User's Manual
TK-78K0/KE2C

Date published: June 2010
Rev. 1.0
© TESSERA TECHNOLOGY INC. 2010
Printed in Japan

Windows and Windows XP are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

- The information is subject to change without notice.
- No part of this document may be copied or reproduced in any form or by any means without prior written consent of TESSERA TECHNOLOGY INC.. TESSERA TECHNOLOGY INC. assumes no responsibility for any errors that may appear in this document.
- TESSERA TECHNOLOGY INC. does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of TESSERA TECHNOLOGY INC. products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of TESSERA TECHNOLOGY INC. or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. TESSERA TECHNOLOGY INC. assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.

CAUTION

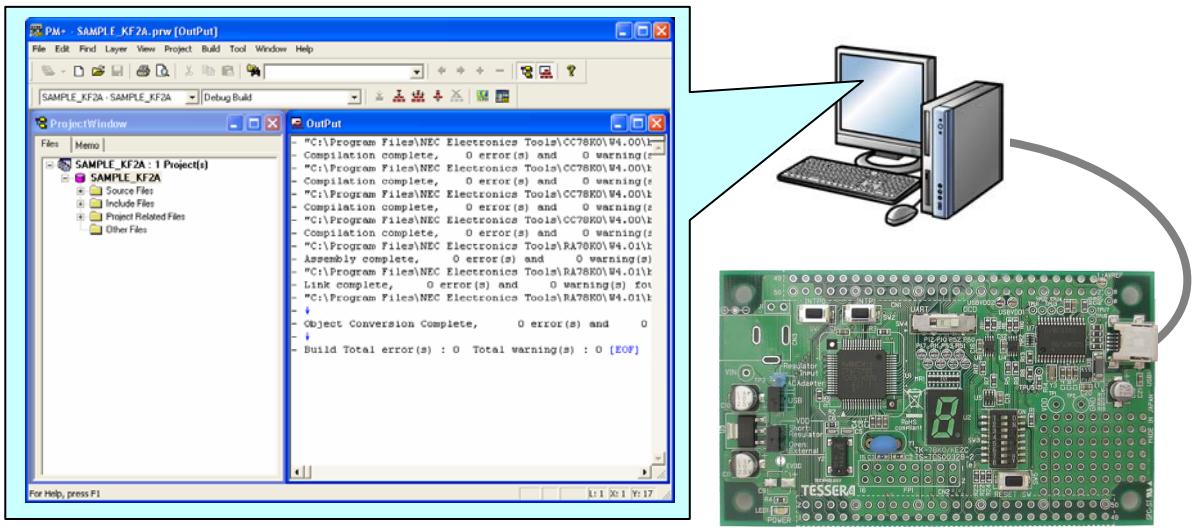
- Do not give any physical damage to this equipment such as dropping
- Do not superimpose voltage to this equipment.
- Do not use this equipment with the temperature below 0℃ or over 40℃.
- Make sure the USB cables are properly connected.
- Do not bend or stretch the USB cables.
- Keep this equipment away from water.
- Take extra care to electric shock.
- This equipment should be handled like a CMOS semiconductor device. The user must take all precautions to avoid build-up of static electricity while working with this equipment.
- All test and measurement tool including the workbench must be grounded.
- The user/operator must be grounded using the wrist strap.
- The connectors and/or device pins should not be touched with bare hands.

Contents

Introduction

TK-78K0/KE2C is the evaluation kit for development using "78K0/Kx2", NEC Electronics 8bit all flash microcontroller.
The user only needs to install the development tools and USB driver, and connect the host machine with the target board to start the code development, build, monitoring the output, and debugging code.
(This demonstration kit uses the on-chip debug feature from the microcontroller itself, without emulator connection)

Configuration for Debugging



Overview This manual consists of the following contents.

Read chapter 1 and 2 first for installing the development tools and using the sample programs.

Read chapter 3-5 for customizing the sample programs and the hardware.

Chapter 1: Preparations
Install the development tools

Chapter 2: Experiences
Experience the basic operations of integrated development environment (PM+) and integrated debugger (ID78K0-QB) with using sample programs.

Chapter 3: Hardware Specifications
Explain the hardware of TK-78K0/KE2C

Chapter 4: Troubleshooting
Describe how to solve troubles you may face, such as errors when starting the integrated debugger (ID78K0-QB)

Chapter 5: Other Information
Introduce other information, such as how to create a new workspace (project) on integrated development environment (PM+), how to register additional source file, and some useful tips of the integrated debugger. The circuit diagrams of demonstration kit are included in this chapter.

Reader This manual is intended for development engineers who wish to become familiar with the development tools for the 78K0.
It is assumed that the readers have been familiar with basics of microcontrollers, C and Assembler languages, and the Windows™ operating system.

Purpose This manual is intended to give users an understanding of the features, hardware configurations, development tools for the 78K0.

CHAPTER 1　Preparation

This chapter describes following topics:

- Overview and installation of development tools
- Installation of development tools
- Overview and preparation of sample programs

Users can experience the development flow such as coding, build, debugging, and test, by using the development tools bundled with TK-78K0/KE2C.

1.1 Development Tools / Software

● Device file DF780765 V2.21

A device file contains device specific information. So, users need a device file to use the development tools.

● Integrated Development Environment (IDE) PM+ V6.30

The IDE works on Windows operation system.
Users can develop a system efficiently by using the editor with idea processor function, compiler, and debugger.

● C Compiler CC78K0　W4.00 (code size limited version)

C compiler for the 78K0 microcontrollers. The object code size is limited to 32 Kbyte.
This compiles C code for 78K0 and ANSI-C code program into assembler code.
This produces object code and linker.
If you access the extended function register with CC78K0 W4.00, then configure below.

```
Example ) Define the "EFR".
#define   RMGPLS   (*((volatile unsigned char *)0xfa44))
void func(){
        RMGPLS = 0x12;
  }
```

● Assembler RA78K0　W4.01 （code size limited version）

Assembler for the 78K0 microcontrollers. The object code size is limited to 32 Kbyte.
This convert the assembler code for 78K0 into object program. The object program will be used for debugger.
If you access the extended function register with RA78K0 V4.01, then configure below.

```
EFRNM EQU 0F898H
C1      CSEG
        MOV A, #12H
        MOV !EFRNM, A
END
```

● 78K0 Integrated Debugger ID78K0-QB V3.20

This is the tool for debugging the object program generated by C compiler and assembler. The debugger enables to do C source level debugging. With the debugger, you can debug the code easily and efficiently by refering and changing variables, using step-in debuging function, and so on.

●WriteEZ5

This is the tool to write HEX file on microcontroller built-in memory without using the debugger (ID78K0-QB).

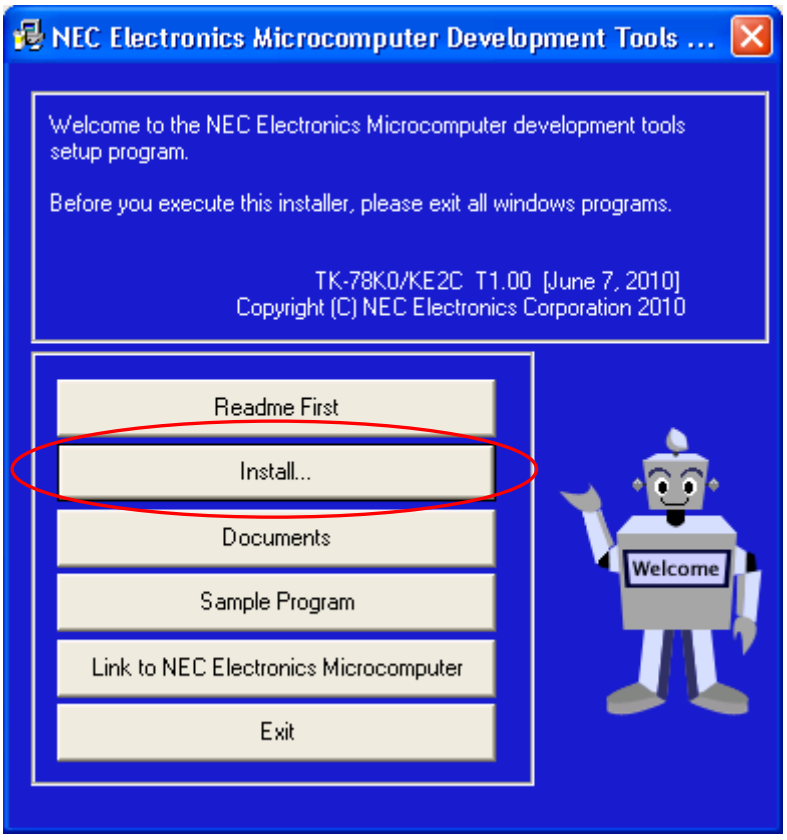
1.2 Installation of Development Tools

1.2.1 Installation Package

The attached CD-ROM includes the development tools and documentations. Users can use the installer to install those development tools and documentations.

1.2.2 Installation of Development Tools

① Please insert the CD-ROM in the drive. The installer will show up automatically. If it does not start automatically, please initiate it by double clicking the SETUP.EXE.

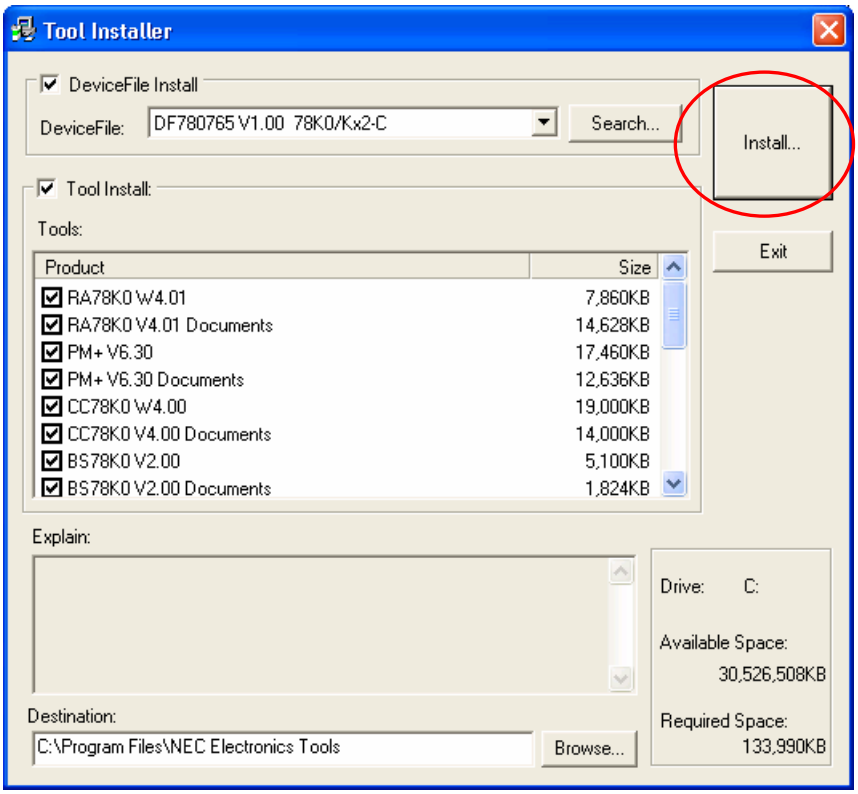


② Click Install... button.

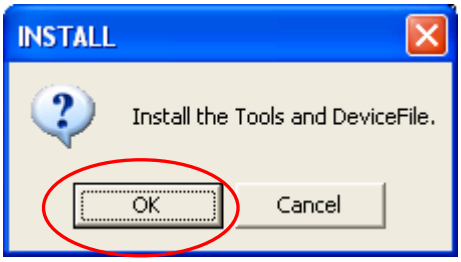
③ "Tool Installer" dialog box is opened.
Select products that you need to install.
(as default, all the products that you need to use the TK-78K0/KE2C are selected.)
"Explain" area displays an explanation of the selected product.
To change the installation destination, click Browse··· .

When all the settings are completed, click Install··· .

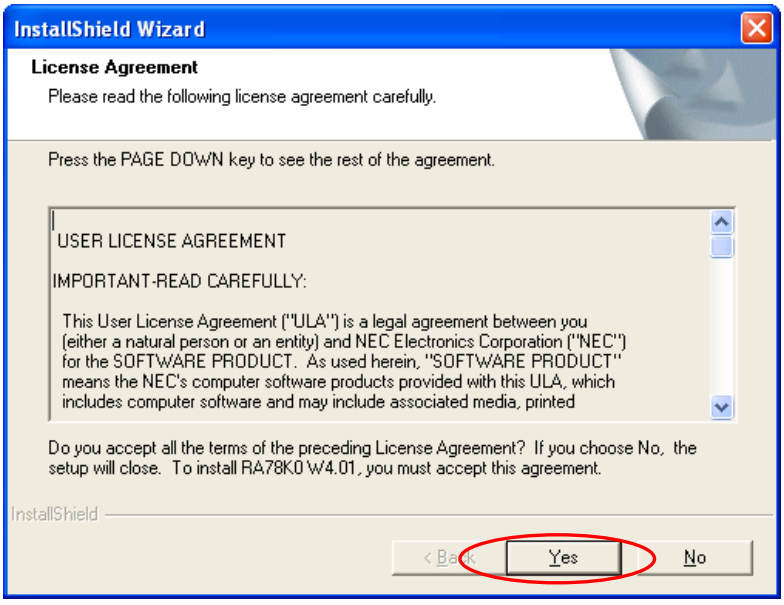
* In this document, it is assumed that users install the programs under "NEC Electronics Tools" directory (default installation directory). Users can find the tools by selecting "Start Menu" -> "Programs" -> "NEC Electronics Tools".



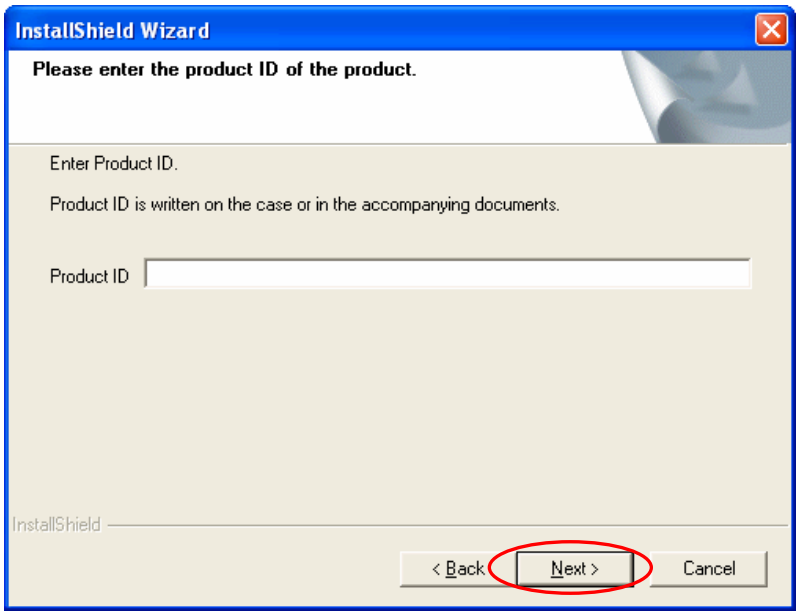
④ Click OK when "Install" comfirmation dialog box is opened.



⑤ Read "software license agreement" and click [Yes] for continuing the installation.
To stop the installation, click [No] .

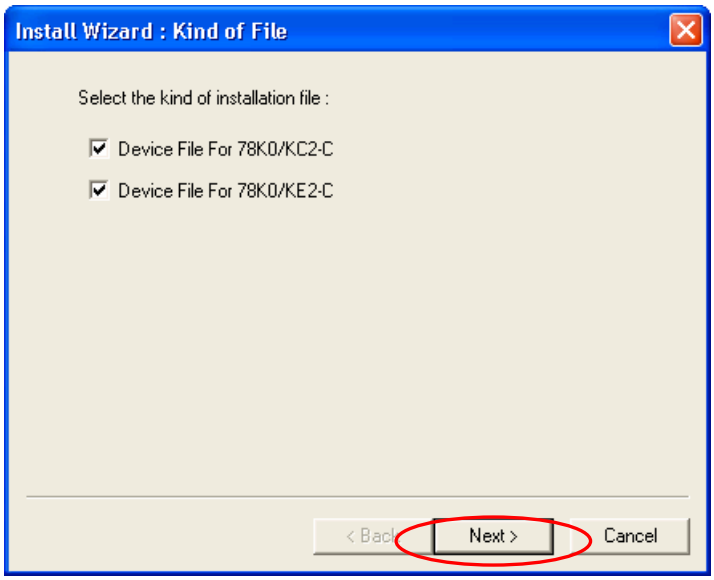


⑥ Enter the product ID, and click [Next] .
＊The product ID is available on the other sheet.

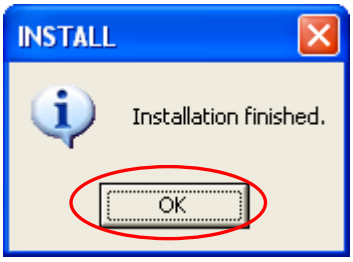


⑦ It starts copying the files.

⑧ Click Next when "Select Files" installation wizard dialog opened.



⑨ When the installation is completed, the following dialog opens.
Click OK .



⑩ "NEC Electronics Starter Kit Virtual UART" USB driver must be installed on PC before you connect to TK-78K0/KE2C.

Install the USB driver by referring "1.3 Installation of USB Driver".

Notes on the installation authority

To install this tool in Windows 2000 or XP, the authority of an administrator is necessary. Therefore, please login as an administrator.

Notes on the install-directory

Please do not use 2-byte characters, such as umlaut in the directory name, where the product is to be installed.

Note on the version of Windows

If the language of the Windows is not English, a file transfer error during installation might be observed. In this case, please abort the installation in the language, and re-install it in an English version of Windows.

The identical problem may be observed, if a language other than English is specified as the system language in the "Regional Settings Properties" tab.

Limitation

Assembler RA78K0 and C compiler CC78K0 limit the object size to 32 Kbyte.

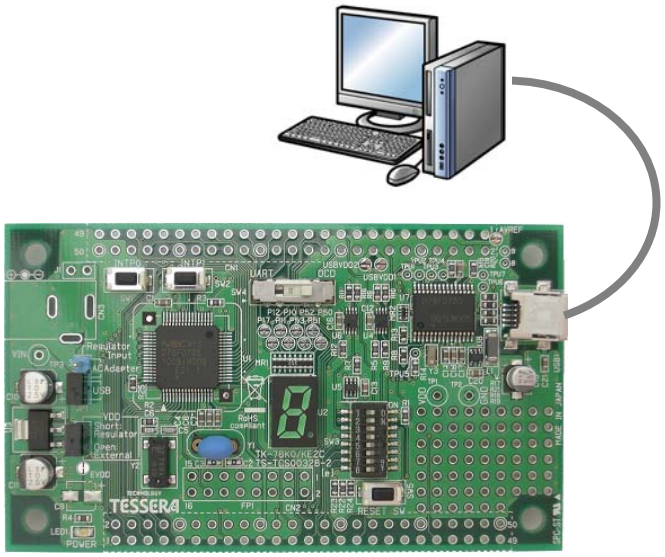
1.3 Installation of USB Driver

"NEC Electronics Starter Kit Virtual UART" USB driver must be installed on PC before you start using the TK-78K0/KE2C.
Please, follow the instruction below to install the driver.

"Starter Kit USB Driver" must be installed on the PC. If not, please refer to "1.2 Installation of Development Tools" to install the driver first.

CAUTION:

Do not use a USB hub for connecting TK-78K0/KE2C.



