program



Note: The kernel occupies memory associated with special-page vector numbers 18-19 and 192-255. The user reset vector is re-mapped to address FFFD8h by the kernel.

6.4 Status After Reset

Table 6.2 lists the values of the registers initialized by the kernel program when the QSK62P Plus board is reset. The register status depends on the microcontroller's internal operation performed after a reset.

Table 6.2: Initial Register Values After Reset

Register Name	Initial Value After Reset
Interrupt Stack Pointer	Initialized to 07F8416
UART1	Set for external clock synchronous mode

6.5 Register Operation Limitations

Table 6-2 lists the limitations on register operation. The registers are inhibited from any modification. If register contents are modified in any way, kernel operation cannot be guaranteed.

Table 6.3: Limitations on Register Operation

Register Name	Restriction
User and Interrupt Stack Pointers	RAM memory range 7B80H – 7BFFH is used by the kernel. Do not set stacks in this area.
UART1 Transmit/Receive Mode Register UART1 Transmit/Receive Control Register 0 UART1 Transmit/Receive Control Register 1	Do not change.
UART1 Interrupt Control Register 0	Do not change.
UART Transmit/Receive Control Register 2	Do not change bits 0 and 2.
UART1 Transmit Buffer Register	Do not write to this register.
UART1 Receive Buffer Register	Do not read this register.
Port 6 and Port 6 DDR	To prevent changes on P6_4 data and direction, use read-modify-write only instructions (BSET, BCLR, AND, OR, etc.).

6.6 Limitations on Interrupts – Vectors that Reside in the Hardware Vector Table

Table 6.4 lists the limitations on hardware interrupt (i.e. fixed) vector addresses.

Table 6.4: Interrupt Vector Addresses

Interrupt Cause	M16C/62P Vector Address	Kit Specification
Undefined	FFFDCh ~ FFFDFh	User available
Overflow	FFFE0h ~ FFFE3h	User available
BRK Instruction	FFFE4h ~ FFFE7h	User inhibited
Address Match	FFFE8h ~ FFFEBh	User inhibited
Single-step	FFFECh ~ FFFEFh	User inhibited
Watchdog Timer	FFFF0h ~ FFFF3h	User available (Note 1)
DBC	FFFF4h ~ FFFF7h	User inhibited
NMI	FFFF8h ~ FFFFBh	User available
RESET	FFFFCh ~ FFFFFh	Reset vector (Note 2)

NOTES:

(1) The Watchdog Timer vector is shared with the oscillation stop and voltage detection interrupts. The vector is available for oscillation stop and voltage detection interrupts, but you must avoid using the vector for watchdog timer interrupts.

(2) The kernel transparently relocates the Reset vector to FFFD8h.

6.7 Stop or Wait Mode Limitations

The kernel cannot be run in STOP or WAIT modes. Do not use these modes when debugging your program.

6.8 User Program's Real-Time Capability (Very Important – Please Read)

Please be aware that while the kernel is in a "STOP" state, the hardware peripherals will continue to run. Therefore, interrupts may not be serviced properly. In addition, the watchdog timer will not be serviced and will likely time out if active.

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.

6.9 Performing 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, perform the following:

- Open the workspace and project in HEW.
- Select 'Renesas M16C Standard Toolchain' from the Build 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 on the <OK> button

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

7.0 QSK62P Plus Board Specifications

7.1 Hardware Specifications

Table 7.1 lists the specifications of the QSK62P Plus Boards.

Table 7.1: QSK30P/62P Board Specifications

Item	Specification
MCU	M30620FC
Clocks	Main Clock: QSK62P – 6MHz crystal
Memory	RAM: 4kB (4096 Bytes) High E/W Data Block: 4kB Flash ROM: 64kB (65536 Bytes)
Connectors	[JR1-JR4]: Four dual row measurement test points connected to the MCU pins. [JA1-JA2]: Two dual row expansion ports where your own hardware can be connected. [J6]: Mini-USB connector used for in-circuit debugging and programming.
Switches	[SW1]: Pushbutton (connected to P8_3) [SW2]: Pushbutton (connected to P8_2) [SW3]: Pushbutton (connected to P8_1) [Reset]: Pushbutton (connected to Reset)
LEDs	[D5] (Green): Run LED (in-circuit debugging/programming activity) [D1] (Red): Power [D2] (Red): User output (connected to P8_0) [D3] (Red): User output (connected to P7_4) [D4] (Red): User output (connected to P7_2)
LCD	2-line × 8-character LCD with KS0066 controller IC

