



Powerful Processors – Easy to Use™

SKP16C62P

User's Manual

Rev. 1.0
December 2003

Table of Contents

1.0 Introduction	2
2.0 Contents of Product Package	3
3.0 Limited Guarantee and Support	4
4.0 System Connectivity.....	5
4.1 Host Computer Requirements	5
4.2 SKP16C62P Board	6
4.3 ICD.....	6
4.4 Software Development Tools.....	6
4.4.1 HEW	6
4.4.2 NC30WA C-compiler	6
4.4.3 KD30 Debugger	6
4.4.4 FoUSB (Flash-over-USB™) Programmer.....	6
5.0 Hardware.....	7
5.1 SKP16C62P Board.....	7
5.2 SKP16C62P Board Block Diagram.....	8
5.3 M16C/62P Group of Microcontrollers	8
5.4 SKP16C62P Board Jumper Configuration.....	9
5.4.1 JP1: MCU (U1) Power	9
5.4.2 JP2: Xout.....	9
5.5 LCD (Liquid Crystal Display).....	10
6.0 System Limitations	10
6.1 Kernel Introduction.....	10
6.2 Pin and Peripheral Limitations	11
6.3 Status After Reset.....	12
6.4 Register Operation Limitations	12
6.5 Limitations on Interrupts	12
6.6 Stop Mode or Wait Mode Limitations.....	13
6.7 User Program's Real-Time Capability.....	13
6.8 Performing Debug Using Symbols.....	13
7.0 SKP16C62P Board Specifications	14
7.1 Hardware Specifications	14
7.2 External Power Supply Requirements	14
7.3 Operating Environment.....	14
Appendix A. Troubleshooting Guide	15
A.1 USB Driver Problems	15
A.2 KD30 Problems	16
A.3 Installing Required Files Manually.....	18
Appendix B. Reference Manuals.....	19
Appendix C. Expansion Headers	20
Appendix D. SKP Board Dimensions.....	21
Appendix E. RTA-FoUSB-MON (ICD).....	22

1.0 Introduction

The SKP16C62P StarterKit Plus (SKP) is a low-cost environment for evaluating M16C/62P group of microcontrollers (MCU) and Renesas Technology America, Inc. software development tools. The kit comes with an integrated software development environment, HEW (IDE, C-compiler, assembler, and linker), KD30 Debugger, and FoUSB (Flash-over-USB™) Programmer.

A real-time, source-level debug environment is implemented using the KD30 debugging software with the RTA-FoUSB-MON Flash Programmer/In-Circuit Debugger (ICD). The Flash-over-USB™ (FoUSB) Programmer software, with the ICD, allows in-system programming of the M16C/62P flash MCU.

The ICD and firmware provide a convenient USB (Universal Serial Bus) interface between the SKP16C62P board and the host PC. This interface reduces resource requirements on the M16C/62P MCU, allows faster code downloads and, can also be used with many other Renesas Flash MCU's, SKP's, and user's target board.

2.0 Contents of Product Package

This section describes the contents of the SKP16C62P product package. When unpacking your SKP16C62P, please check to see that all products listed below are included.

2.1 SKP16C62P StarterKit Plus Product List

Table 2-1 lists the products included in the SKP16C62P.

Table 2-1 SKP16C62P Product List

Product Name	Quantity	Remark
SKP16C62P Board	1	M16C/62P SKP Board
RTA-FoUSB-MON (ICD)	1	KD30 Debugger/ FoUSB Programmer Interface Device
6" 10-Pin Target Cable	1	Connects SKP16C62P Board (J5) and the ICD
6' Mini USB Cable	1	Connects ICD to the Host PC
SKP CD-ROM	1	Auto-install program HEW (IDE, C-compiler, assembler, and linker) KD30 Debugger FoUSB Programmer Manuals Tutorials Sample programs

2.1.1 SKP CD-ROM

The CD-ROM contains the electronic manuals and software necessary for developing programs. Your computer must have Netscape Navigator® or Microsoft®'s Internet Explorer to view the help files and Acrobat Reader to view the manuals.

Insert the enclosed CD into your computer and SKP installer will auto-start. The SKP installer program will create a C:\MTOOL and C:\MTOOL\SKP16C62P folder on your machine. All development tools (HEW, KD30 Debugger, and FoUSB Programmer) can be found under C:\MTOOL. Documentation, sample code, and other SKP related files would be under C:\MTOOL\SKP16C62P folder.

If the SKP installer program does not start up, browse the CD's root folder and double-click on '**skp_installer.exe**' to start installation. If you continue having problems on the install, please read the '**Install FAQ.txt**' file on the root folder of the CD.

3.0 Limited Guarantee and Support

Renesas Technology America, Inc., warrants the SKP16C62P to be free from component or assembly defect 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 apply, 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 error or omissions and reserves the right to make changes without further notice.

“Flash-Over-USB” is a trademark of Renesas Technology America, Inc. All trademarks are the property of their respective owners.

4.0 System Connectivity

The following lists the hardware and software products required for using the SKP16C62P StarterKit Plus.