Depending on the version of Windows OS, the installation will be differed. Please check your Windows version, and follow the instructions

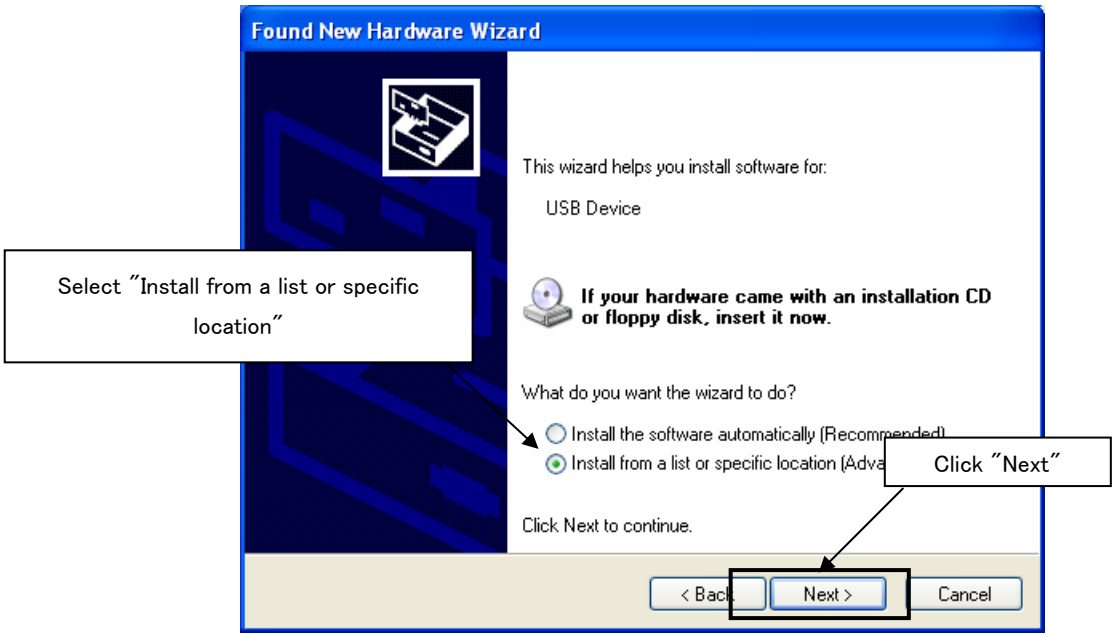
- Windows XP -> "1.3.1 Installation on Windows XP"
- Windows 2000 -> "1.3.2 Installation on Windows 2000"

After the installation, go to "1.3.3 Completion of USB Driver Installation"

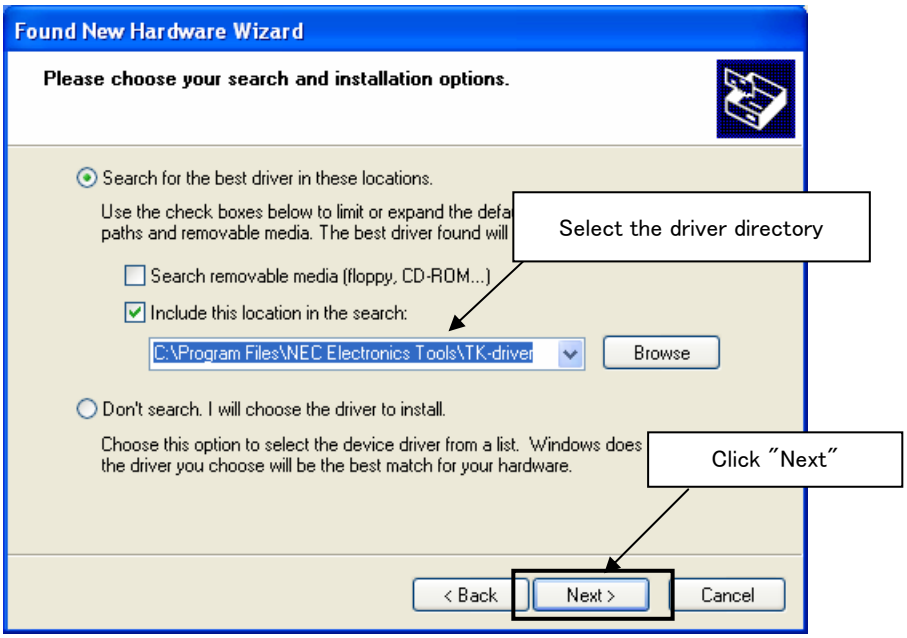
1.3.1 Installation on Windows XP

1. Once the TK-78K0/KE2C is connected with USB, the "Found New Hardware Wizard" will be started.
   Select "No, not this time" and click Next > .

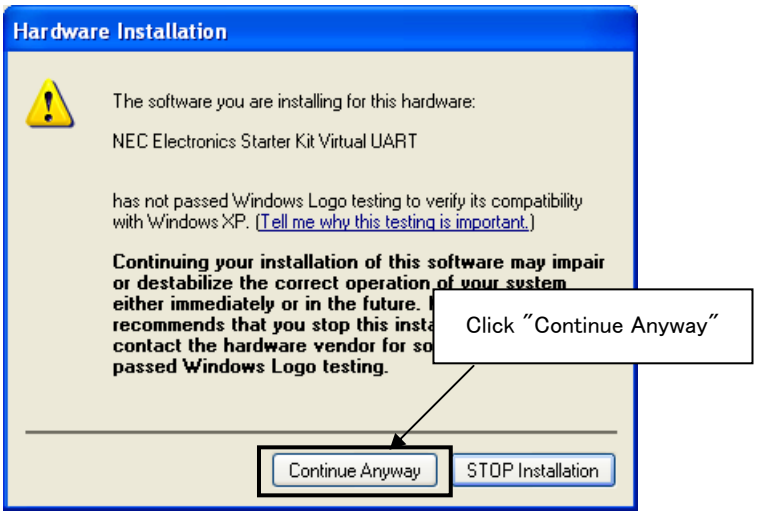
2. Select "Install from a list or specific location" and click Next > .



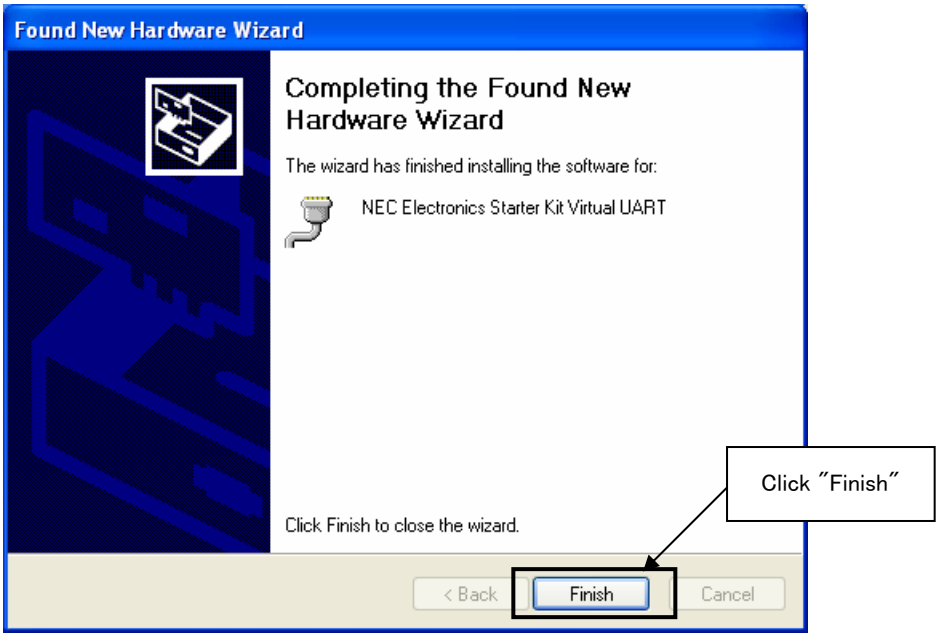
3. Select "Search for the best driver in these locations.", check "Include this location in the search:", and then click "Browse..." to select the driver directory path. The path should be "C:¥Program Files¥NEC Electronics Tools¥TK-driver" as default installation. If the installation directory is not default, then select "TK-driver" under the installation directory.
   Click [Next >] .



4. If the following dialog is opened, click [Continue Anyway] .



6. The installation of "NEC Electronics Starter Kit Virtual UART" driver is completed. Click Finish .

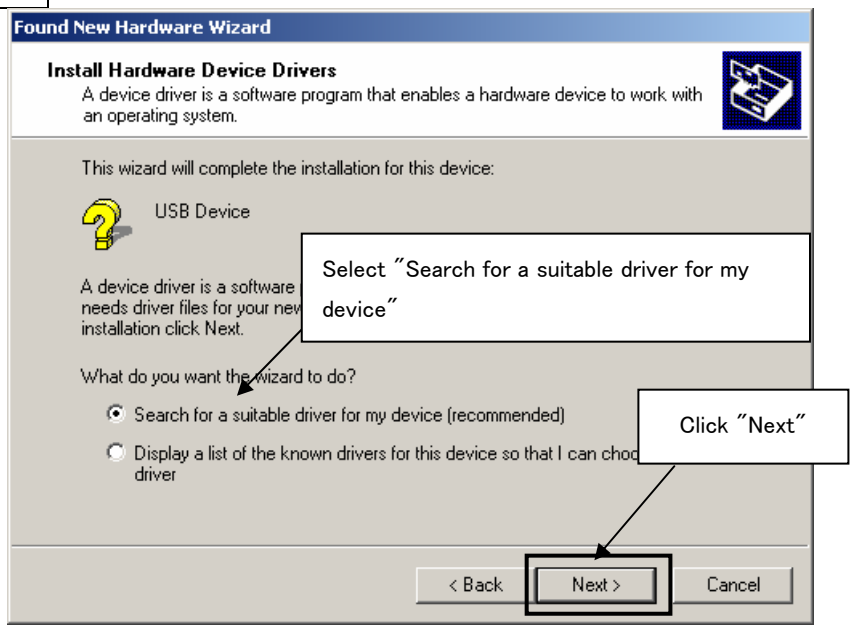


7. Go to "1.3.3 Completion of USB Driver Installation".

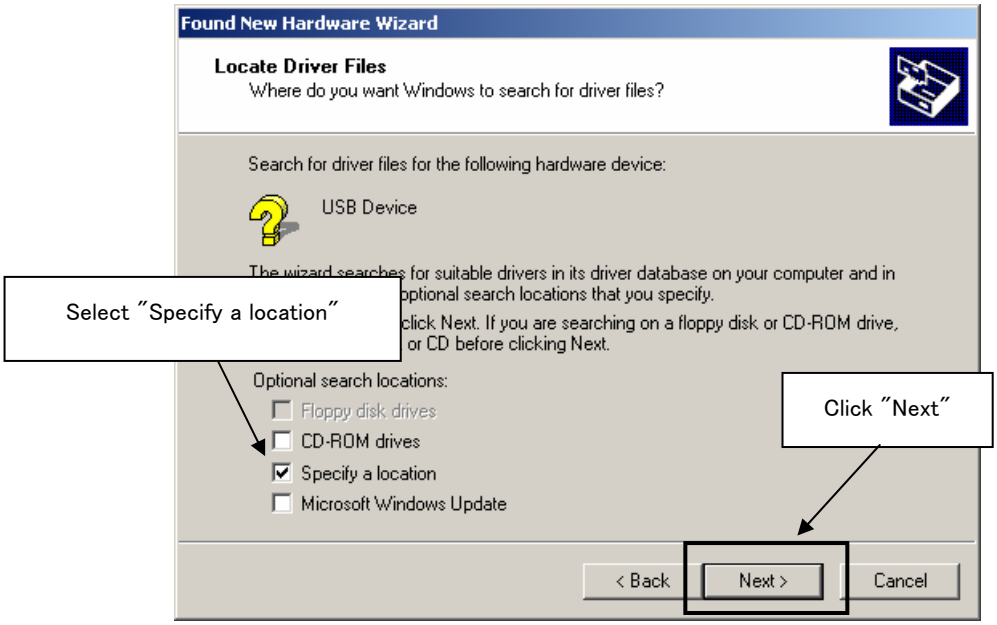
1.3.2 Installation on Windows 2000

1. Once the TK-78K0/KE2C is connected with USB, the "Found New Hardware Wizard" will be started.
   Select "No, not this time" and click [Next >] .

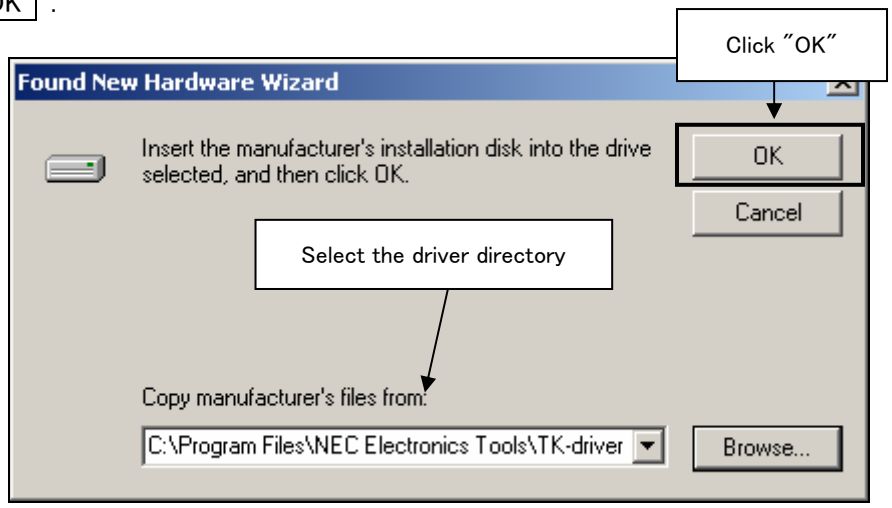
2. Select "Search for a suitable driver for my device".
   Click [Next >] .



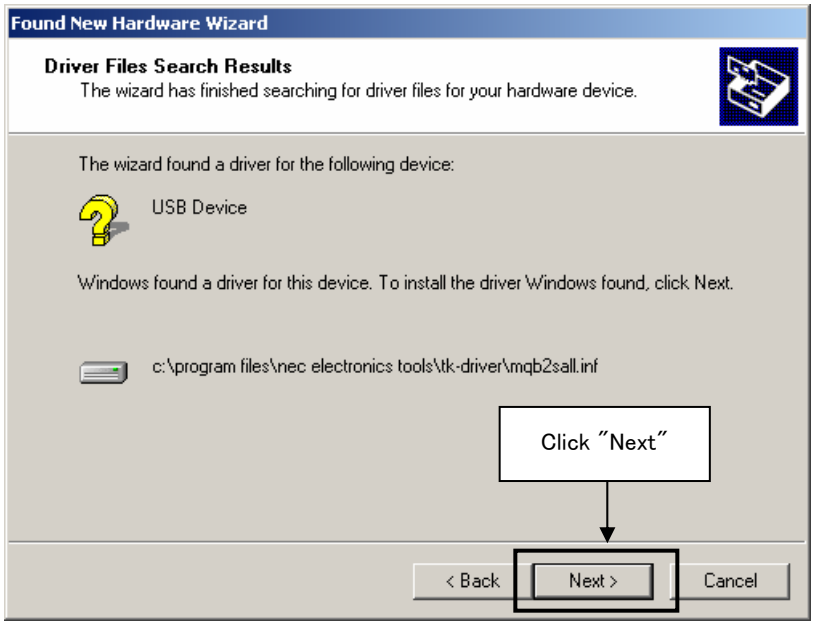
3. Select "Specify a location".
   Click [Next >] .



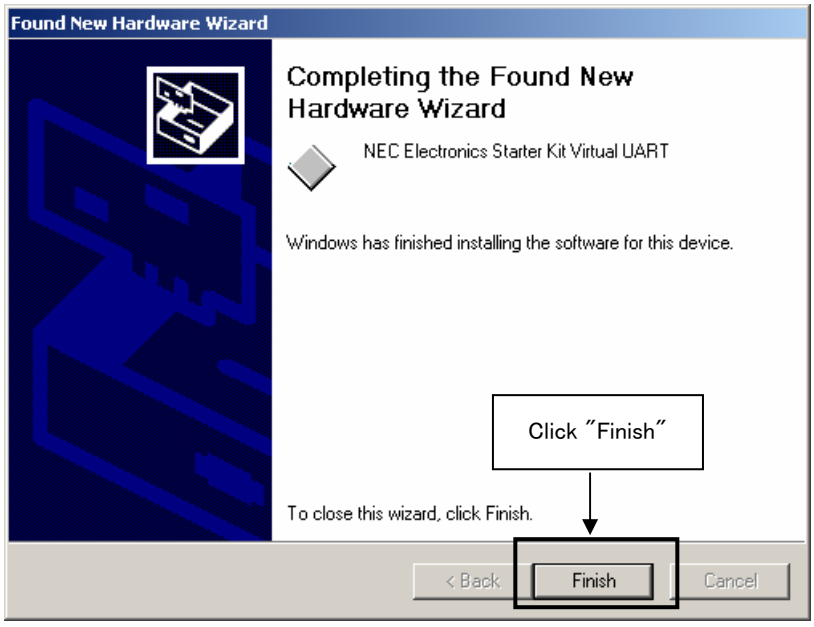
4. Select the driver directory path. The path should be "C:¥Program Files¥NEC Electronics Tools¥TK-driver" as default installation.
   If the installation directory is not default, then select "TK-driver" under the installation directory.
   Click [OK] .



5. Click Next > .



6. The installation of "NEC Electronics Starter Kit Virtual UART" driver is completed. Click Finish .



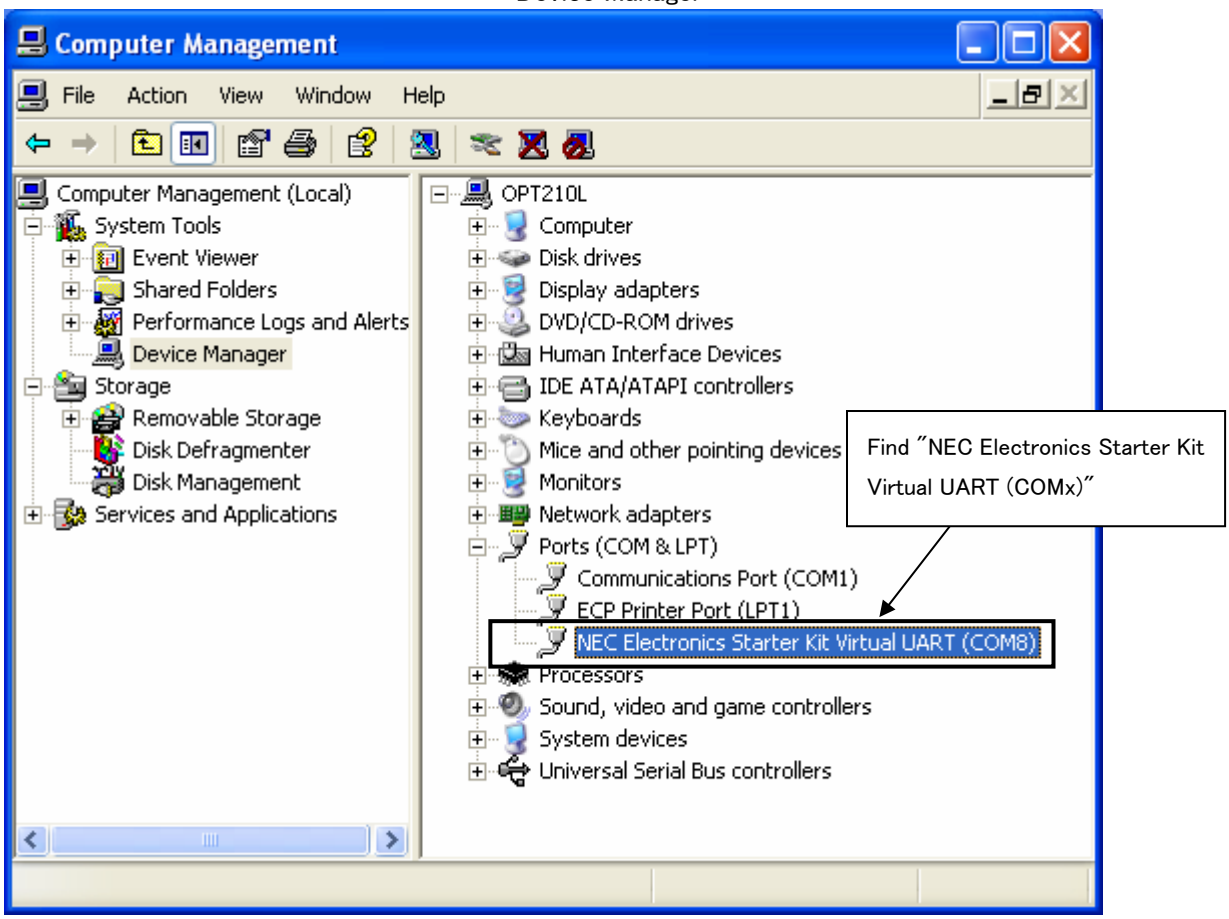
7. Go to "1.3.3 Completion of USB Driver Installation".