7.2 QSK62P Plus Power & Run LED Status

The green Connect LED (D5) and the red Power LED (D1) indicate operating status of the QSK62P Plus board. The board's 2215 MCU that controls the In-Circuit Debugging and Programming functionality of the board also controls the D5 LED. For the LEDs to function as per the table below, the 2215 MCU must be programmed with the correct MCU Monitor Image (MMI) for the QSK62P Plus. Boards shipped from the factory have the correct MMI loaded. In case your board's MMI ever gets corrupted, see "Appendix B Updating the QSK62P Plus Board's In-Circuit Debugger & Programmer Firmware" for help.

QSK62P Plus Status	Enum LED (green)	Power LED (red)
No Power (USB cable disconnected)	Off	Off
Connected, USB not enumerated (driver not installed)	Blinking	On
Connected, USB enumerated	On, Solid	On

7.3 Power Supply Requirements

The QSK62P Plus Board is powered through USB. When connected to USB, the board's ICD circuit is powered and the current drawn will be about 75mA.

7.4 Operating Environment

Table 7.2 lists the environmental conditions for using and storing the QSK62P Plus board. Store the board in a conductive bag inside the original factory packaging.

Table 7.2: 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)

Appendix A: Troubleshooting Guide

This section discusses possible problems you may encounter while installing the development tool software, USB drivers, or running the HEW debugger and E8 (The E8 is the debug hardware on the back side of the PCB). This section also discusses the countermeasures and solutions to resolve these problems.

If, for any reason, you cannot resolve the problem, further assistance is available via the following support channels:

1. Online User Forums at www.renesasuniversity.com
2. Digi-Key Technical support: www.digikey.com; click the "Contact Us" link
3. Renesas Technical Support Center: email at techsupport@renesas.com
4. Email BNS at: techsupport@bnssolutions.com

A.1 Manual Installation

Before connecting the QSK62P Plus to your PC, the driver files (.inf and .sys) and executables must be copied to the C:\Renesas\E8 directory.

To do this, download and run from the website Renesas' latest version of HEW and NC30 compiler, and the E8 debugger.

A.2 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 QSK62P Plus is not recognized. An indication of this problem is the QSK62P Plus' green connect LED blinking. When the driver is installed properly, the green connect LED will be on solid.

Before trying the following steps, try restarting your PC to see if this 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). If the "Renesas E8 Emulator" appears under the Universal Serial Bus controllers with no red X or yellow exclamation point, the driver was installed properly.

NOTE: If you are using Windows 2000 or XP, you will need Administrator privileges to be able to install the drivers.

For cases where "Renesas E8 Emulator" appears with a red X or yellow exclamation point in the Windows Device Manager, please try the following:

1. Locate the following files on your C: drive:

   E1usb.inf
   E1usb.sys

2. Open the Windows Device Manager (Start > Control Panel > System Properties > Hardware > Device Manager > Universal Serial Bus controllers).
3. Double-click on 'Renesas E8 Emulator'. A Renesas E8 Emulator Properties dialog box appears.
4. Click on the [Driver] tab and click the <Update Driver> button.
5. Select 'Display a list…' and click on the <Have Disk> button.
6. Browse to where you found the E1usb.sys file and click "install".
7. If this process does not work, please follow the instructions below.

   If you encounter problems on installing the drivers, you can try the following:

Windows 2000

a. Copy the E1usb.inf file from where you located it to the \WINNT\INF folder.
b. Copy the E1usb.sys file from the where you located it to the \WINNT\SYSTEM32\drivers folder.

Windows XP

a. Copy the E1usb.inf file from where you located it to the \WINDOWS\INF folder.
b. Copy the E1usb.sys file from where you located it to the \WINDOWS\SYSTEM32\drivers folder.

A.3 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
- Can't debug in HEW

A.3.1 Erratic Debug Behavior

HEW allows you to launch multiple instances of itself. However, if more than one instance of HEW is open during a debug session, erratic behavior can result. Having more than one QSK62P Plus board or a Renesas external ICD installed can also cause erratic problems or cause HEW to crash.