- Host Computer (supplied by user)
- SKP16C62P Board
- RTA-FoUSB-MON (ICD)
- Mini USB Cable
- Target Cable
- Software Tools (HEW IDE, NC30 Compiler/Linker, KD30 Debugger, FoUSB Programmer)

Figure 4-1 shows the system connectivity for the SKP16C62P.

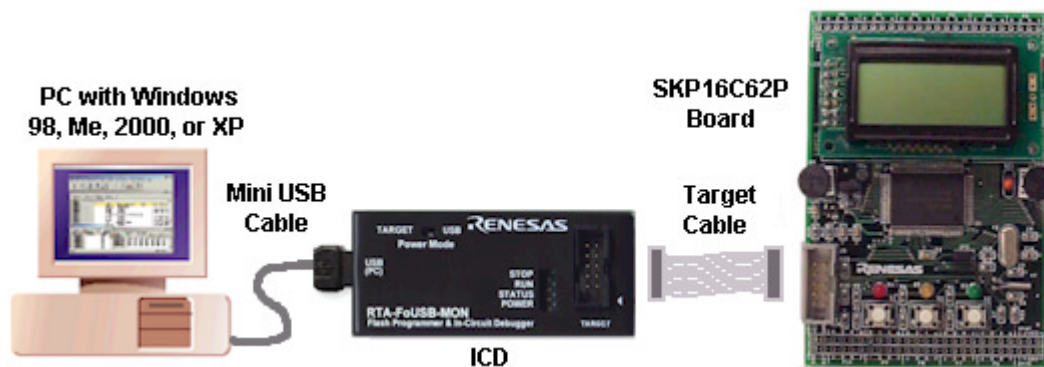


Figure 4-1 SKP16C62P System Connectivity

4.1 Host Computer Requirements

The minimum requirement to be able to use the software that comes with the SKP16C62P is a PC with a USB port and Microsoft Windows 98, ME, 2000, or XP.

Note: For Windows 95 and NT, that do not support USB, please contact Renesas representative or email techsupport.rta@renesas.com for options.

4.2 SKP16C62P Board

The SKP16C62P board provides an evaluation and development environment for the M16C/62P group of MCU. It has pushbutton switches, LED's and LCD for user interface. Standard connector ports are available to expand the range of applications through the use of expansion boards, etc.

4.3 ICD (RTA-FoUSB-MON)

The ICD provides a plug-and-play debugging and programming interface to the SKP16C62P board by using the host computer's Universal Serial Bus (USB). The USB port also provides power to the SKP16C62P board and ICD thereby eliminating the need for an external power supply.

4.4 Software Development Tools

Aside from SKP documentation and sample code, the SKP installer program also installs the development tools. For details, please see the quick start guide on how to run the SKP installer.

4.4.1 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, and editor.

4.4.2 NC30WA Entry Version

The NC30WA Entry version C-compiler included in the SKP has no time expiration, but has the following limits:

1. The software comes without warranty.
2. The compiler always compile with “-fansi” option.
3. The compiler optimization settings is fixed and cannot be changed
4. The ‘-finfo’ option for other software tools such as stack viewer, etc. is not supported.

If your development needs exceed the above limitations, please go to the following web site http://www.renesas.com/eng/products/mpumcu/toolhp/datsheet/m16c_e/nc30wa.htm and download the full version (4 calendar month trial).

4.4.3 KD30 Debugger

KD30 is a remote debugger that runs on the host PC. While communicating with a kernel (i.e. a ROM monitor program) on the target MCU through the ICD, KD30 provides a highly efficient evaluation environment. KD30 features include:

- Source-line debug for assembly language, structured assembly language, C language
- Run command with 8 breakpoints for M16C/62P (will depend on the M16C MCU)
- RAM monitor function
- C variable “watch” window

4.4.4 FoUSB (Flash-over-USB™) Programmer

User programs can run on the SKP16C62P board without the host computer running KD30 by programming the on-board M16C/62P flash MCU with firmware using FoUSB Programmer. It can also be used to program other Renesas’ M16C Flash MCU’s and 740 series MCU’s.

5.0 Hardware

5.1 SKP16C62P Board

Figure 5-1 shows the SKP16C62P Board with major components identified.