1.3.3 Completion of USB Driver Installation

Confirm the USB driver is installed on PC.

Start "Device Manager", and find "NEC Electronics Starter Kit Virtual UART" (without "?" mark) under the "Ports (COM & LPT)".

Device Manager



The screen above shows that the COM port number is "COM8". If ID78K0-QB is not in use, you can use this port number for connecting TK-78K0/KE2C.
When you change the USB port connection, the COM port number will be changed as well.

Select this COM port number when you use WriteEZ5.

CAUTION

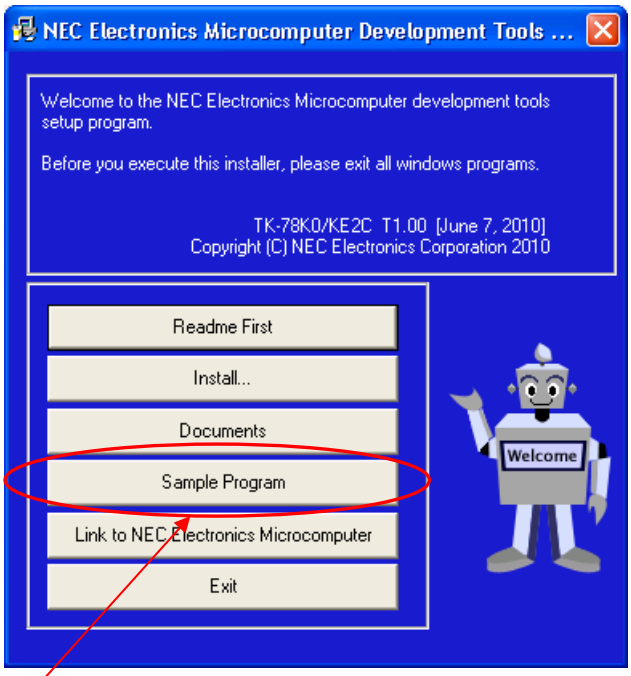
・Do not do "Hardware Modification Scan" when you communicate with the target device.

1.4 Sample Programs

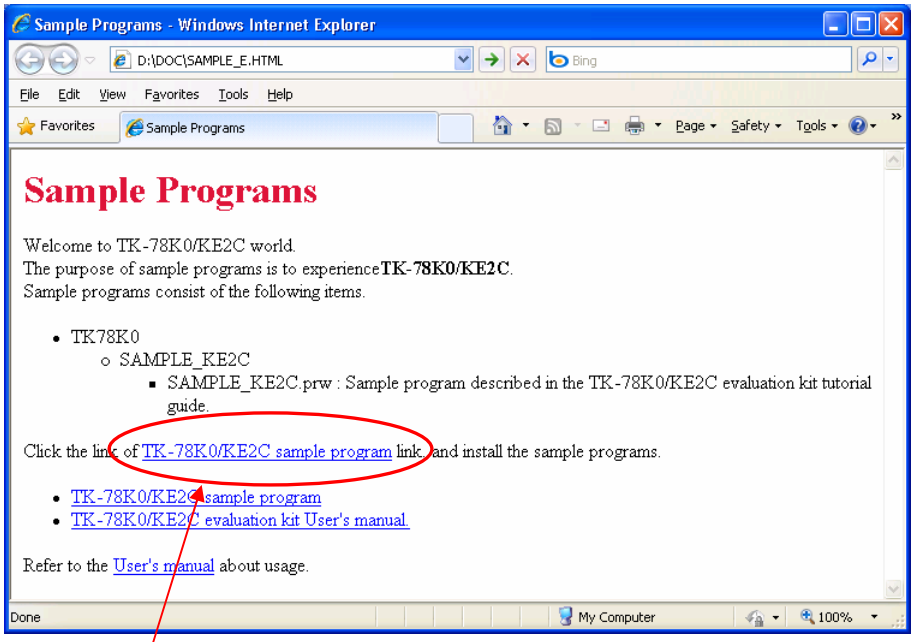
This section explains the overview and preparation of sample programs.

1.4.1 Preparation of Sample Programs

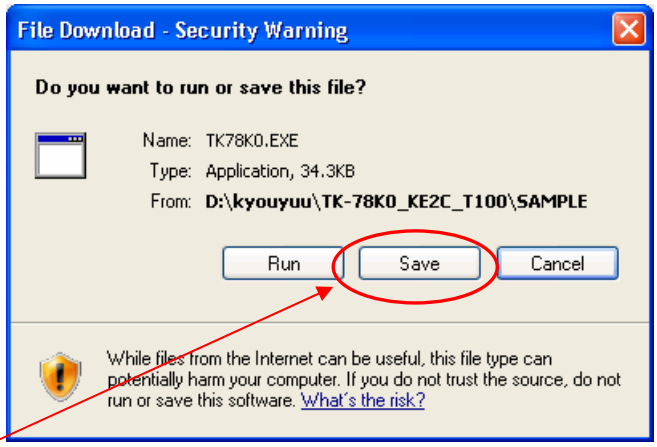
① Insert the CD-ROM disk in the CD-ROM drive of your PC. The [NEC Electronics Microprocessor Development Tools Setup] screen automatically appears.(if this screen does not appear automatically, start setup.exe from Explorer. etc.)



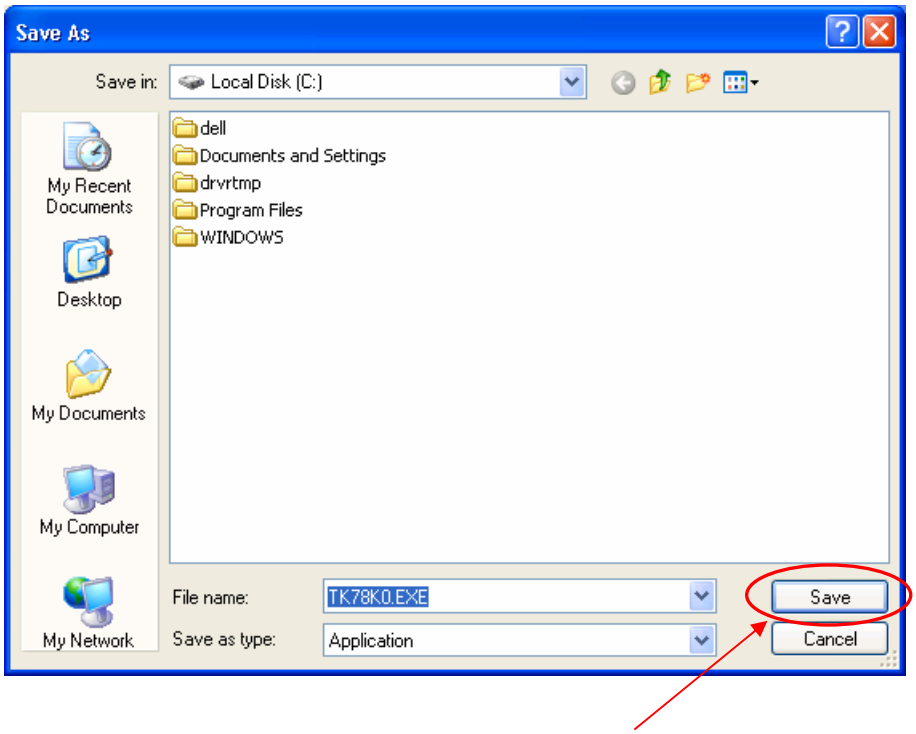
② Press the Sample Program button to start the WWW browser.



③ Click the "TK-78K0/KE2C Sample Programs" link , the following download confimation window appears.



④ Click the Save butten.

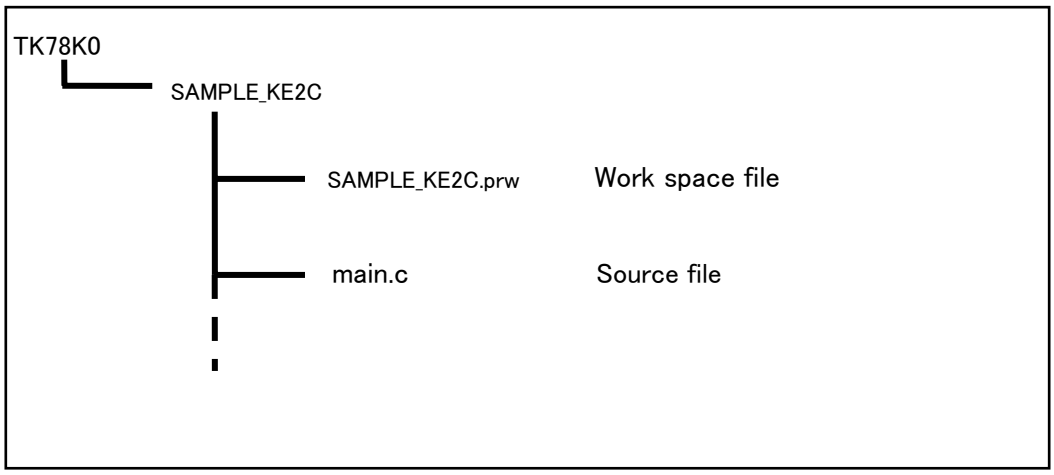


⑤ After specifying the download destination folder, click the Save button.

⑥ The self-extraction sample program set (TK78K0.exe) is copied to the specified folder.
The folder that the "TK78K0" folder is made when this file is executed, and the sample program is stored under the folder in addition is made.

1.4.2 Overview of Sample Programs

The sample programs consist of following directories.



CHAPTER 2　Experiences

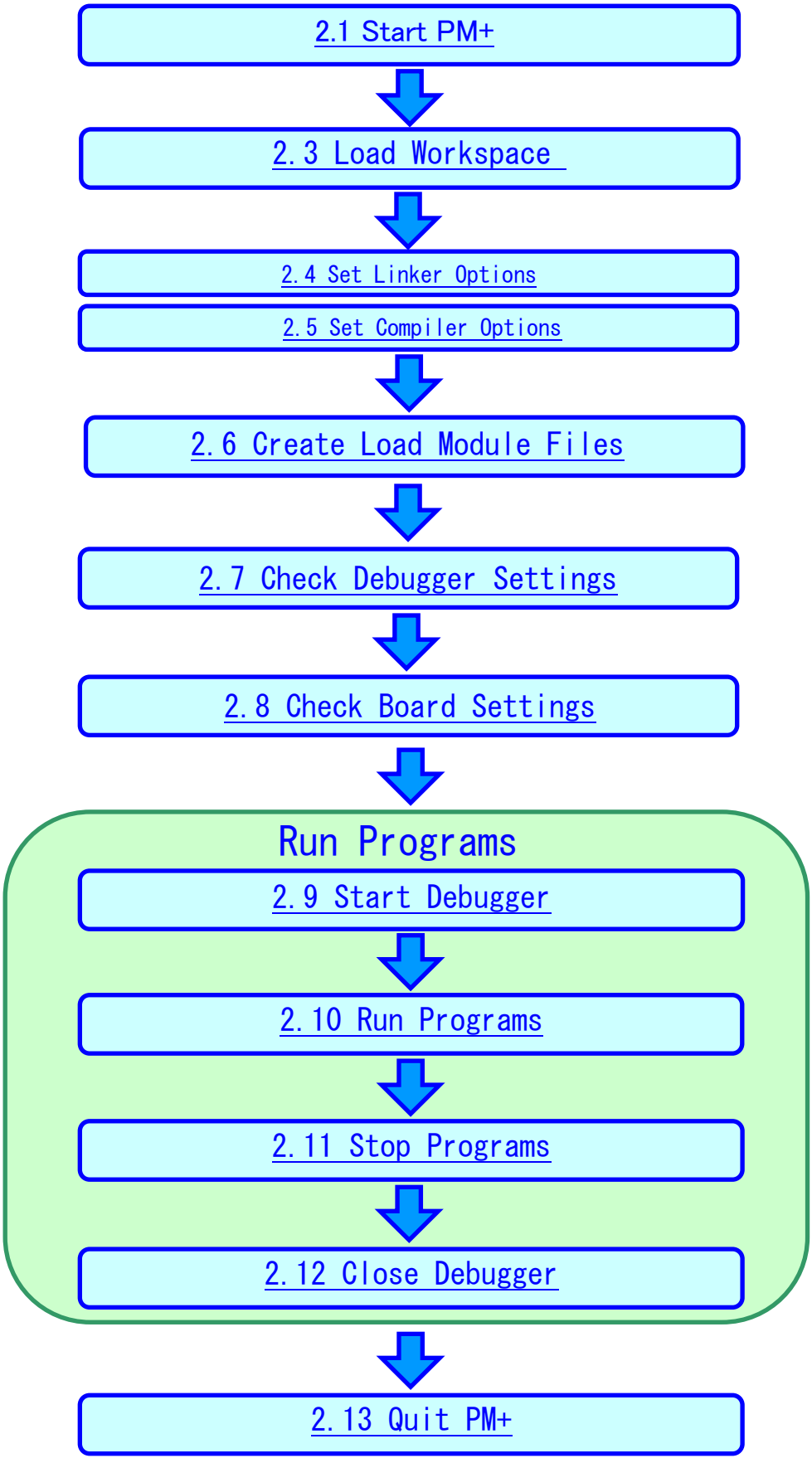
In this chapter, you will experience how to use the development tools with using the sample programs.
The development tools are :

- Integrated Development Environment (IDE), PM+
- Integrated Debugger, ID78K0-QB

You will use the programs that you prepared in "1.4 Sample Programs", as the sample programs for TK-78K0/KE2C.

You will be able to understand how to use the development tools and the concept of project files which you need for producing application programs.

The overall steps are as follows:



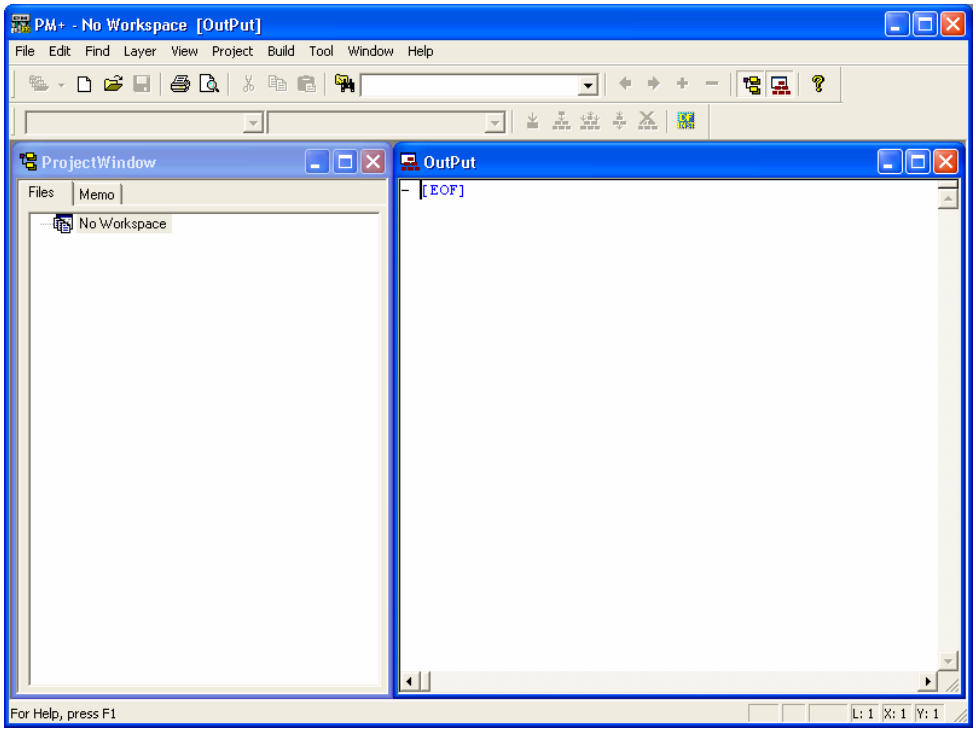
2.1 Start PM+

Let's start using the development tools.

First, start the PM+

Select "Windows Start Menu" -> "Program" -> "NEC Electronics Tools" -> "PM+ V6.30".

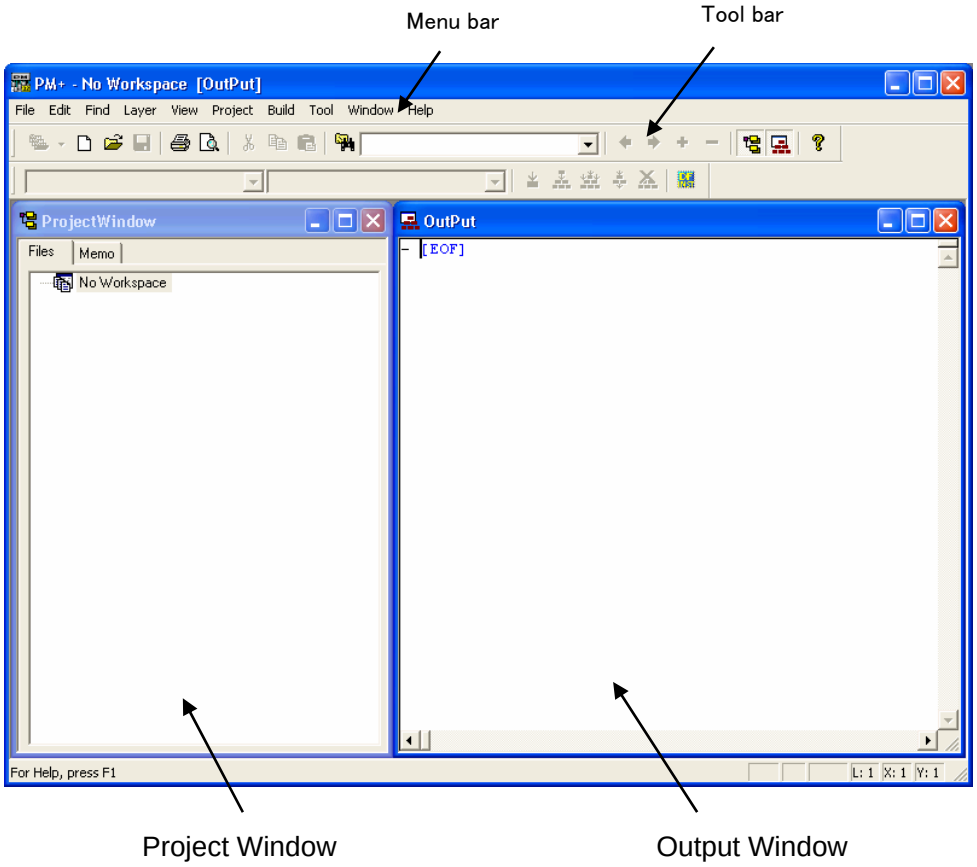
PM+ starts up



2.2 What is PM+

In PM+, application programs and environment setting are handled as a single project, and series of actions such as program creation using the editor, source management, build, and debugging are managed.

Also, one of more project files is managed together as a workspace.



Project window	A window in which project names, source files, and include file are displayed using a tree structure.
Output window	A window in which the build execution status is displayed.

For details regarding menu bars and tool bars, refer to "Help" menu in PM+.
"Help" on menu bar , then "PM+ Help"

What is a project?

A project is the unit that is managed by PM+. A project refers to an application system and environment development based on PM+.

PM+ saves project information in a "project file".

What is a project file?

A project file contains project information that includes the source files, device name, tool options for compiling, editor, and debugger information.

The file name format is "xxxxx.prj".

Project files are created in the directory you specifies when you create a new workspace.

What is a project group?

A project group is a group comprised of a number of projects in an application system.

The target device of each project must be the same within a project group.

What is a workspace?

A workspace is the unit used to manage all the projects and project group required for one application system.