A.3.2 Can't Connect to Target

If the message "Can't connect with the target" is displayed when attempting to connect, there are several reasons that may have caused this message to appear. Each cause and its corresponding countermeasure is discussed below.

- The QSK62P Plus board is not connected correctly.

  Unplug the QSK62P Plus board from the USB cable. Then reconnect the QSK62P Plus board to your PC's USB port via the supplied mini USB cable. Please see section "4.0 System Connectivity."

- The QSK62P Plus board has no power (Power LED of the QSK62P Plus is off).

  The QSK62P Plus board'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 QSK62P Plus via a 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.

- Ensure you are selecting 3.3V in the emulator settings dialog. Selecting 5V will not allow the connection with the target to continue.

- The selected MCU In the debug session and the actual target MCU (M16C/62P) do not match.

  This would be rare for projects generated with the wizard. Start a new project workspace with the correct target and add the source files to it.

- 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.3.3 Can't Debug in HEW

HEW 4.0 is the first version of HEW to integrate the M16C debug interface. If you have inadvertently installed an older version of HEW, you will need to uninstall it and re-install HEW4.

A.3.4 Issues that May Arise During Debug Operations

While debugging user code, some issues may come up because the limitations discussed in section "6.0 System Operation & Limitations " were not satisfied. The common issues are listed in Table A.3, including the countermeasures.

Table A.3

Problem	Possible Cause/s and Solution
After stepping a few instructions, HEW cannot stop	Changes were made to the UART1 Special Function Registers (SFRs). Do not change UART1 SFRs in your code.
Breakpoints do not seem to work	System is in FreeRun mode. Change the RUN mode to Sampling from the Init window (Emulator System icon).
HEW locks up (cannot stop program) or Communication Error message is displayed	Changes were made to the UART1 SFRs. Do not change UART1 SFRs in your code. Ensure that no limitations in Section 6 were violated. Re-initialize the system without closing debug session. See note below. Do a hardware reset. User-program runaway may be corrupting the kernel (RAM, interrupt vectors, flags, etc.). Close the debug session, hit the reset button on the QSK board to reset the board, then restart.
Download problems	HEW project was not set up properly (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.

To re-initialize the system without closing a debug session, try the following:

- Click the <OK> button on the error dialog box to close it.
- When an Exit dialog box appears, click the <Cancel> button to close it.
- Press the Reset button on the QSK62P Plus board.
- Click the HEW Reset icon.

After initialization, debugging can resume. However, it is recommended that you download your program again before debugging.

B

Appendix B: Updating the QSK62P Plus Board's In-Circuit Debugger & Programmer Firmware

This section discusses how to update the firmware of the QSK62P Plus board's 2215 MCU, which is used to implement the board's in-circuit debugging (ICD) E8 and programming functionality. Under normal circumstances, you should never have to perform this update, unless the ICD MCU's memory becomes accidentally corrupted.

The 2215 MCU has a USB boot mode that can be used to program the MCU's user Flash area. The procedure to activate the boot mode to re-program the Flash firmware is described in the following steps.

1. Unplug the QSK30P/62P board unit from the USB cable.
2. Locate the J3 jumper (under the LCD) and, if not done already, solder a 2-pin, 2.54mm (0.100 mil) jumper header into it.
3. Shunt the jumper header with a 2.54mm (0.100 mil) jumper, thus connecting the BOOT signal to VCC. This will force the 2215 MCU to run in boot mode when it is powered up.
4. Plug the USB cable back in. The 2215 MCU will now be in boot mode and will communicate as a USB device to the PC. In boot mode, the 2215 MCU uses a different USB Driver than the In-Circuit Debugger/Programmer application, so you will need to load another USB Driver when doing this procedure for the first time. The Windows New Hardware Wizard should start automatically and guide you through the installation of the required USB driver. The driver is located in C:\Renesas\FOUSB\USB Drivers.
5. Obtain the 2215 code image from one of the Tech Support contacts elsewhere in this document. Using FDT basic, select the processor (2215 RUTE), add the image file, click "connect," and follow the instructions.
6. Unplug the USB Cable and remove the jumper connecting the BOOT signal to VCC. Reconnect the QSK62P Plus board to your PC with the USB cable.
7. After you reconnect the QSK62P Plus board to the PC, the E8 Programmer should show the target MCU device name you selected earlier.