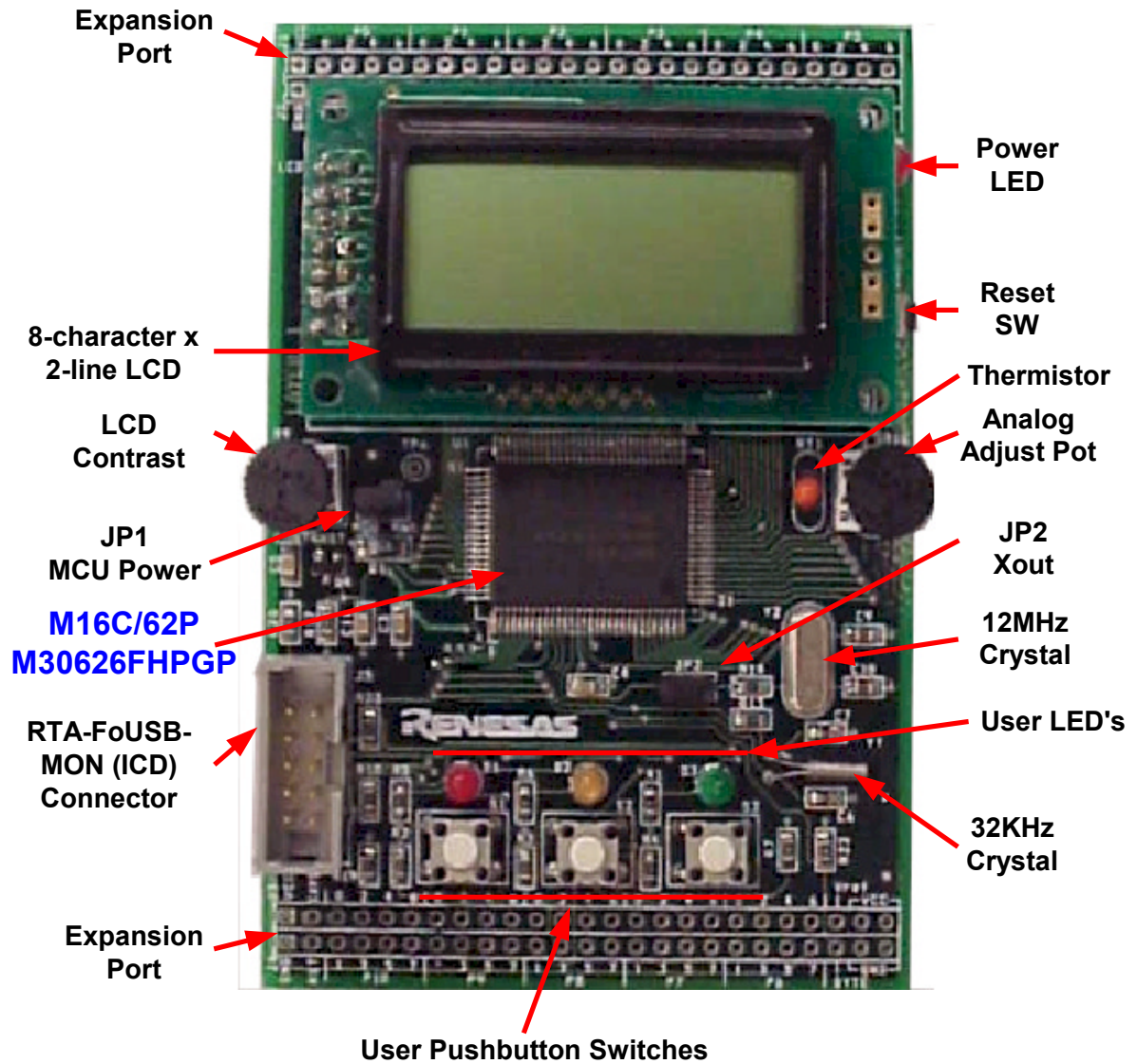


Figure 5-1. SKP16C62P Board

5.2 SKP16C62P Board Block Diagram

The SKP16C62P board incorporates an M30626FHPGP (100-pin QFP) from the M16C/62P group of microcontrollers designated as U1. Figure 5-2 shows the SKP16C62P block diagram.