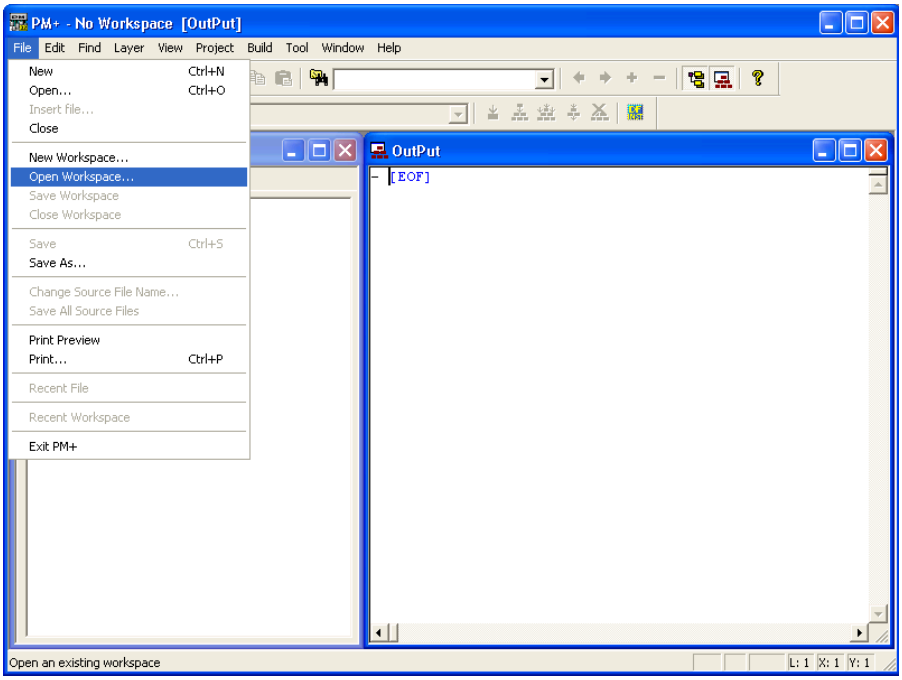
A workspace file contains one or more project files.

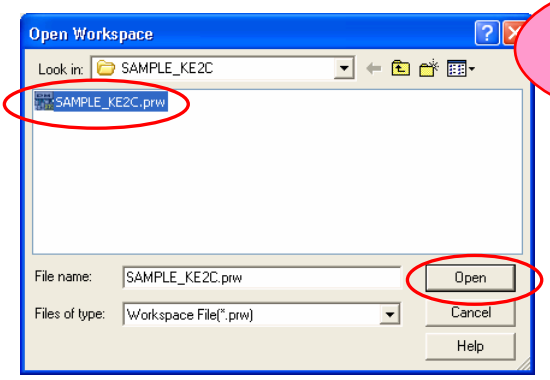
The file name format is "xxxxx.prw".

2.3 Load Workspace (project)

In this section, you will use the workspace that you created in "1.4 Sample Programs"
For creating a new workspace, refer to "Chapter 5 Other Information".
The workspace has information about the build environment for the sample programs.

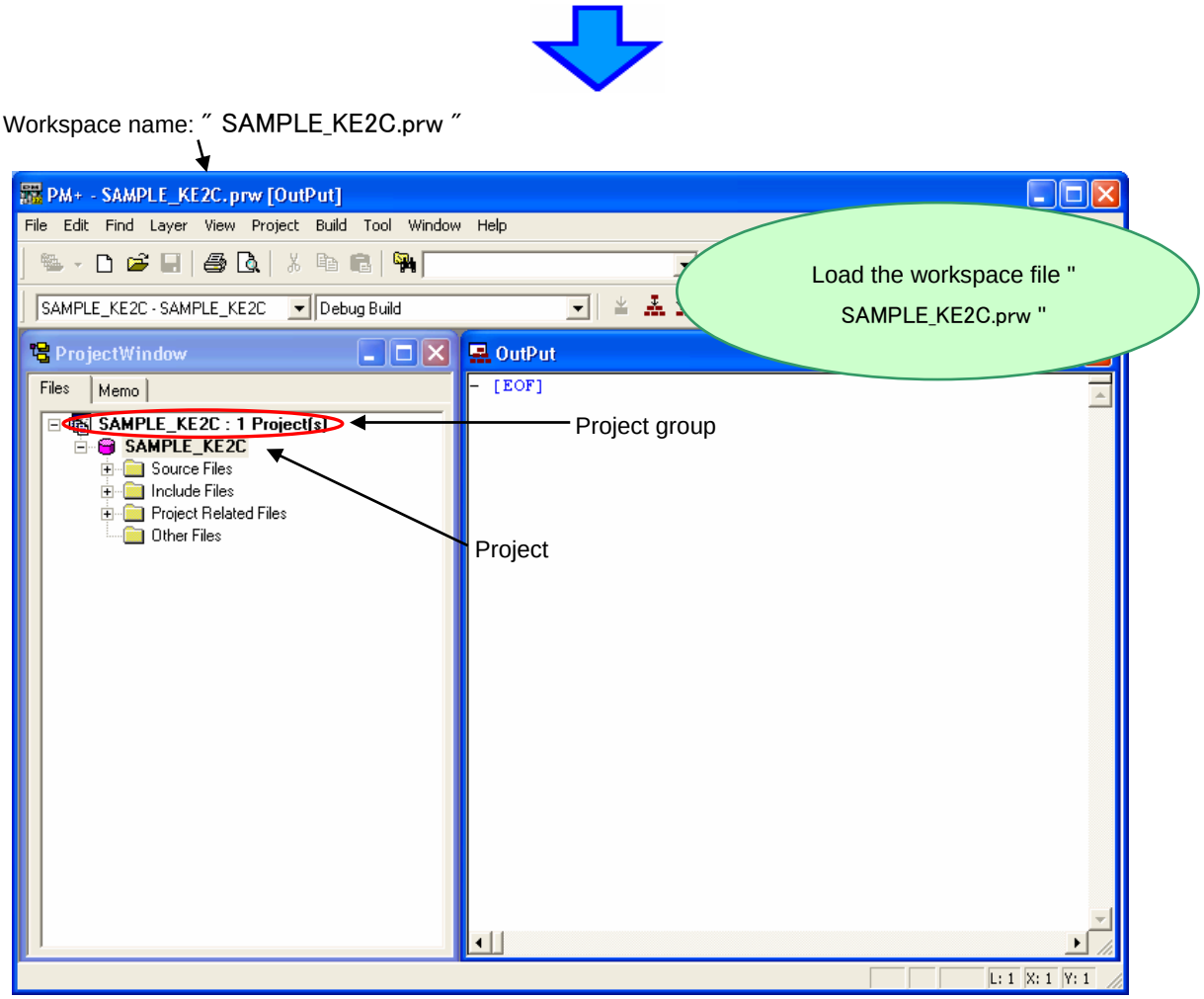
Select "File" on menu bar and "Open Workspace…".
Then, select "SAMPLE_KE2C.prw" under the directory "TK78K0¥SAMPLE_KE2C¥".





Select the directory that contains the sample programs.

Select "SAMPLE_KE2C.prw", then click Open .



The workspace file "SAMPLE_KE2C.prw" contains one project called "SAMPLE_KE2C ". You will use this project "SAMPLE_KE2C".

CAUTION:

Please ignore when you get a prompt saying "files could not be found". This may occurred when the installation directory is not a default.

2.4 Set Linker Options

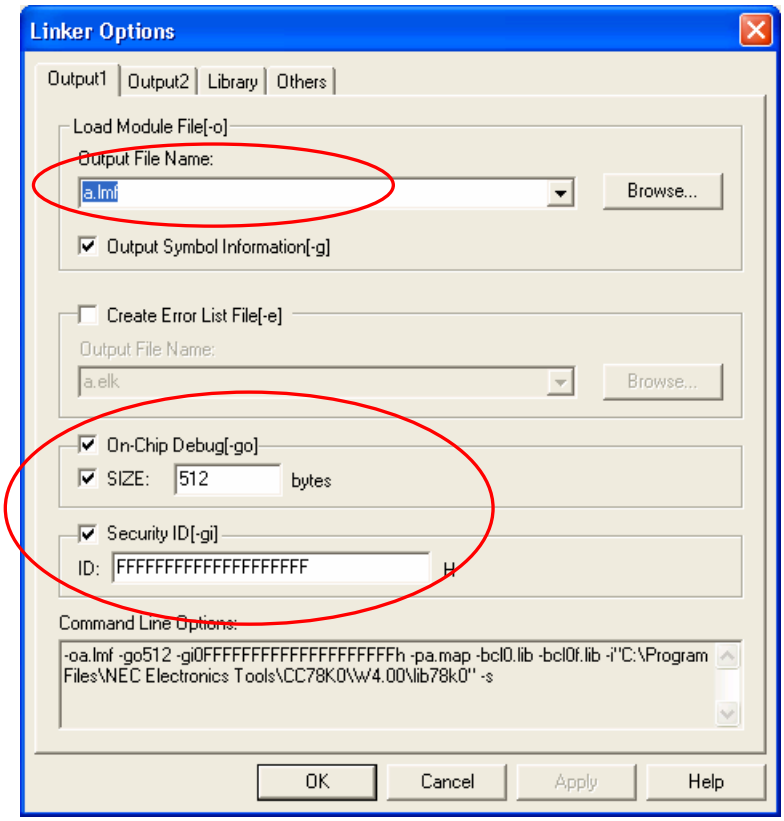
The linker options have been set by the project file. However, some option settings will be covered in this section because the linker option settings are important for debugging. Following two settings are covered specifically.

- Outputs from debugging
- On-chip debug (Disable/Enable, security ID)

Select "Tools" on menu bar, then "Linker options....".

2.4.1 "Output1" Tab

Select "Output1" tab on "Linker Options" window, and see following settings.



- Load Module File settings

  Check "Output Symbol Information".

  This enables to do source level debugging (setting break points, monitoring variables in watch window, etc).

  Also, you can specify the load module file name.

- On-Chip Debug Option Byte

  Check "On-Chip Debug Option Byte". This setting enables the on-chip debugging function of the microcontroller.

  These settings reserve the memory address area for the monitor program (the flash memory area that the debugger uses for on-chip debugging).

  <Address area that reserved by on-chip debugging>

  - 2H, 3H
  - From 8FH to the byte set in "Size"

- Security ID

  Check "Security ID", and enter the security ID which is a unique ID code (10 bytes) to authenticate when the debugger is launched.

  The security ID is stored in the flash memory (85H-8EH), and checked if it is the same as the code entered in Linker options dialog when the debugger is launched.

  The debugger will not be launched when the security ID is unmatched. By using this function, you can secure the programs from leaks.

  If you do not need to set the security, it is recommended to set the security ID "FFFFFFFFFFFFFFFFFFFF" as this is the initial code.

If you forget about the security ID (stored in the address of 85H-8EH) or if you set wrong on-chip debug option byte, you will not be able to use the debugger (ID78K0-QB).
In this case, you can erase 78K0 built-in flash memory with "WriteEZ5" to connect with ID78K0-QB.
For details, refer to "5.4 Erase microcontroller built-in flash memory".

2.5 Set Compiler Options

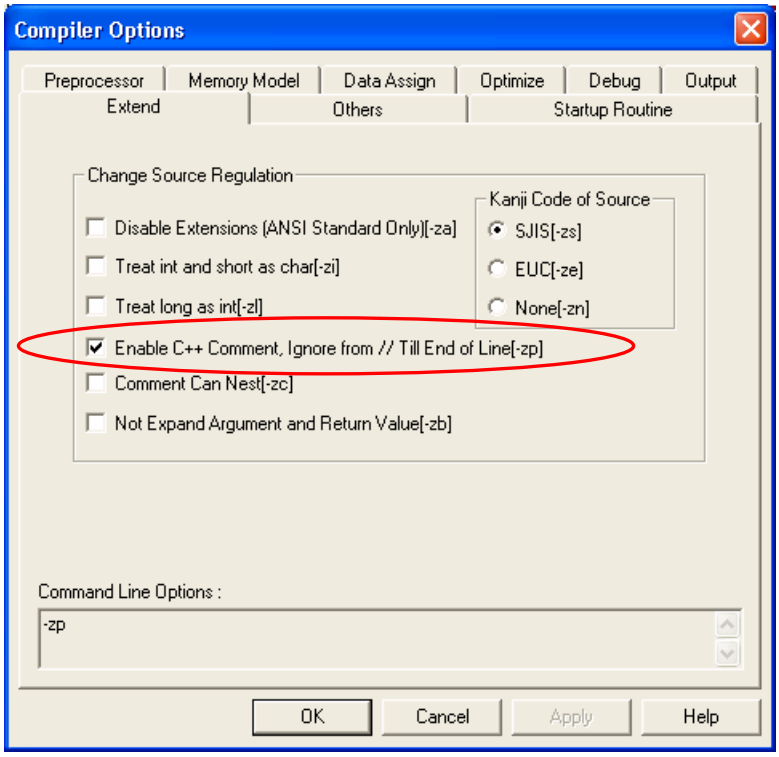
The compiler options have been set by project file. However, because some compiler options are useful, following two settings are covered specifically in this section.

- Enable C++ comments

Select "Tools" on menu bar, then "Compiler options".

2.5.1 "Extend" Tab

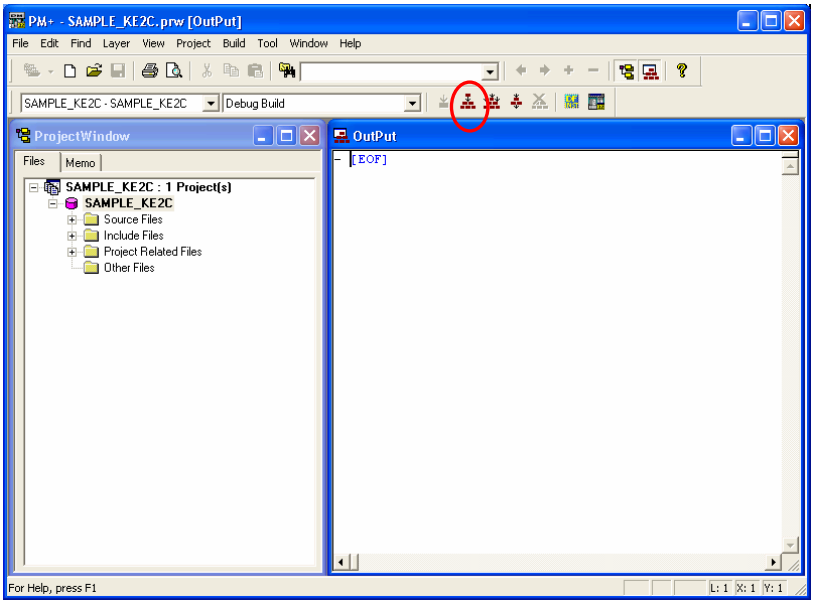
Select "Extend" tab, and check "Enable C++ Comment".
This setting allow you to use the C++ comment using "//".
It is useful feature when developing code.

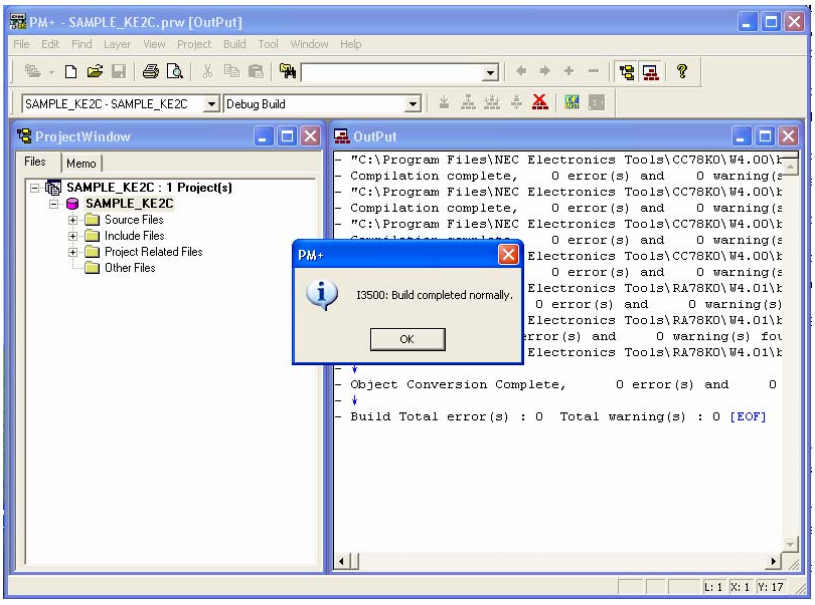


2.6 Create Load Module Files

After developing the source code, you have to create load module files by compiling, assembling, and linking. This process is called build.

Click the build button  , or select "Build" on menu bar, then "Build".





Build has been completed successfully.

What is build?

Build is a function that creates an executable file from source files in a project.

PM+ automatically performs compiling, assembling, linking, and other processing actions.

To reduce the time for the build, PM+ detects and compiles/assembles only the files that have been updated from the previous build process.

What is rebuild?

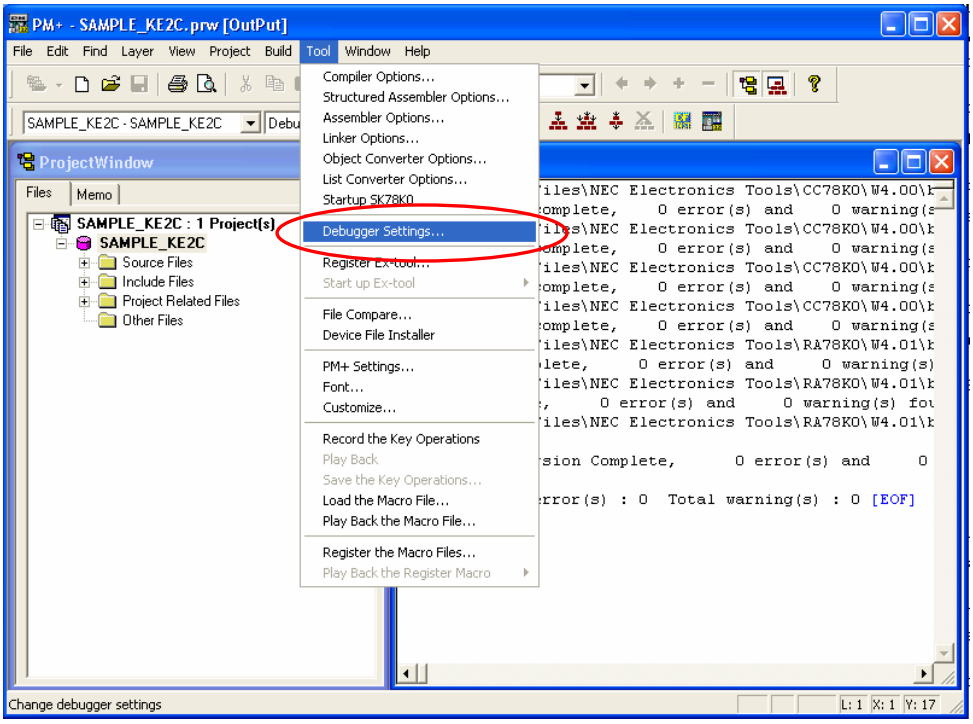
Build compiles and assembles only the source files that have been updated from the previous time, whereas rebuild compiles and assembles all the source files.

When setting, such as compiler options, have been changed, you must rebuild instead of build.