Appendix C: Reference Manuals

Item	Title	Description
1	QSK62P Plus Quick Start Guide	Document that will help you get started on using the QSK62P Plus.
2	QSK62P Plus User's Manual	This document. Installation and operation guide for the QSK62P Plus.
3	QSK62P Plus Board Schematic	Schematic diagram for the QSK62P Plus Starter Kit board.
4	QSK3062P Plus Board BOM	Bill of materials for the QSK62P Plus Starter Kit board.
5	M16C/62P Group Data Sheet	Operation and Specifications for the M16C/62P MCU's.
6	M16C/30/60 Series C-Language Programming Manual	ANSI C-language programming guide for the M16C/30/60 series MCU.
7	M16C/30/60 Series Assembler Language Programming Manual	Assembler language programming guide for the M16C/30/60 series MCUs.
8	M16C/30/60 Series Software Manual	This document details the instruction set and timing information for the M16C/30/60 series CPU cores.
9	M16C/30/60 Series Sample Programs	This document details some sample programs and application notes for the M16C/30/60 series MCUs.
10	HEW User's Manual	Integrated Development Environment for Renesas' Tools.
11	AS30 User's Manual	Assembler Language Guide for AS30 assembly language compiler.
12	NC30 User's Manual	ANSI C-Compiler Guide for KNC30WA C-language compiler.
13	RTA-E8 User's Manual	User guide and operation manual for the ICD.

NOTE: The installer will copy all these manuals during installation. They can be viewed using the Document Descriptions file by clicking on Start > Programs > Renesas > QSK62P Plus > Document Descriptions.

Appendix D: Expansion Headers

The M30622F8 MCU on the QSK62P Plus target board is housed in a 100-pin QFP package. Pin 1 of the package is identified by the number '1' on the board's top silkscreen. Connectors JA1 to JA2 and JR1 to JR4, provide access to almost all of the MCU's pins. You can use them as test points to check MCU signals or to connect your own expansion board. The silkscreen identifying the connectors Is on the QSK62P Plus board. The following table shows the mapping of JA1 and JA2 pins to MCU pins and signal names.

Signal	JA1	M16C/62P
Vcc	1	16, 62
Gnd	2	14, 64
Vcc	3	16, 62
Gnd	4	14, 64
AVcc	5	99
AVss	6	96
Vref	7	98
P97	8	100
AN0	9	97
AN1	10	95
AN2	11	94
AN3	12	93
P93	13	4
P94	14	3
A8	15	63
A9	16	61
A10	17	60
A11	18	59
A12	19	58
A13	20	57
A14	21	56
A15	22	55
D13_INT3	23	75
-	24	-
IIC_SDA	25	30
IIC_SCK	26	29

Signal	JA7	M16C/62P
RESN	1	12
CON_Xin	2	-
Nmin	3	17
Gnd	4	14, 64
-	5	-
TXD0	6	35
INT0	7	20
RXD0	8	36
INT1	9	19
CLK0	10	37
UD	11	42
CTS_RTS	12	38
UP	13	-
UN	14	-
VP	15	-
VN	16	27
WP	17	-
WN	18	-
TA2Out	19	26
TA4Out	20	22
TA2In	21	25
TA4In	22	21
INT2	23	18
TRSTN	24	43
P92_TXD3	25	5
P91_RXD3	26	6

Appendix E: Board Schematic & BOM

The circuit board schematic and Bill-Of-Materials (BOM) are available as separate PDF documents.

They can be viewed via Start > Programs > Renesas > QSK62P > Board Hardware, or by browsing to the folder C:\Renesas\QSK62P\Docs and opening the files:

QSK62P+Schematic.pdf

Published QSK62P Plus BOM rC.pdf

Appendix F: Other Resources

1. For updates and other evaluation tools, and sample programs for the QSK62P Plus Kit, see: www.bnssolutions.com/QSK

2. Renesas microcontroller application notes and sample programs can be viewed at Renesas Technology America's website: http://www.renesas.com