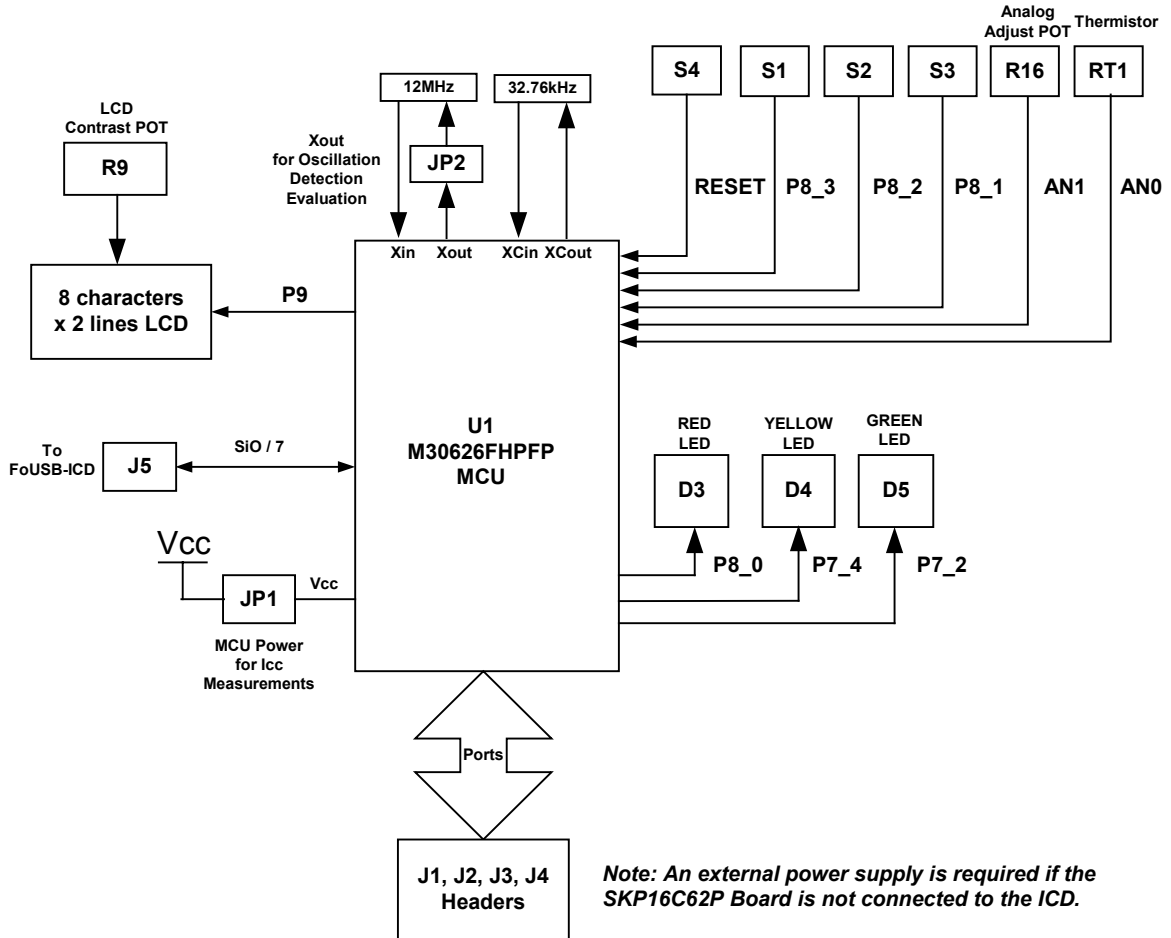


Figure 5-2. SKP16C62P Block Diagram

5.3 M16C/62P Group of MCUs

M3062x group of 16-bit single-chip, flash microcontroller (MCU) is built using high-performance silicon gate CMOS process with an M16C/60 series CPU core. The datasheet for the M16C/62P group of microcontrollers can be found under C:\MTOOL \SKP16C62P\Docs folder of your PC after SKP software installation or from the Start menu (Start > Programs > RENESAS-TOOLS > SKP16C62P > Document Descriptions).

5.4 SKP16C62P Board Jumper Configuration

5.4.1 JP1: MCU (U1) Power

JP1 is used to connect the power/Vcc pins of the M16C/62P MCU to the Vcc of the board. It can be used to measure current/power consumption of the MCU during various modes of operation. For normal operations, JP1 must be shorted.

JP1 is shorted by default.

5.4.2 JP2: Xout

JP2 is used to connect the output of the MCU's internal amplifier to the crystal for oscillation. It can be used to disconnect Xout when an externally driven clock is connected to Xin. For normal operations, JP2 must be shorted.

JP2 is shorted by default.

5.4.3 Default Jumper Setting

Jumper	JP1 MCU Power	JP2 Xout
Setting	Shorted	Shorted

5.5 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 SKP16C62P provides sophisticated debugging features at a low cost but it does have some limitations when used with the KD30 Debugger and ICD. Section 6.1 introduces the kernel (ROM monitor) program and what it is for. The limitations when this kernel is running with the user program are listed in table 6-1.

Table 6-1 Systems Limitations (when used with KD30)

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
Debugger Limitations	6.7 Stop or Wait Mode Limitations
	6.8 User Program's Real-time Capability

6.1 Kernel (ROM Monitor) Introduction

During debug (used with the KD30 debugger), a small program, called a kernel, is downloaded to the M16C/62P. The kernel communicates with the KD30 Debugger through the ICD about MCU status during user code debugging operations.

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

After starting KD30, the ICD downloads the kernel to the M16C/62P if it does not exist (e.g. blank device or programmed with FoUSB Programmer). After downloading the kernel, KD30 opens the Program Window and the M16C/62P is ready for downloading code.

Connecting the ICD without starting KD30 will not affect the lines connected between the ICD and the M16C/62P – the ICD keeps the lines in high-impedance state. The ICD only drives the pins after KD30 or FoUSB Programmer is started.

After program debug and verification, you can then create and download a binary, Intel (.hex) or Motorola (.mot), file to the M16C/62P. This operation erases the kernel and only leaves the user program.

6.2 Pin and Peripheral Limitations

SIO/UART1 pins are used for communication between the SKP16C62P board kernel and KD30 Debugger through the ICD. Do not connect these pins to any other circuit, as UART1 cannot be used in the user program. For details, please see ICD (RTA-FoUSB-MON) User Manual on Target M16C ROM Monitor Resources or related ICD application notes.