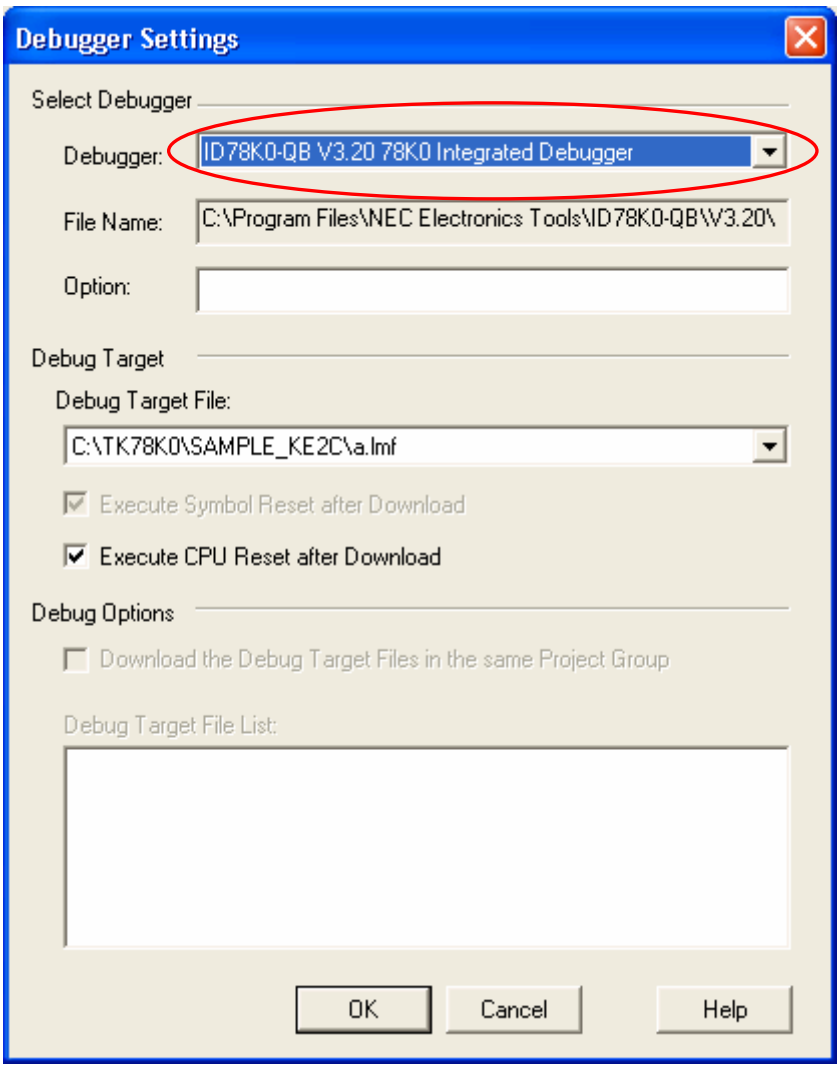
2.7 Check Debugger Settings

After the build, you should configure the debugger settings.
The debugger settings have been set by the project file as well. However, because those settings are important for debugging, some settings are covered in this section.

Select "Tools" on menu bar, then "Debugger Setting...".



Check if "ID78K0-QB V3.20 78K0 Integrated Debugger" is selected on "Debugger".

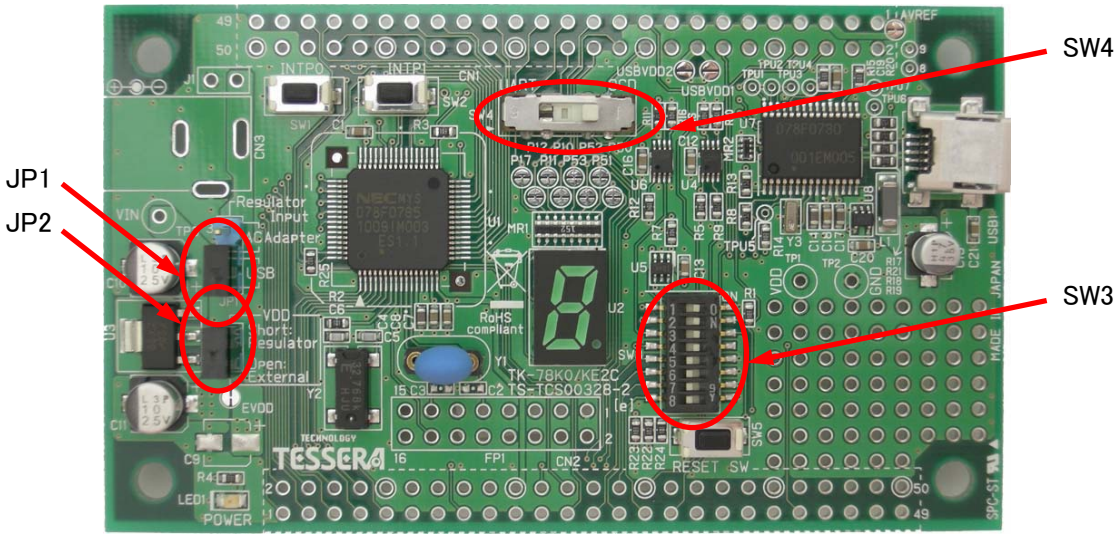


If you cannot select "ID78K0-QB V3.20 78K0 Integrated Debugger", select "Project" on menu bar, "Project settings" -> "Tool version settings" -> "Detailsetting" -> then select "ID78K0-QB".

2.8 Check Board Settings

Before connecting the PC and the TK-78K0/KE2C with USB, you should check the setting of JP1,2 ,SW3,4 on the board.
Set the JP1,2 ,SW3,4 of the TK-78K0/KE2C as follows.

JP1		1-2 Short
JP2		Short
SW4		OCD
SW3	Bit1	ON
	Bit2	ON
	Bit3	ON
	Bit4	ON
	Bit5	OFF
	Bit6	OFF
	Bit7	OFF
	Bit8	OFF



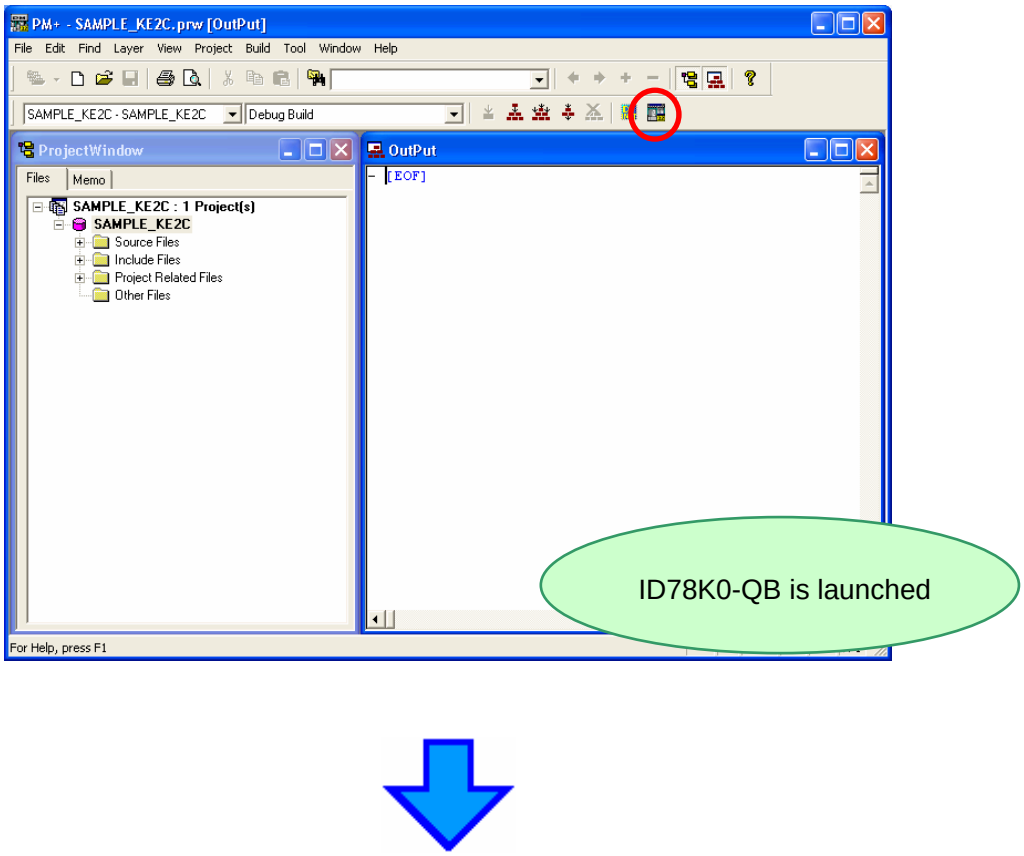
After the switch settings are completed, connect the PC to USB1 on TK-78K0/KE2C with USB cable.
If the "Found New Hardware Wizard" is started, install USB driver with referring "1.3 Installation of USB Driver".

2.9 Start Debugger (ID78K0-QB)

Click the debug button  , or select "Build" on menu bar, then "Debug".

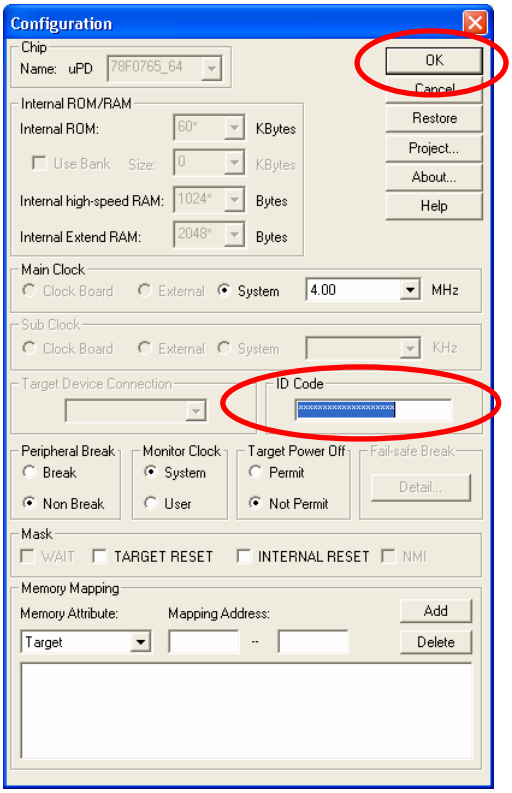
If you do not see the debug button, go to "2.7 Check Debugger Settings" for changing the settings.

The steps to start the debugger will be explained below.

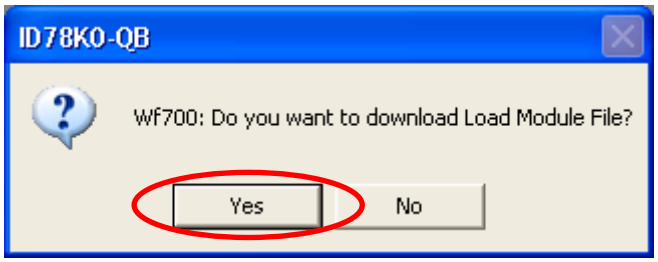


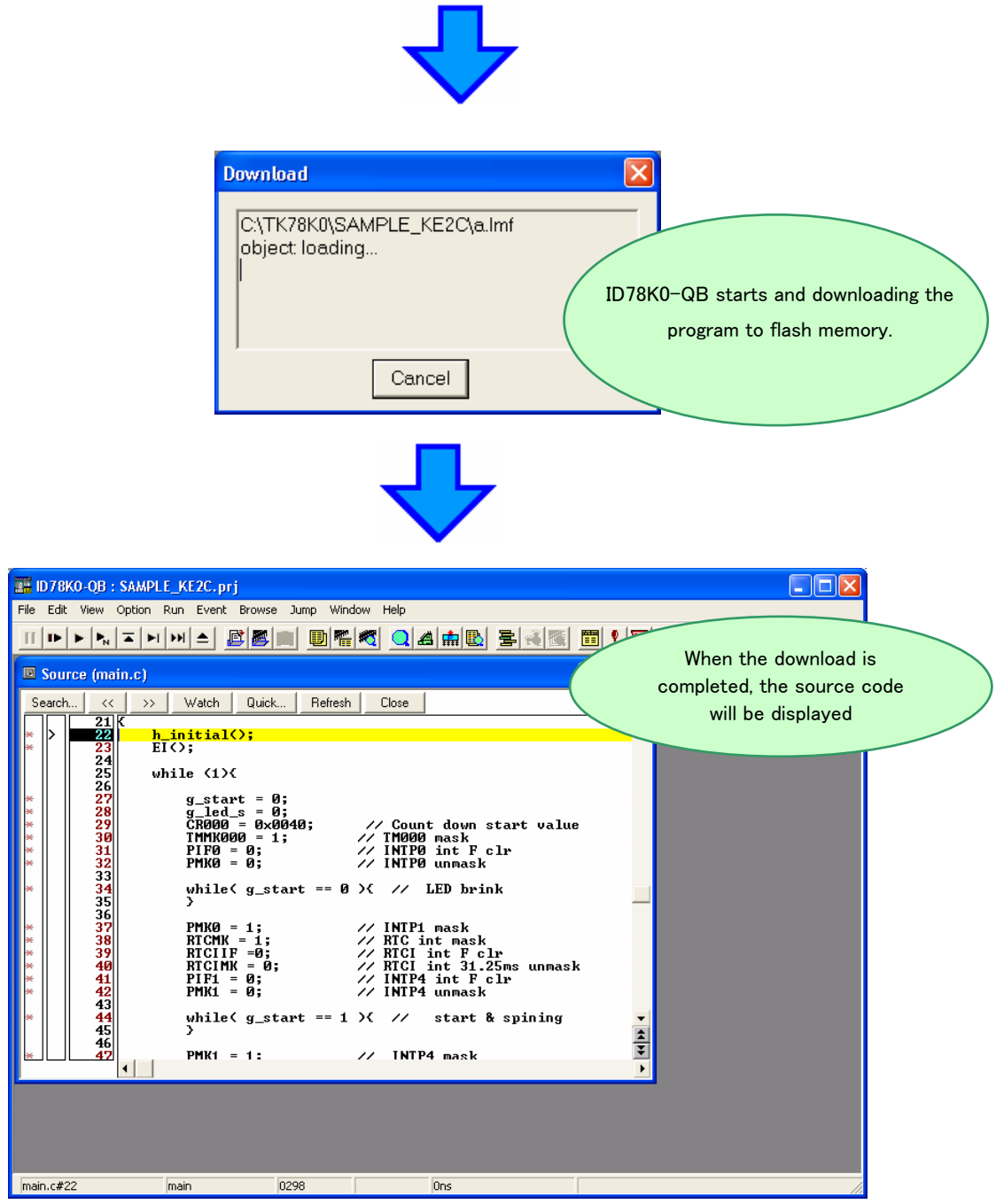
Configuration dialog opens. Set the settings shown below, and then click "OK".

ID Code Enter "FFFFFFFFFFFFFFFFFFFF" (F x20)



Click [Yes] when the confirmation dialog for downloading load module file is opened.





NOTE:

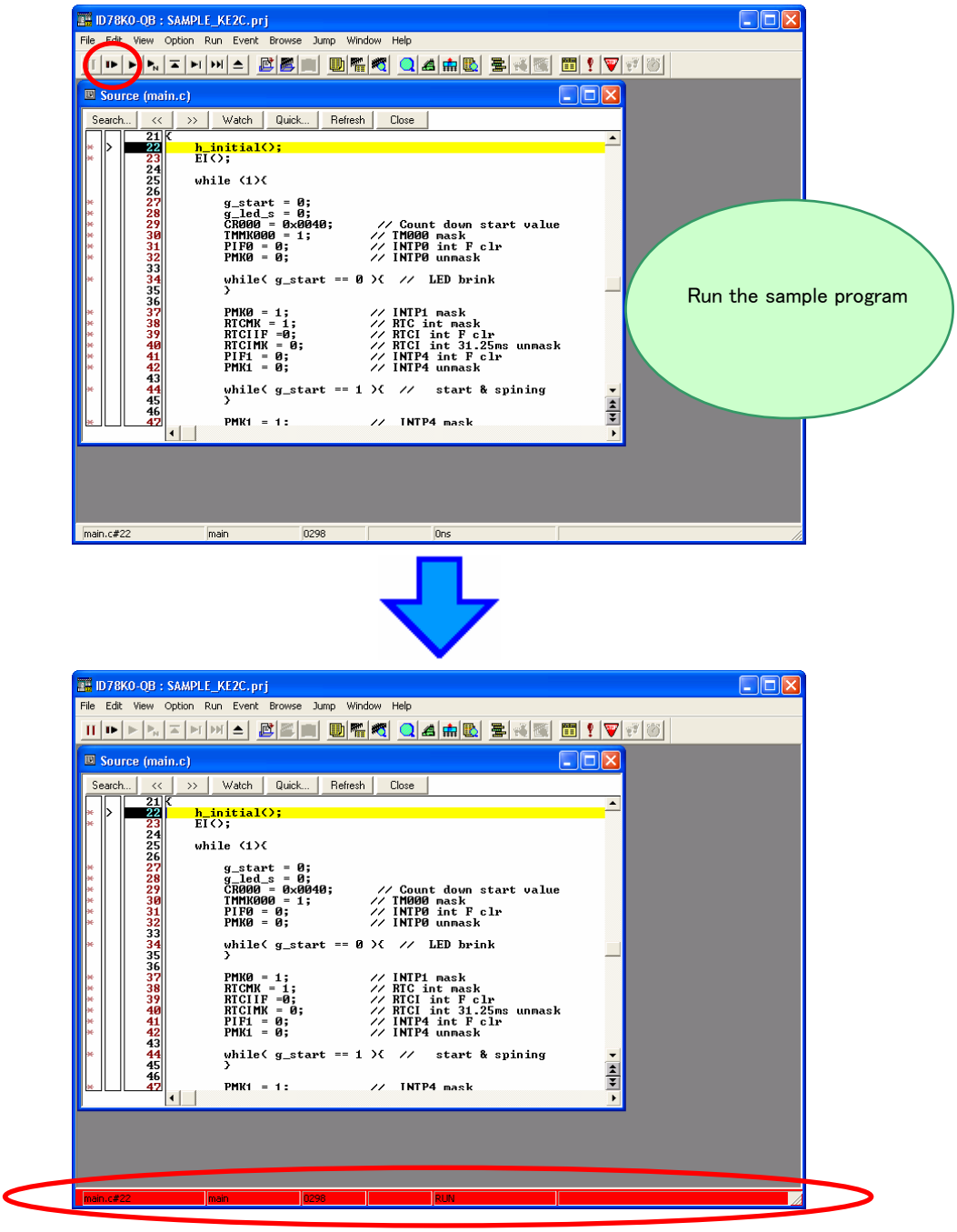
Completion of the download does not mean running the programs. Therefore, it does not make anything happened. To run the demonstration, see "2.10 Run Programs".

2.10 Run Programs

Now, you are ready to run the program.

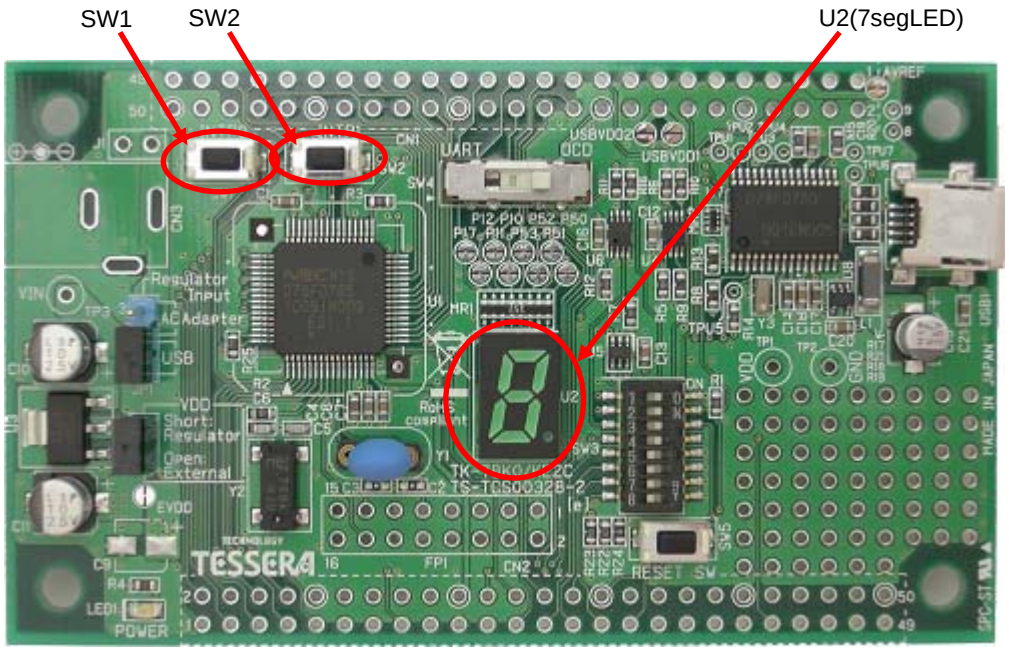
Click the restart button  , or select "Run" on menu bar, then "Restart".

The sample program runs.



When programs are running, the status bar will be red.

Check that the 7segment LED on TK-78K0/KE2C is lighted.