6.3 Memory Map

The amount and memory locations of the kernel program, to communicate with the KD30 and ICD, are using on the M16/62P on the SKP16C62P board is shown on Figure 6-2.

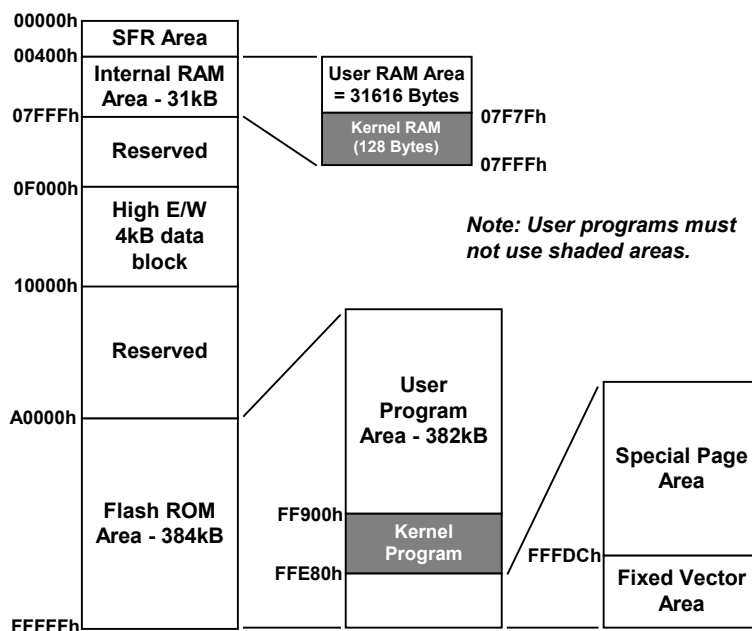


Figure 6-2 M30626FHPGP Memory Map with the Kernel Program

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 SKP16C62P starter board is reset. The register status depends on the microcomputer'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 07F84 ₁₆
UART1	Set for external clock synchronous mode

6.5 Register Operation Limitations

Table 6-3 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 Interrupt Stack Pointer	Range 7B80H – 7BFFH is used by the kernel
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 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 ~ FFFE8BH	User inhibited
Single-step	FFFECH ~ FFFE8FH	User inhibited
Watchdog Timer	FFFF0H ~ FFFF3H	User available (Note 1)
DBC	FFFF4H ~ FFFF7H	User inhibited
NMI	FFFF8H ~ FFFF8BH	User available
RESET	FFFFCH ~ FFFFCH	Reset vector (Note 2)

NOTES:

- (1) The Watchdog Timer vector is shared with the oscillation stop and voltage detection interrupt. The vector is available for oscillation stop and voltage detection interrupts, but not 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 the STOP or WAIT instructions in your program.

6.8 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 may not be serviced properly. Also, 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 in KD30. 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 or not this behavior is acceptable.

6.9 Performing Debug Using Symbols

Normally when a new project is started using HEW, debugging symbols are enabled. If you are unable to debug your program using symbols in KD30, 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 Options pull-down menu.
- Click on Link tab.
- Select [Output] under the [Category] list box.
- Click on checkbox for **[-g]** 'Outputs source debug information....'
- Click on [OK] button

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

7.0 SKP16C62P Board Specifications

7.1 Hardware Specifications

Table 7-1 lists the specifications of the SKP16C62P Board.

Table 7-1. SKP16C62P Board Specifications

Item	Specification
MCU	M30626FHPGP
Clocks	Main Clock: 12MHz crystal (16MHz max), ceramic resonator, ring oscillator, or PLL (24MHz max) Sub Clock: 32.768KHz crystal
Memory (Internal with KD30 + ICD)	RAM: 31kB (31616 Bytes) High E/W Data Block: 4kB Flash ROM: 382kB (391,423 Bytes)
Connectors	[J5]: Serial interface connector (UART1 for KD30 + ICD) [J1-J4]: 50 pin (user supplied) header (for user target connection)
Jumpers	[JP1]: MCU Power for Icc Measurements [JP2]: Xout to main crystal Y2
Switches	[S1]: pushbutton (connected to P8_3) [S2]: pushbutton (connected to P8_2) [S3]: pushbutton (connected to P8_1) [S4]: pushbutton (connected to Reset)
LED's	[D1] (Red): User output (connected to P8_0) [D2] (Yellow): User output (connected to P7_4) [D3] (Green): User output (connected to P7_2)
LCD	2-line x 8-character LCD with KS0066 controller IC

7.2 Power Supply Requirements

The SKP16C62P Board will draw 35mA (max). With the ICD, the current draw will be about 75mA.

7.3 Operating Environment

Table 7-3 lists the environmental conditions for using and storing the SKP16C62P board. When storing the board, place it in a conductive bag and then in the packing box your product was shipped in from the factory.

Table 7-2. Operating Environment

Environmental Condition	Ambient Temperature	Ambient Humidity
Operating	0 - 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 software (and drivers) and while running the KD30 or FoUSB Programmer applications. This section also discusses the countermeasures and solutions to resolve these problems.

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

A.1 USB Driver Problems

This section discusses the usual problems with the driver installation and how to fix it. The most common problem encountered is that Windows did not properly install the driver and so the ICD is not recognized. This may also cause the device status to indicate that the device is not working properly. An indication of this problem is the ICD status yellow LED - it blinks about 2-3 times a second. When the driver is installed properly, the yellow LED should only blink every second.

Before trying the following steps, try restarting your PC and see if this resolves the problem. You can check the status using the Device Manager. If the ICD appears under the Universal Serial Bus Controllers with NO red X or yellow exclamation point, the driver was installed properly.

If you are using Windows 2000 or Windows XP, you need Administrator privileges to install the drivers.

For cases where the 'Device Status' states the device is not working properly, please try the following:

- Double-click on 'Renesas USB-Monitor' and a Renesas USB-Monitor Properties dialog box appears.
- Click on 'Driver' tab and click on 'Update Driver' button.
- Select 'Display a list...' and click on 'Have Disk' button.
- Specify and locate the C:\MTOOL\FoUSB\USB Drivers folder on your PC and install 'usbmon.sys' driver.
- If this process does not work, please try the instructions below.

For cases where in the driver was not installed properly by Windows (Windows 98, Windows 2000) or not listed in the Device Manager > Universal Serial Bus Controllers, please try the following:

- Unplug the USB Cable so Windows removes driver from memory.
- Delete the driver 'usbmon.sys' from '\WINNT\SYSTEM32\DRIVERS\ folder in Windows 2000 or \WINDOWS\SYSTEM32\DRIVERS folder in Windows 98.
- Plug in ICD and try installing the driver as written above, but use the driver from the CD. The folder is C:\MTOOL\FoUSB\USB Drivers.

A.2 KD30 Problems

This section discusses the cause of the problem and countermeasures to resolve it. The common problems encountered with KD30 are:

- USB option cannot be selected from the initialization screen
- Cannot connect to target
- KD30 already exists
- Issues that may come up during debug operations

A.2.1 USB Option Unavailable

When USB cannot be selected from KD30's Init dialog box, you might be using an old version of KD30 that does not support USB. Uninstall this version of KD30 and install latest version of KD30 from the \Tools\KD30 directory of the SKP CD.

A.2.2 Cannot Connect to Target

When the message 'Can't connect with the target' is displayed after KD30 startup, there are several reasons that may cause this message to appear. Each cause and the corresponding countermeasure are discussed below.

- The SKP is not connected correctly.

Please connect the ICD to your PC. Note: Regardless of whether the ICD is bus or target-powered, please connect target board to ICD board first before plugging the USB cable to your PC. Please see section 4 on system (SKP16C62P) connectivity.

- The ICD has no power (Power LED on ICD is off).

Please ensure that the Power Mode switch on the ICD is on the 'USB' side. In 'Target' position, the ICD is powered from the target side.

- USB was not selected on the Init dialog box.

Please select 'USB' from the Init dialog box that is displayed right after you start KD30.

- The selected MCU on the ICD board and the actual target MCU (M16C/62P) do not match.

Close the error message by clicking on 'OK' button, and then click on the 'Cancel' button of the KD30 Init window to close KD30. Make sure you select 'M30626FHP.mcu'. If the MCU loaded on the ICD is different, KD30 will re-program the ICD to match it.

- The target MCU is damaged.

Try a different target board and see if KD30 will come up as you may have a damaged board or MCU.

A.2.3 KD30 Already Exist

When a message 'KD30 already exists' is displayed, the usual cause of this problem is that the KD30 application was not properly closed. Please check if KD30 is already running by looking at your task bar. If KD30 cannot be found there, bring up Task Manager (press CTRL-ALT-DEL once and click on 'Task Manager' button). Select KD30.exe on the 'Processes' list and click on 'End Process' to terminate KD30.

A.2.4 Issues that may come up During Debug Operations

While using KD30 to debug user code, some issues may come up because the limitations discussed in section 6 were not satisfied. These common issues are listed on table A.2, including the countermeasures.

Table A.2. Problems while using KD30

Problem	Possible Cause/s and Solution
After stepping a few instructions, KD30 hangs	<ul style="list-style-type: none"> Changes made to UART1 SFR's.
Breakpoints do not seem to work	<ul style="list-style-type: none"> KD30 is in "FreeRun" mode. Change the RUN mode to "Sampling" from the "Init" window (Environment > Init).
KD30 locks up (cannot stop program) or Communication error message is displayed.	<ul style="list-style-type: none"> Changes made to UART1 SFR's. Ensure no limitations in Section 7 were violated. Re-initialize the system without closing KD30. See note below. Do a hardware reset. User-program runaway may be corrupting kernel RAM or interrupt vectors, flags, etc. Close KD30, hit S1 on the SKP16C62P to reset the board, and then restart KD30.
Download Issues	<ul style="list-style-type: none"> Filenames or directory names contain spaces or special characters. HEW project 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 HEW user's manual.