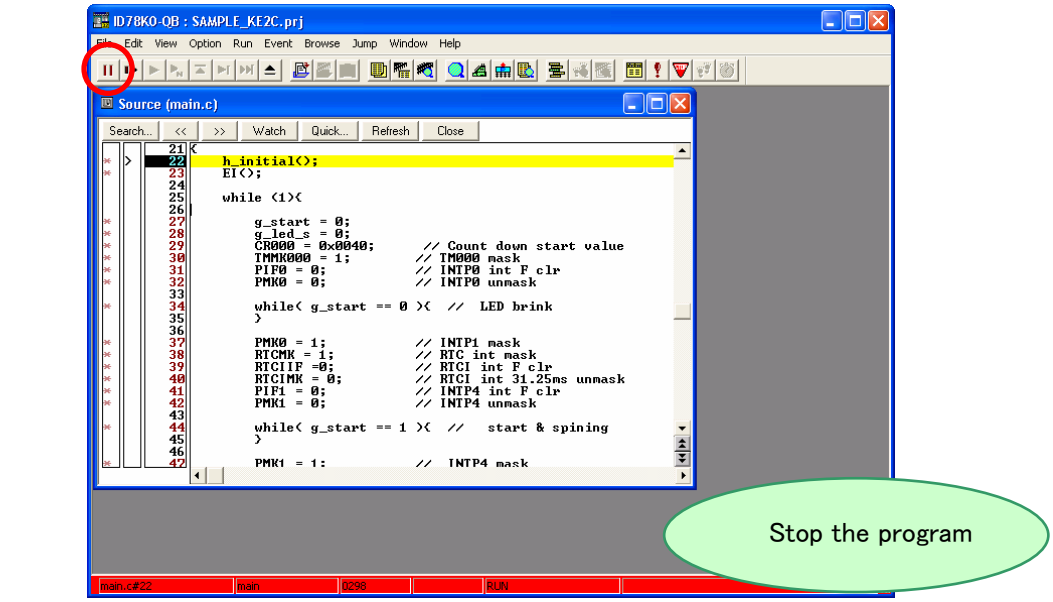
When you press SW1, the segment of the7segLED rotates.
When you press SW2, number is displayed on the 7segLED after a while.

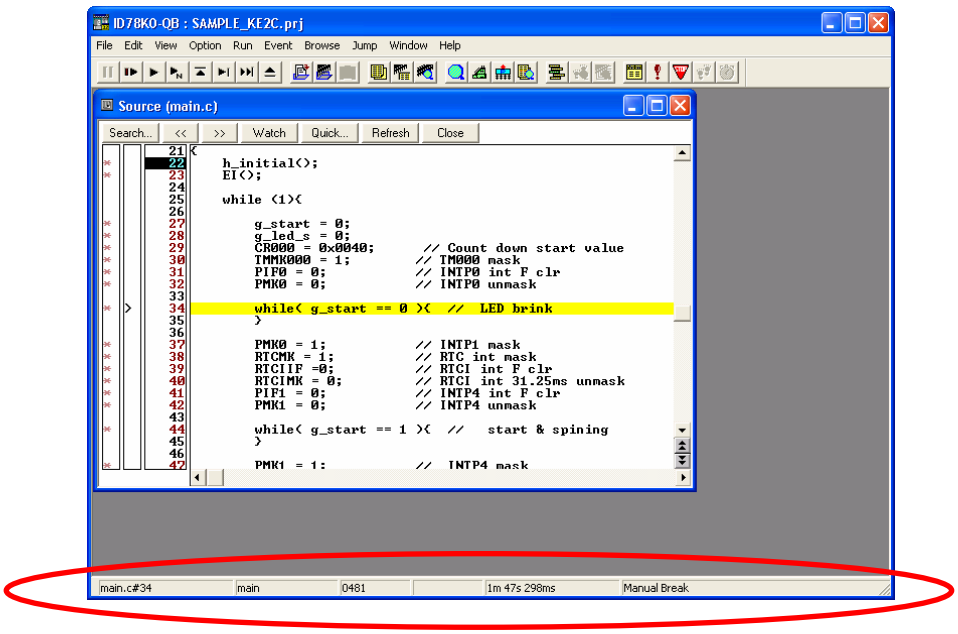
You could confirm the sample program is working.

2.11 Stop Programs

Now, you are going to stop the program.

Click the stop button , or select "Run" on menu bar, then "Stop".

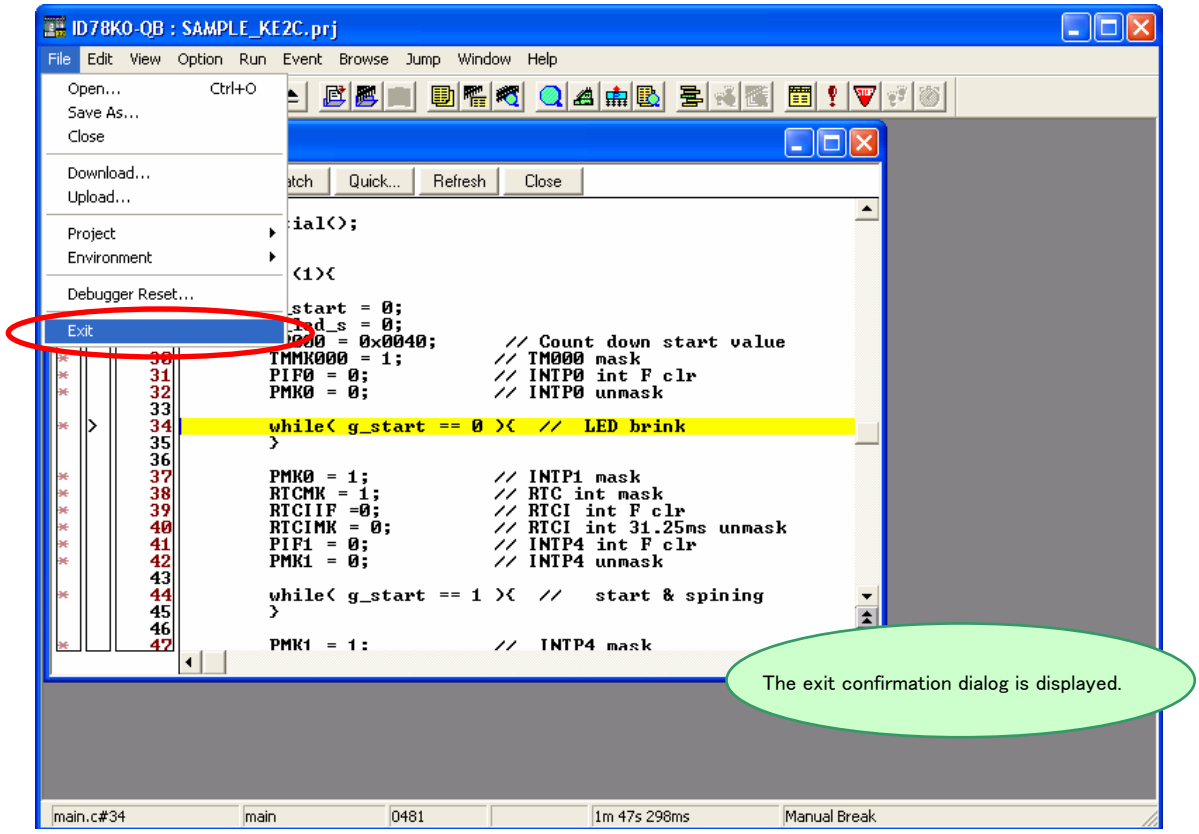


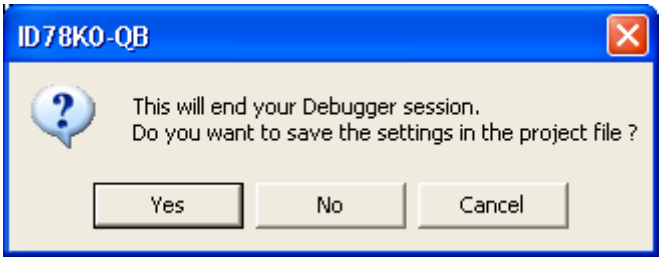


When the program stops, the status bar changes back to the original color.

2.12 Close Debugger (ID78K0-QB)

Select "File" on menu bar, then "Exit".



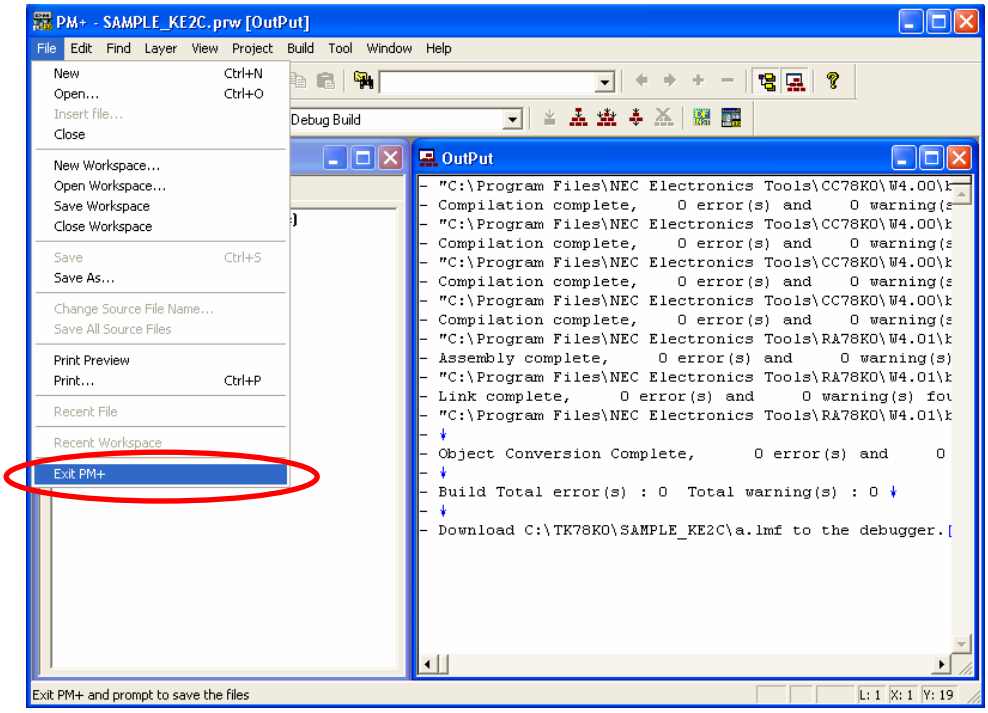


If you click [Yes] , it saves the settings in the project file, and then closes the ID78K0-QB. It is recommended to save the settings as it saves the window you used, window size, layout, etc.

If you click [No] , it does not save the settings and closes the ID78K0-QB.

2.13 Quit PM+

Select "File" on menu bar, then "Exit PM+".



PM+ is closed.

The experiences section ends now.
You can find more information how to use the development tool and information about other useful features in "Chapter 5 Other Information".

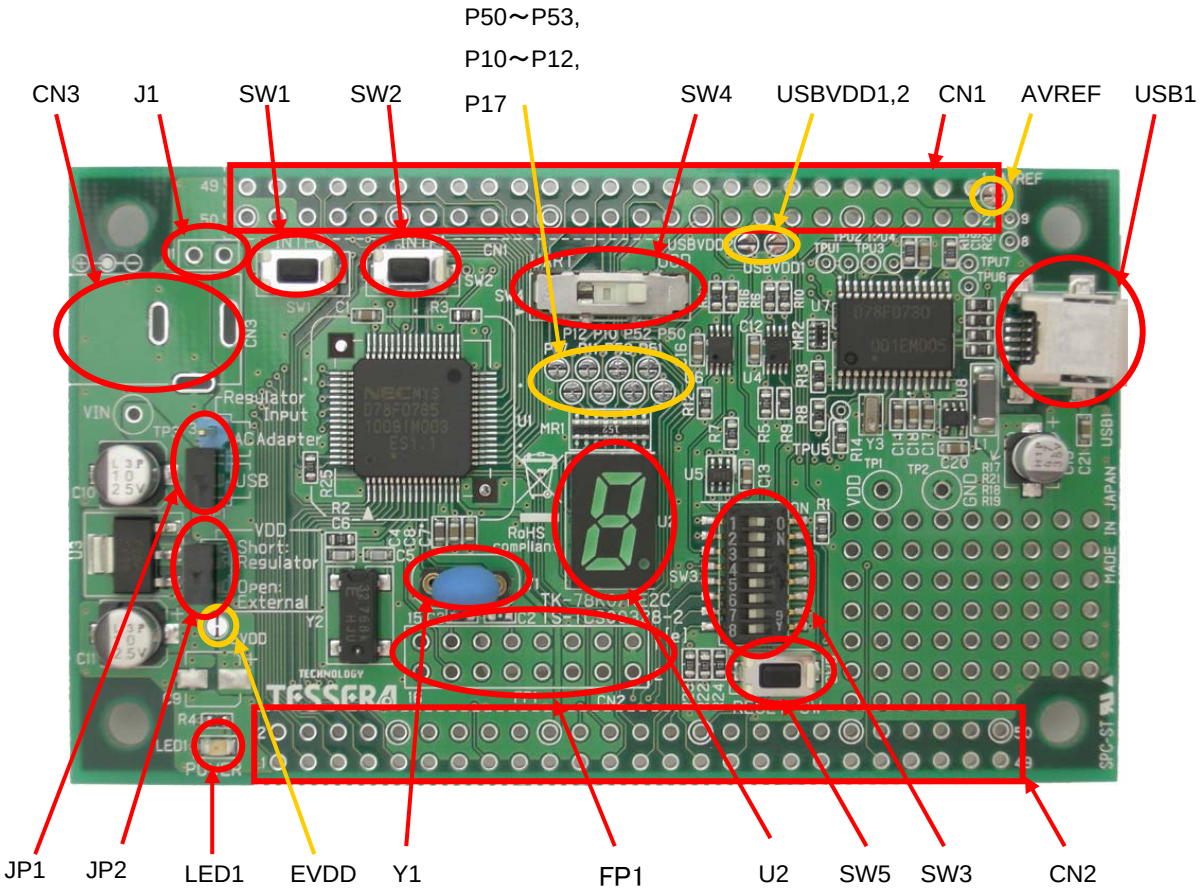
CHAPTER 3 Hardware Specifications

In this chapter, the hardware of TK-78K0/KE2C will be explained.

Microcontroller	μPD78F0765 ※78K0/KE2C
Clock	External main system clock: 20MHz Subsystem clock: 32.768KHz Internal high-speed oscillation : 8MHz
Interface	Mini B USB (USB1) Expansion connector 50Pin socket x2 (CN1,2) pad only MINICUBE2 connector (FP1) pad only
Power supply voltage	5V (USB)
Input/output for operation check use	・7segLED(U2) ・Push switch (SW1,SW2) ・Dip switch (SW3 5,6,7,8bit) ・Reset switch (SW5)
Other hardware	・Mode switch(SW1 1,2,3,4bit) ・Power LED(LED1)

* The name with bracket is the name printed on the board.

3.1 Layout of hardware functions



3.2 Hardware Functions

3.2.1 SW3, SW4

The bit 1-4 on SW3 are for mode settings, and bit 5-8 are DIP switches for general purpose ports that connected to P00, P02～P04 pins in microcontroller.

● For the use of ID78K0R-QB, use following settings.

SW3

Bit 1	ON/OFF *1
Bit 2	ON
Bit 3	ON
Bit 4	ON

SW4 [OCD] side

***1 OFF :The microcontroller stays being reset until ID78K0R-QB is started.**
ON :The microcontroller runs the programs stored in the flash memory as soon as it gets power supply.

***2 If you use ID78K0-QB, it uses P31 and P32 for communicating with host machine. Therefore, you cannot use P31 and P32.**

● To run the programs stored in built-in flash memory without using ID78K0-QB, use following settings and re-supply USB power.
You can use P13(TXD6),P14(RXD6) for multipurpose serial port via μPD78F0730.
You can use the On-chip debug function with connecting a MINICUBE2. (Connecter was not mounted)

SW3

Bit 1	OFF
Bit 2	OFF
Bit 3	OFF
Bit 4	OFF

SW4 [UART] side *2

*2 Select [center] position when P13,P14 are used on the expansion connector.

● Please change to the following settings when writing it in the flash memory with built-in CPU by using Write EZ5.

SW3

Bit 1	ON
Bit 2	ON
Bit 3	ON
Bit 4	OFF

SW4 UART side

● Bit5-8 of SW3 are connected to following CPU pins
It becomes "Low" if the switch is pushed, and it becomes "Open" if it separates. Please turn on pull-up resistor (PU0) with built-in CPU when using it.
(For details about settings for microcontroller built-in pull-up option resistor, refer to "78K0/Kx2C User's Manual <U19822>".)

SW3

Bit 5	P00
Bit 6	P02
Bit 7	P03
Bit 8	P04

3.2.2 SW1,SW2

SW1,SW2 are the tact switch.

When you push the switch, it becomes "Low", and when you release the switch, it becomes "Open". Before using this, you need to set the microcontroller built-in pull-up option resistor (PU3,PU12) to ON. (For details about settings for microcontroller built-in pull-up option resistor, refer to "78K0/Kx2C User's Manual <U19822>".)

SW1,SW2

	Port	Port function	Note
SW1	P120	P120/INTP0/EXLVI	
SW2	P30	P30/INTP1	

3.2.3 SW5 (RESET SW)

SW5 is the reset switch. You can reset the microcontroller by pressing this switch.

3.2.4 JP1

JP1 is the jumper short pin to select power supply.

1-2 Short	Use USB power supply from the USB1 connector
2-3 Short	Use CN3, J1 connector power supply

3.2.5 JP2

JP2 is the jumper short pin for CPU power supply.

Short	Use regulator output to the internal power
open	Separate regulator output to the internal power. You can use other power supply to the internal power.

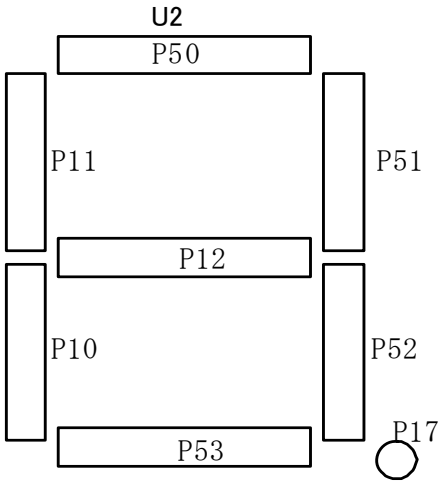
3.2.6 LED1 (POWER)

LED1 is the POWER LED. It is lighted when it gets power supply.

3.2.7 U2(7seg LED)

U2 is general purpose 7SegLED. LED is lighted when each port outputs "Low".

7SegLED segment connection



The figure of 0-9 can be displayed by writing the following values in P1, P5 register.

Example of displayed figure and set data.

Figure	P5 Data	P1 Data	Figure	P5 Data	P1 Data
0	0xF0	0xFC	5	0xF2	0xF9
1	0xF9	0xFF	6	0xF3	0xF8
2	0xF4	0xFA	7	0xF8	0xFF
3	0xF0	0xFB	8	0xF0	0xF8
4	0xF9	0xF9	9	0xF8	0xF9

3.2.8 FP1

The connecter for MINICUBE2.
The connecter was not installed. (HONDA TSUSHIN KOGYO CO., LTD.　FFC-16BMEP1)

3.2.9 CN1, CN2

The connecter for CN1,CN2 expansion connection.
The connecter was not installed. (HIROSE ELECTRIC CO., LTD. HIF-3H-50DA-2.54DSA)

3.3 Solder-short pad label

With using the solder-short pad to cut the circuit, users can customize the circuit.
The solder-short pad looks like the picture below.
To open, use cutter to cut the dent part. To short, put solder on the pad.

Solder-short pad
(Open)

Solder-short pad
(Short)

Solder-short pad name	Before Shipment	Connection	
P50～P53, P10～P12, P17	Short	Short	The ports are connected to 7seg LED via 1.5KΩ.
		Open	You can use port 5 for multipurpose I/O.
AVREF	Short	Short	VDD = AVREF
		Open	AVREF is separated from AVREF.
EVDD	Short	Short	VDD = EVDD
		Open	EVDD is separated from EVDD.
USBVDD1,USBVDD2	Open	Short	Supply USB 5V to CN1 connecter.
		Open	You can use CN3 and J1.

Note: All ground signals are each connected.

3.4 Operation by external power supply

The power supply of the AC adaptor connected with CN3 is connected only with the power supply terminal of the connector of the board in the surrounding (10, 12, and 16pin of CN1).
The AC adaptor can be made a power supply by the connection of the AC adaptor of 5V to CN3 and connect JP2 jumper short, and JP1 2-3 short.
Moreover, it is also possible to connect the stabilizing supply etc. in the lead line instead of the AC adaptor because CN3 is connected with J1 of a through hall.