To re-initialize the system without closing KD30, try the following:

- Press the [OK] button in the error dialog box to close it.
- When an Exit dialog box appears, press the [Cancel] button to close it.
- Hit reset button on the SKP16C62P board.
- Press KD30 reset button.

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

Note: If it has been identified that there are problems with the ICD, please see the ICD's (RTA-FoUSB-MON) user's manual and troubleshooting.

A.3 Installing Required Files Manually

This section discusses how to install the SKP16C62P software and development tools manually (i.e. when not using the SKP16C62P installer).

- (1) Run the KD30 installer, '**KD30V320R1b_e.exe**', from the **\Tools\KD30** folder on the SKP CD.
- (2) Run the FoUSB installer, '**fousb_V2.00.exe**', from the **\Tools\FoUSB** folder on the SKP CD.
- (3) Run the HEW installer, '**setup.exe**', from the **\Tools\HEW** folder on the SKP CD.
- (4) Run the NC30 installer, '**nc30wa_entry.exe**', from the **\Tools\NC30WA** folder on the SKP CD.
- (5) Create an **SKP16C62P** folder under C:\MTOOL on your PC and then copy the directories from the **\SKP16C62P\files** folder on the SKP CD to this folder.

This completes the software installation but the drivers for the ICD also needs to be installed. We need to copy driver files (inf and sys) files and executables to certain Windows directories.

Open up the Windows Explorer window and browse the C: directory. One of the directories is **C:\MTOOL\FoUSB\USB Drivers**, which contains the driver files necessary to use the ICD. Plug the ICD board to your PC using the enclosed mini USB cable.

Follow the instructions on installing new hardware and when Windows asks for the drivers or location of the drivers, select browse or specify **C:\MTOOL\FoUSB\USB Drivers**, and then continue with the new hardware installation.

NOTE: If you are using Windows 2000, you need Administrator privileges to install the drivers.

If you still encountering problems on installing the drivers, you can try the following.

(1) Windows 2000

- i. Copy the **fousb.inf** and **usbmon.inf** files from **\USB Drivers** folder to **\WINNT\INF** folder.
- ii. Copy the **fousb.sys** and **usbmon.sys** files from **\USB Drivers** folder to **\WINNT\SYSTEM32\drivers** folder.

(2) Windows 98

- i. Copy the **fousb.inf** and **usbmon.inf** files from **C:\MTOOL\FoUSB\USB Drivers** folder to **\WINDOWS\INF** folder.
- ii. Copy the **fousb.sys** and **usbmon.sys** files from **C:\MTOOL\FoUSB\USB Drivers** folder to **WINDOWS\SYSTEM32\drivers** folder.

Appendix B. Reference Manuals

Item	Title	Description
1.	SKP16C62P Quick-Start Guide	Document that will help you get started on using the SKP16C62P.
2.	SKP16C62P User's Manual	Installation and operation guide for the SKP16C62P.
3.	SKP16C62P Board Schematic	Schematic diagram for the SKP16C62P Starter Kit board.
4.	SKP16C62P Board BOM	Bill of materials for the SKP16C62P Starter Kit board.
5.	M16C/62P Group Data Sheet	Operation and Specifications for the M16C/62P MCU's.
6.	M16C/20/60 Series C-Language Programming Manual	ANSI C-language programming guide for the M16C/20/60 series MCU.
7.	M16C/20/60 Series Assembler Language Programming Manual	Assembler language programming guide for the M16C/20/60 series MCUs.
8.	M16C/20/60 Series Software Manual	This document details the instruction set and timing information for the M16C/20/60 series CPU cores.
9.	M16C/20/60 Series Sample Programs	This document details some sample programs and application notes for the M16C/20/60 series MCUs.
10.	HEW User's Manual	Integrated Development Environment for Renesas' Tools.
11.	KD30 User's Manual	Easy-to-use source and assembly debugger.
12.	AS30 User's Manual	Assembler Language Guide for AS30 assembly language compiler.
13.	NC30 User's Manual	ANSI C-Compiler Guide for KNC30WA C-language compiler.
14.	RTA-FoUSB-MON User's Manual	User guide and operation manual for the ICD.

Appendix C. Expansion Headers

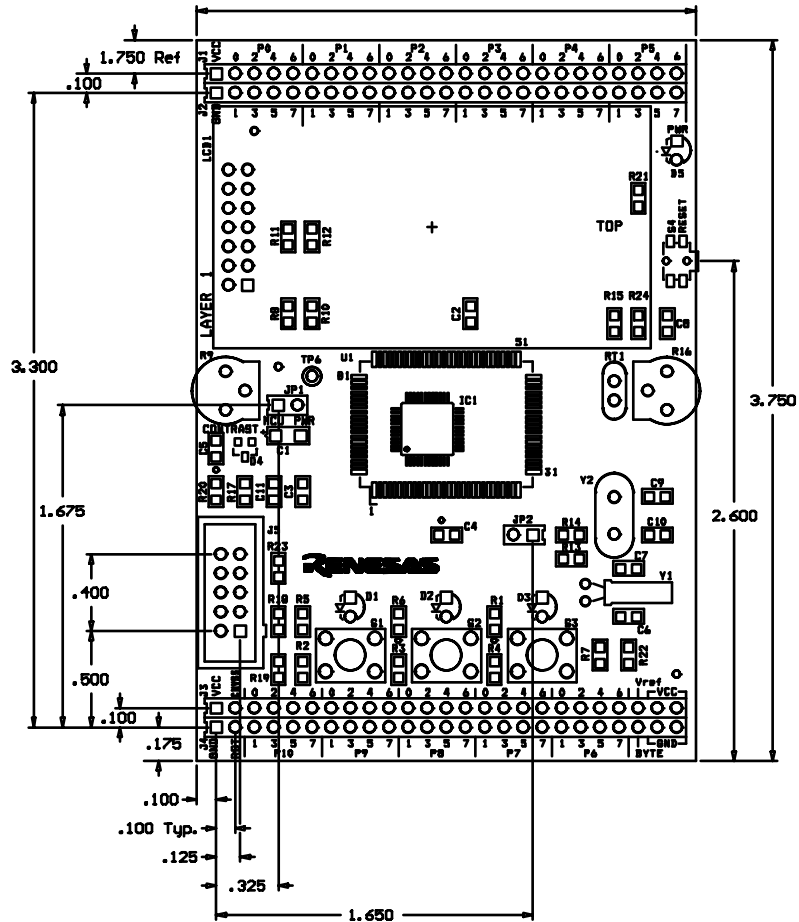
J1 Pin	Function
1	Vcc
2	P0_0, AN00, D0
3	P0_2, AN02, D2
4	P0_4, AN04, D4
5	P0_6, AN06, D6
6	P1_0, D8
7	P1_2, D10
8	P1_4, D12
9	P1_6, INT4, D14
10	P2_0, AN20, A0
11	P2_2, AN22, A2
12	P2_4, AN24, A4
13	P2_6, AN26, A6
14	P3_0, A8
15	P3_2, A10
16	P3_4, A12
17	P3_6, A14
18	P4_0, A16
19	P4_2, A18
20	P4_4, CS0
21	P4_6, CS2
22	P5_0, WRL, WR
23	P5_2, RD
24	P5_4, HLDA
25	P5_6, ALE

J2 Pin	Function
1	GND
2	P0_1, AN01, D1
3	P0_3, AN03, D3
4	P0_5, AN05, D5
5	P0_7, AN07, D7
6	P1_1, D9
7	P1_3, D11
8	P1_5, INT3, D13
9	P1_7, INT5, D15
10	P2_1, AN21, A1
11	P2_3, AN23, A3
12	P2_5, AN25, A5
13	P2_7, AN27, A7
14	P3_1, A9
15	P3_3, A11
16	P3_5, A13
17	P3_7, A15
18	P4_1, A17
19	P4_3, A19
20	P4_5, CS1
21	P4_7, CS3
22	P5_1, WRH, BHE
23	P5_3, BCLK
24	P5_5, HOLD
25	P5_7, RDY, CLKOUT

J3 Pin	Function
1	Vcc
2	CNVss
3	P10_0, AN0
4	P10_2, AN2
5	P10_4, AN4, KI0
6	P10_6, AN6, KI2
7	P9_0, TB0in, CLK3
8	P9_2, TB2in, Sout3
9	P9_4, TB4in, DA1
10	P9_6, Sout4, ANEX1
11	P8_0, TA4out, U
12	P8_2, INT0
13	P8_4, INT2
14	P8_6, XCOout
15	P7_0, TxD2, SDA2, TA0out
16	P7_2, CLK2, TA1out, V
17	P7_4, TA2out, W
18	P7_6, TA3out
19	P6_0, CTS0
20	P6_2, RxD0, SCL0
21	P6_4, CTS1, CLKS1
22	P6_6, RxD1, SCL1
23	Vref
24	Vcc
25	Vcc

J4 Pin	Function
1	GND
2	RESET
3	P10_1, AN1
4	P10_3, AN3
5	P10_5, AN5, KI1
6	P10_7, AN7, KI3
7	P9_1, TB1in, Sin3
8	P9_3, TB3in, DA0
9	P9_5, CLK4, ANEX0
10	P9_7, ADTRIG, Sin4
11	P8_1, TA4in, /U
12	P8_3, INT1
13	P8_5, NMI
14	P8_7, XCIin
15	P7_1, RxD2, SCL2, TB5in
16	P7_3, CTS2, TA1in, /V
17	P7_5, TA2in, /W
18	P7_7, TA3in
19	P6_1, CLK0
20	P6_3, TxD0, SDA0
21	P6_5, CLK1
22	P6_7, TxD1, SDA1
23	BYTE
24	GND
25	GND

Appendix D. SKP Board Dimensions



Appendix E. RTA-FoUSB-MON (ICD)

For details on how to use the ICD, please see RTA-FoUSB-MON user's manual.