- Acceptable jack (CN3) : HEC0470-01-630 by Hosiden Corp(not mounted)
- Acceptable plug : 2.1mm DC jack(center plus)
- Current capacity : 100mA or more

CHAPTER 4 Troubleshooting

This chapter describes how to solve troubles you may face.

4.1 If you cannot find USB driver when you connect PC to the kit

Check Point 1

If you use USB hub, do not use it. (USB hub is not supported)

Check Point 2

Check if you installed "Starter Kit USB Driver" in "1.2 Installation of Development Tools". If not, install the driver.

Check Point 3

Check if the settings of SW3,4 on the kit are correct with referring to "1.3 Installation of USB Driver".

Check Point 4

If above 3 check points are confirmed, disconnect the USB cable from PC and re-connect again. It should show the "Found New Hardware Wizard" wizard. Operate the installation with referring to "1.3 Installation of USB Driver". After the installation, make sure you go through "1.3.3 Completion of USB Driver Installation" to confirm the USB driver installation.

4.2 Error when you start the debugger

There could be several reasons to make errors happen.
The solving processes differ depending on errors. Please check the error message first.
The solving processes for each error are as follows.

4.2.1 "Can not communicate with Emulator..." (F0100 or A0109)

Check Point 1

If you use USB hub, do not use it. (USB hub is not supported)

Check Point 2

Check if the settings of 3,4 on the kit are correct with referring "1.3 Installation of USB Driver".

Check Point 3

Confirm the USB driver installation with referring to "1.3.3 Completion of USB Driver Installation".

Check Point 4

If above 3 check points are confirmed, close the debugger and disconnect the USB cable from PC. Re-connect USB cable properly to both the PC and the kit, and then re-start the debugger.

4.2.2 "Incorrect ID Code." (Ff603)

This error occurs when the security ID stored on microcontroller built-in flash memory is different from the ID code you entered at the start of debugger.

Security ID entry area at the start of debugger

Check Point 1

Enter correct security ID and click OK on the configuration window.

Check Point 2

If you forgot the security ID, you have to erase the microcontroller built-in flash memory. Before erasing, check if you actually set the security ID with referring to "2.4 Set Linker Options". Also remember the code you set for the security ID.
After this, erase the flash memory with referring to "5.4 Erase microcontroller built-in flash memory".

4.2.3 "The on-chip debug function had been disabled in the device." (F0c79)

This error occurs when the value at address C3H (On-chip debug option byte) in microcontroller built-in flash memory is incorrect. You need to erase the flash memory.

Check Point 1

Check if you actually set the correct on-chip debug option byte with referring to "2.4 Set Linker Options". If it is not correct, then set correctly.

Check Point 2

Erase the flash memory with referring to "5.4 Erase microcontroller built-in flash memory".

4.2.4 "Disabling the on-chip debug function is prohibited." (F0c33)

Basically, this error occurs when you start (download) the debugger without doing the settings described at "2.4 Set Linker Options". Do the same checking processes as"4.2.3 The on-chip debug function had been disabled in the device. (F0c79)".

CHAPTER 5 Other Information

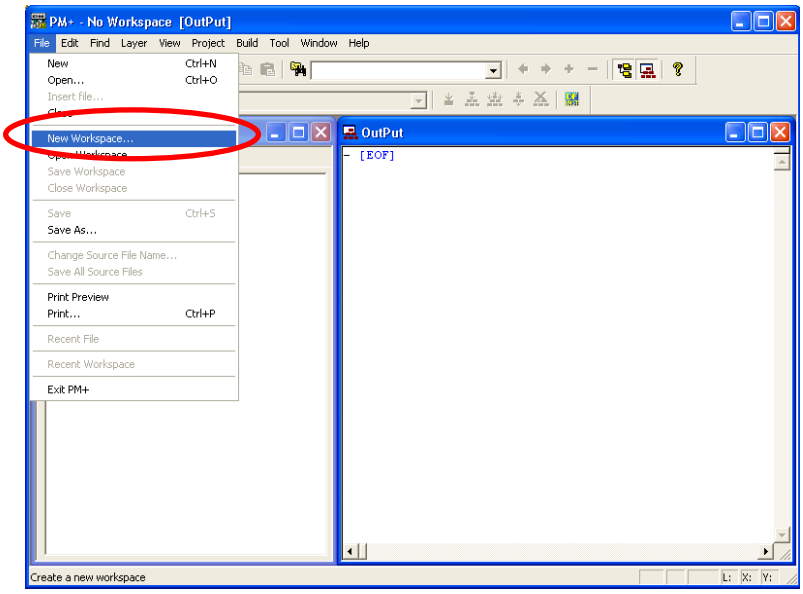
This chapter explains some useful operation techniques of development tools and circuit diagram of the kit for developing of user programs.

5.1 Create a new workspace

Now, create a new workspace and project.

PM+ allows you to create a new workspace with following "New WorkSpace" dialog.

Select "File" on PM+ menu bar, then "New Workspace...".



"New WorkSpace" dialog opens

<Description of items>

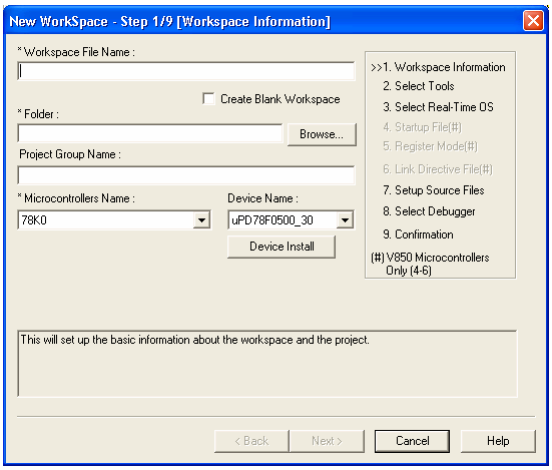
Workspace File Name:
-> Specify the name of the workspace file that manages the project files.
.prw is automatically suffixed as the file type.
A project file (.prj) of the same name is simultaneously created.

Folder:
-> Specify the folder for saving the workspace file by writing its absolute path.
This item can be selected from a reference dialog box by pressing the Browse... button.

Project Group Name:
-> Specify this item if wishing to manage multiple projects together in function units.
If nothing is specified, this item is the same as the workspace file name.

Microcontroller Name:
-> Specify the name of the microcontroller to be used.

Device Name:
-> Specify the name of the device to be used.



The concrete information set here is described on the following pages

Input the workspace information setting as follows.

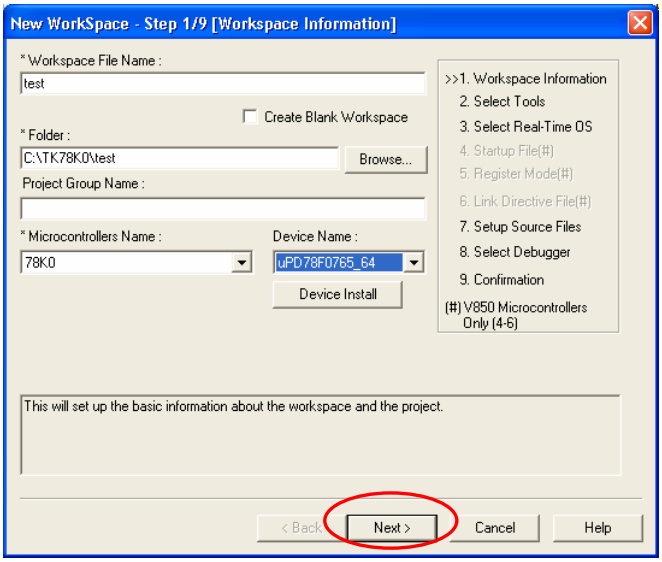
Workspace file name
→ test

Folder
→ C:¥TK78K0¥test

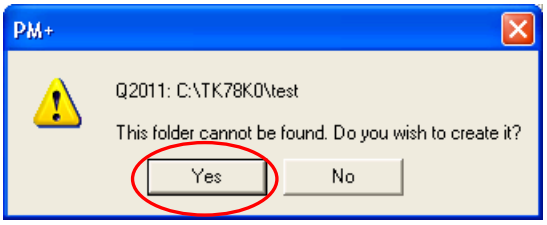
Project Group Name
→ (no input)

Microcontroller Name
→ 78K0

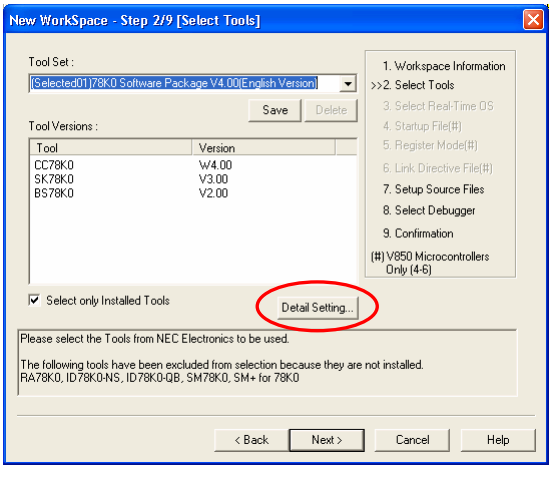
Device Name
→ uPD78F0765_64



Click Next > button

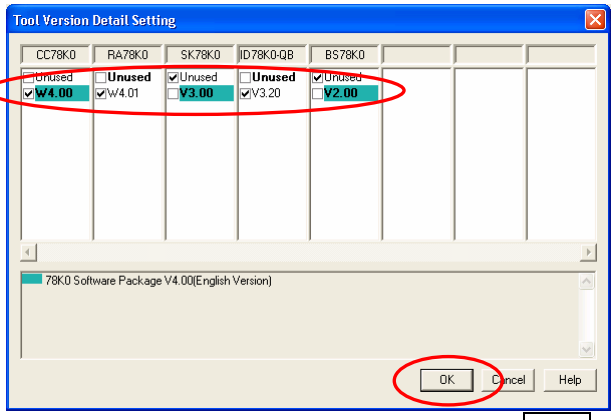


Click Yes button

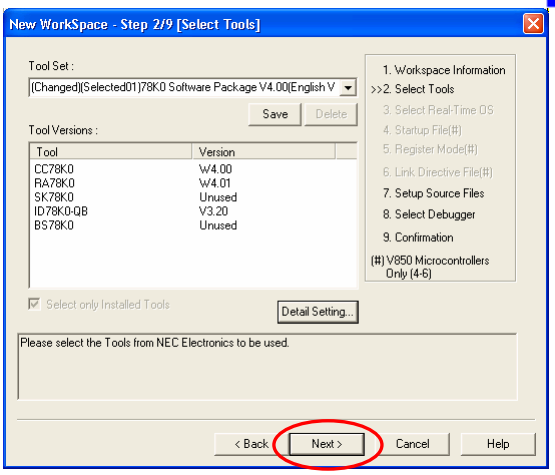


Click Detail Setting button

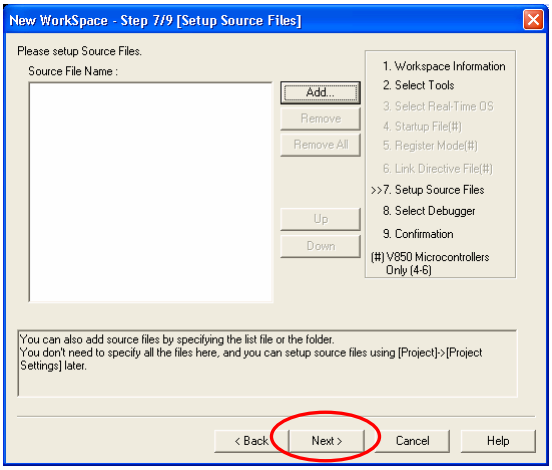
Set the version of tools as follows.
CC78K0 : W4.00
RA78K0 : W4.01
ID78K0-QB : V3.20



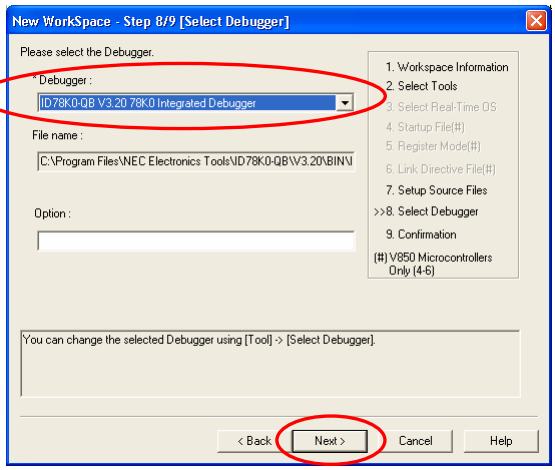
Select tools as above screenshot, then click OK .



Click Next >

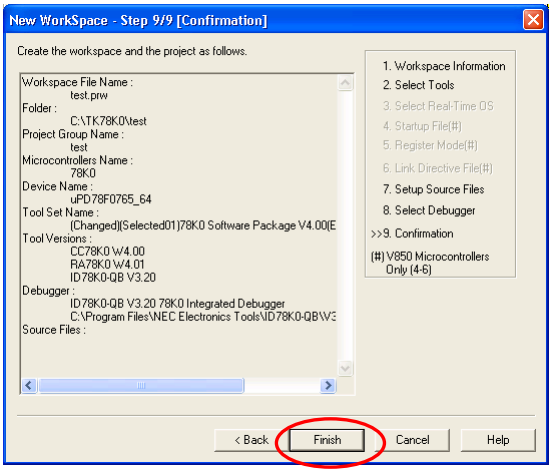


Click Next >



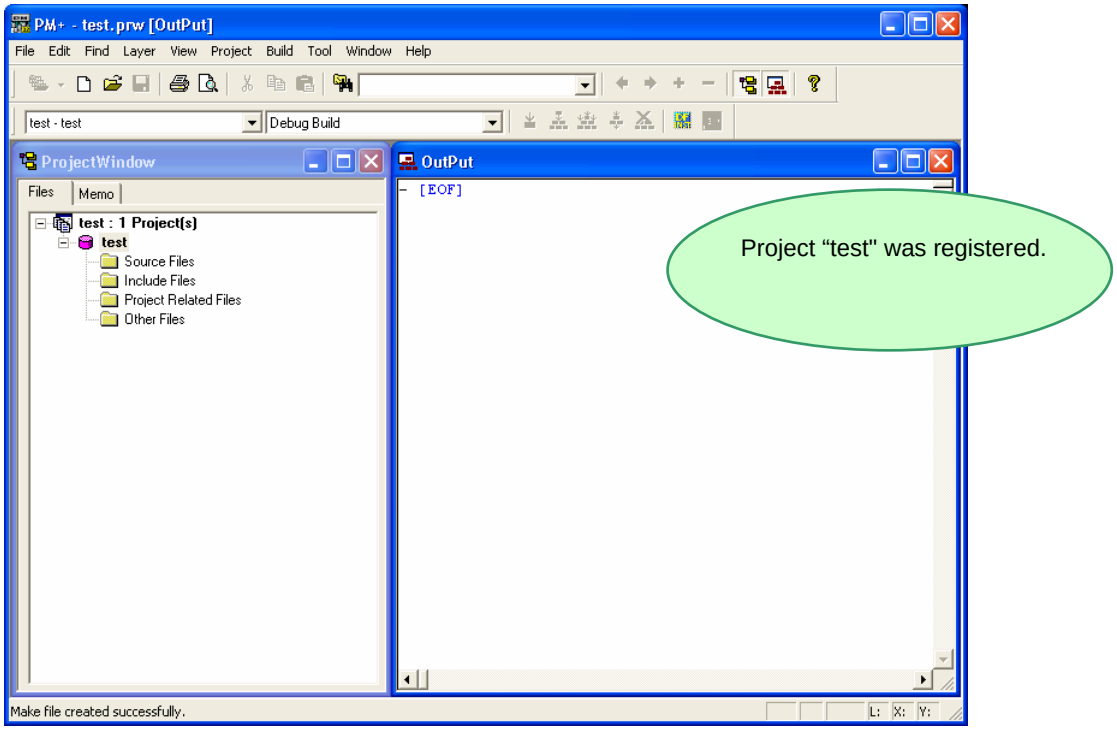
Select ID78K0-QB V3.20

Click Next >



Click Finish

Check the project information settings



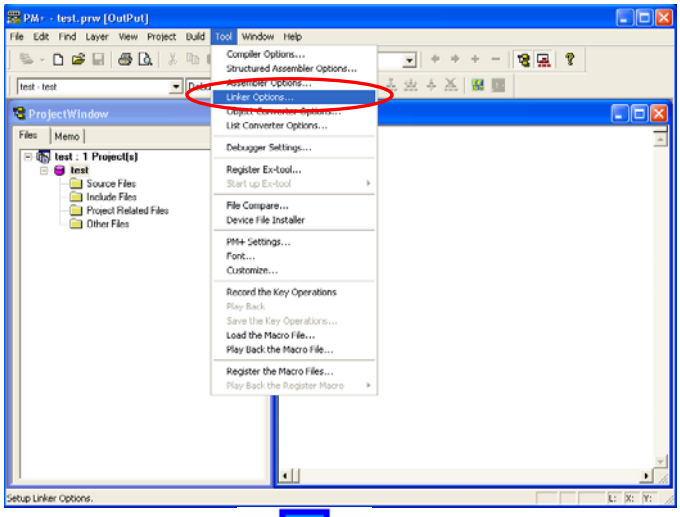
This completes workspace and project creation.

Additional source files can be registered at any time thereafter.

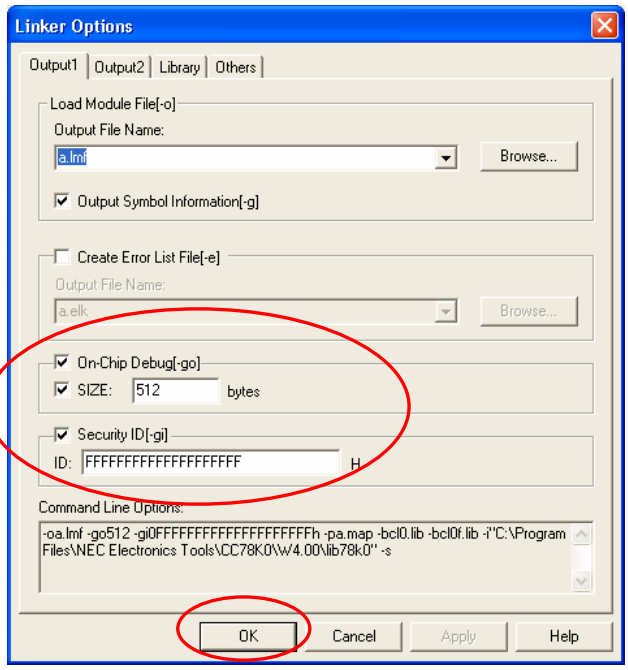
For details, refer to "5.2 Register additional source file".

Next, on chip debug function can be used.

[Tool] → [Linker Options...]is selected.



Checks are put in "On-Chip Debug" and "SIZE". Then input a default value of "512" into the "SIZE".
A check is put in "Security ID".
The actualities of "FFFFFFFFFFFFFFFFFFFF" are taken if there is no problem in the value of ID in security.



Push the OK button.

Next, please add the following "option.asm" file and set the method of attesting security ID as the setting of the option byte. Please refer to the user's manual of the device for details of the option byte.
Refer to "78K0/Kx2C user's manual" for the details of the security ID.
"option.asm" please the file must be included in the sample program, and copy this file and use it.
Please refer to "Registering additional source file of next page for the method of adding the file.

```
OPT       CSEG     AT 0080H

OPTION: DB       00H
        DB       00H
        DB       00H
        DB       00H
        DB       03H

TMP       CSEG     AT 0FEDFH
ttmp:  DS        1

END
```

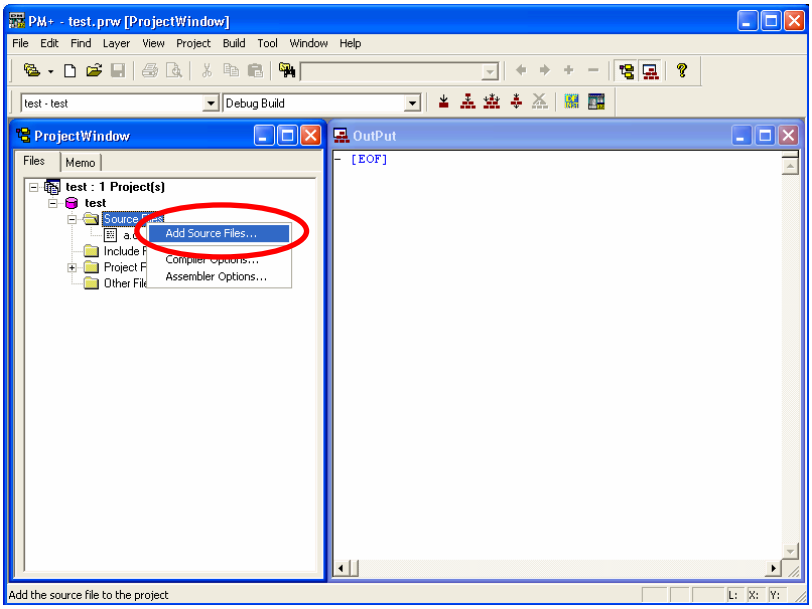
Example for the Option.asm

5.2 Register additional source file

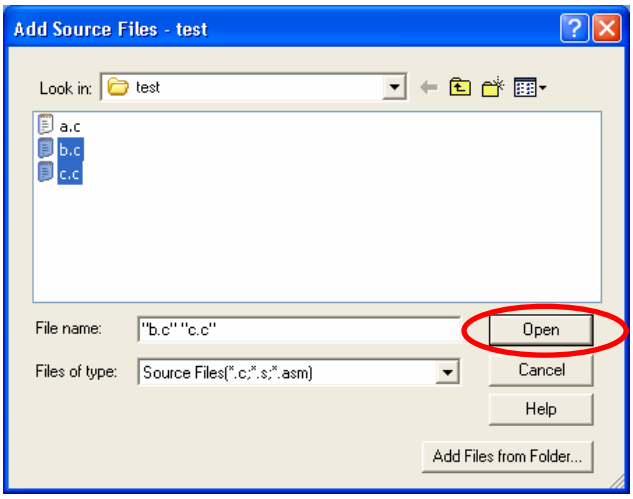
Now, register additional source files.

The following example shows the additional registration of source files "b.c" and "c.c" with source file "a.c" already registered.

Place the cursor on the source file in the Project window of PM+, and select [Add Source Files...] displayed in the right-click menu.

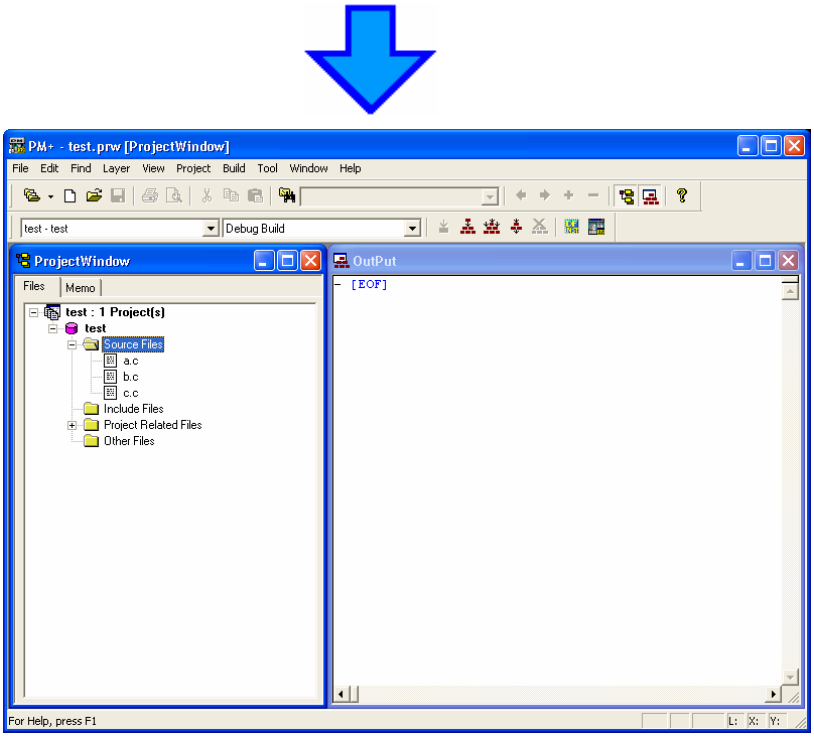


Select source files "b.c" and "c.c", then click Open



Multiple source files can be selected by clicking them with pressing Ctrl key.

Source file "b.c" and "c.c" are additionally registered to the project.



5.3 Debugger tips

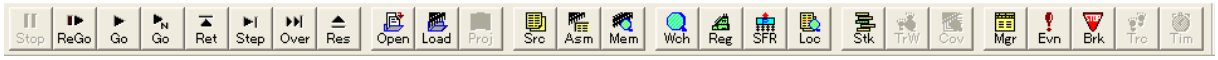
This section describes some useful techniques for the debugger (ID78K0-QB).

5.3.1 Change display of buttons

Execution controls (run, stop, step-in debugging, reset, etc) and opening functional window can be made by below buttons. However, it could be difficult to know which button does what.

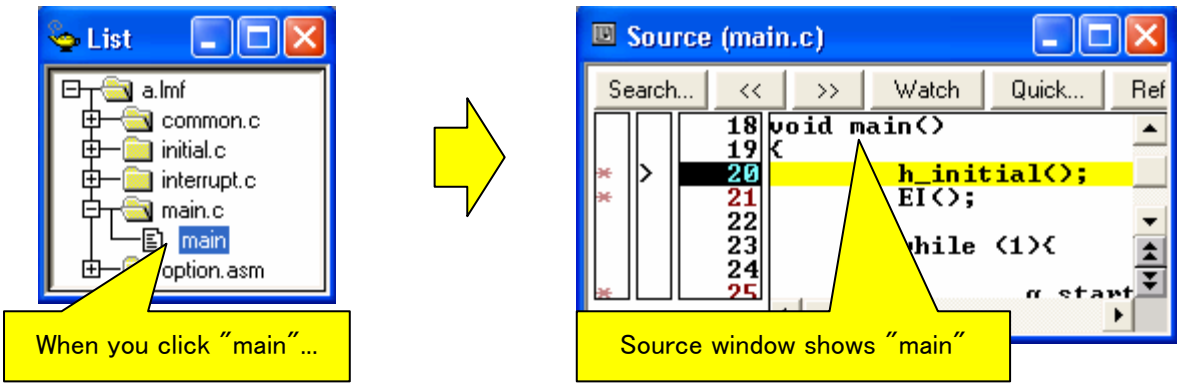
In this case, select "Options" on menu bar, then "Debugger Options". Check "Pictures and Text" on setting area.

With this setting, the buttons display the text as well, so that it is easier to know what they are.



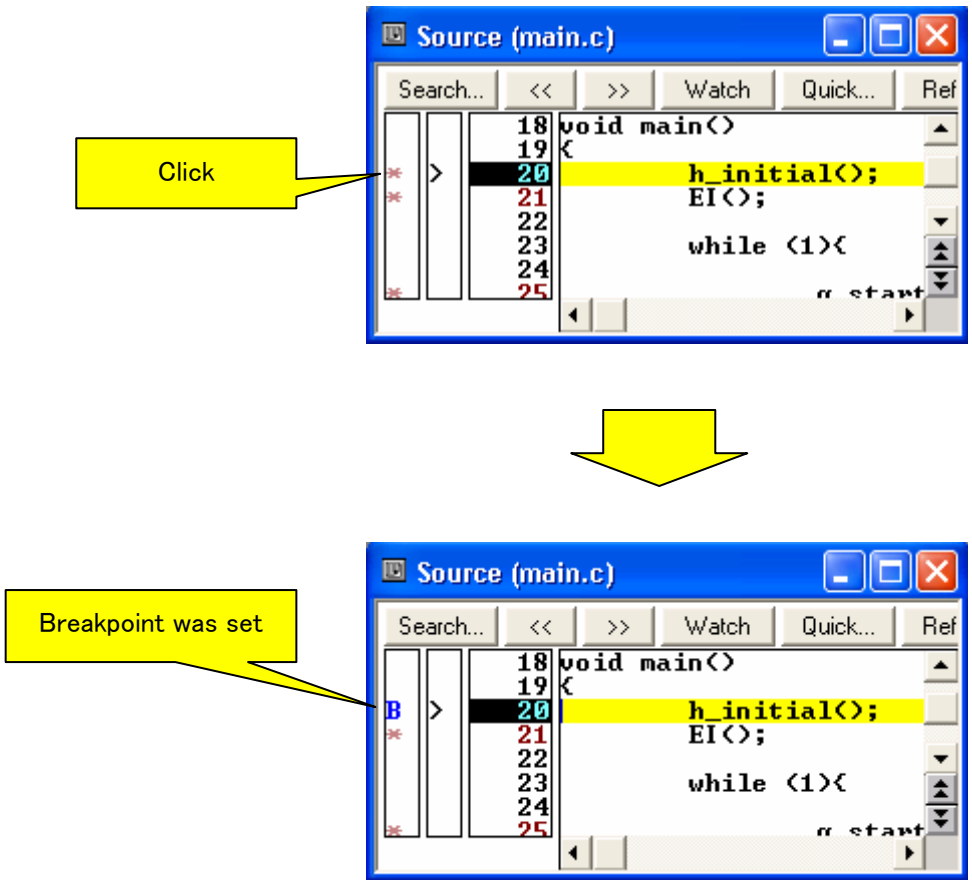
5.3.2 Display source list and function list

When you wish to see source file list or function list, select "Browse" on menu bar, then "Other" -> "List" to open the list window. The information in the windows is synchronized. Therefore, it is not just for referring to the list, but it is useful when you wish to update files or functions.



5.3.3 Set/delete breakpoints

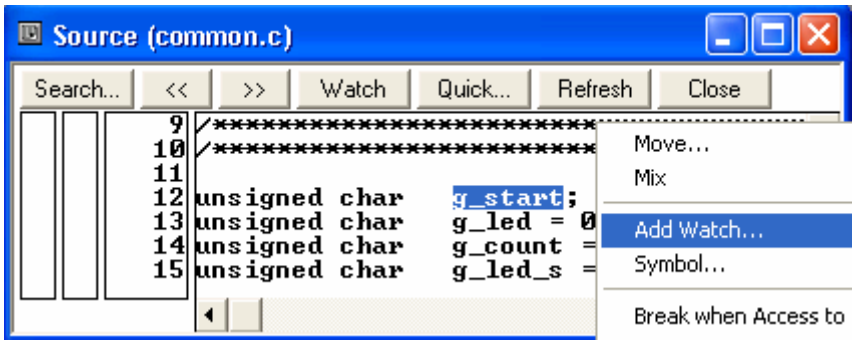
Breakpoints are executed by clicking lines in which " * " is displayed
"B" is displayed in the line where a breakpoint is set.
Breakpoints are deleted by clicking "B".



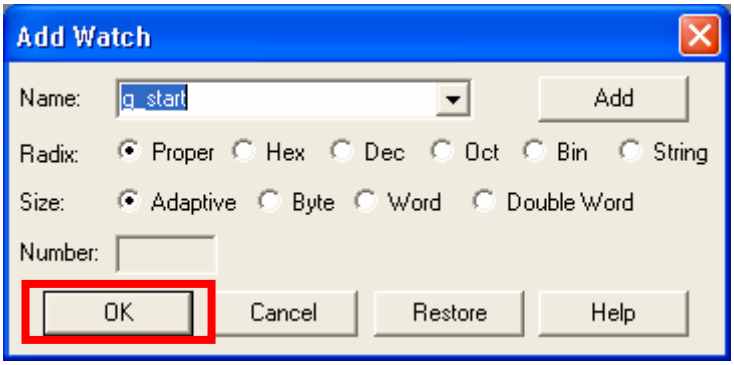
5.3.4 Display global variables

With using Watch Window, you can display global variables. There are several ways to register global variables to watch window. In this section, how to register from source window is described.

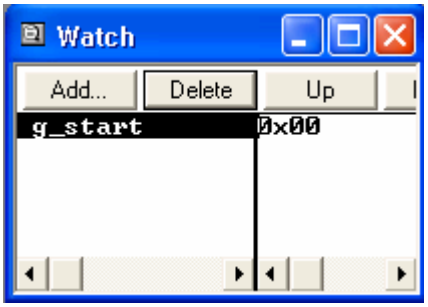
①Right-click the variable on source window, then select "Add Watch..."



②Add Watch dialog opens. Click OK .



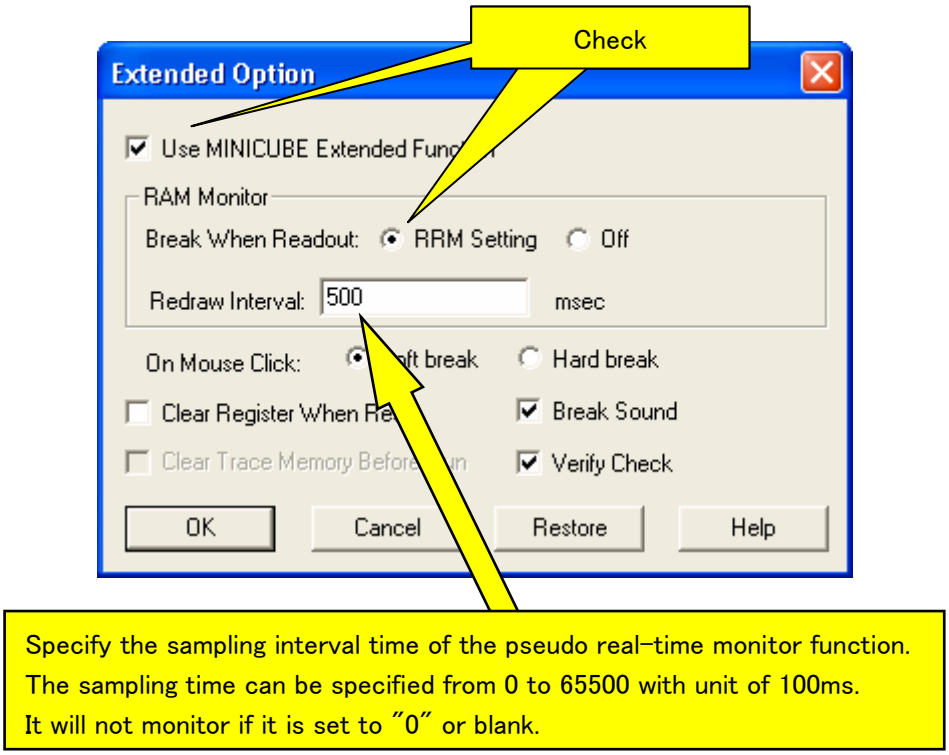
③Adding a variable to watch window is completed.



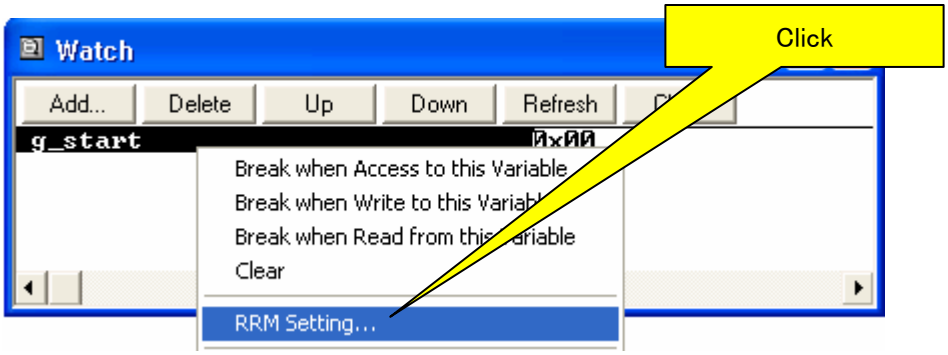
5.3.5 Display global variables while programs are running

Global variables can be referred by the pseudo real-time monitor function even when the programs are running.

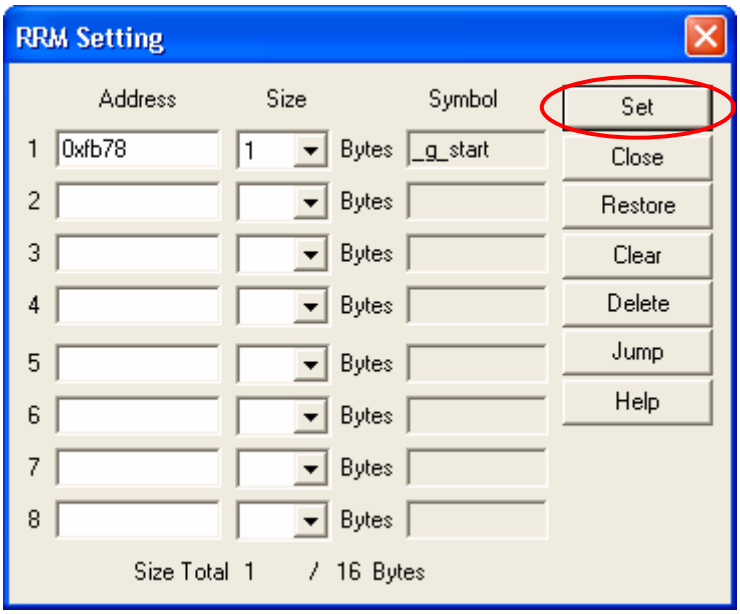
①Select "Option" on menu bar, then "Extended Option...".
②Configure the settings as "Use MINICUBE Extended Function" and "RRM Setting".



③Add a global variable to watch window.
④Right-click the global variable on watch window, then select "RRM Setting..."



③Click the "Set", then Close the "RRM Setting "window.



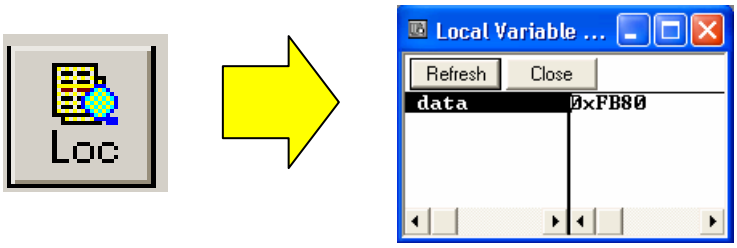
This completes the settings.

Note:

- The user program momentarily breaks upon a read.
- It is recommended to close the memory window when you use the pseudo real-time monitor function.

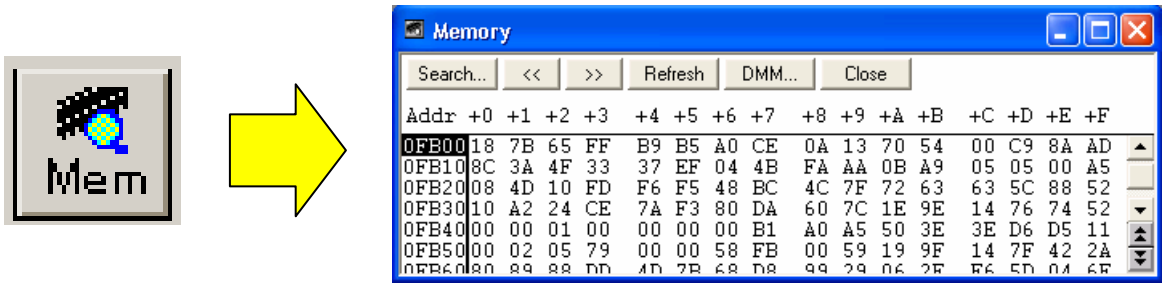
5.3.6 Display local variables

Local variable window is used to display local variables.
By clicking the button below, you can open the local variable window.
Unlike global variables, local variables cannot be displayed when programs are running.

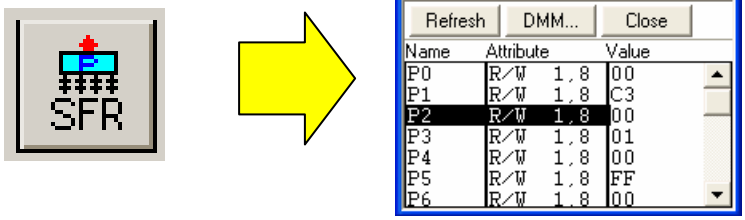


5.3.7 Display memory and SFR contents

By clicking the button below, you can open the memory window.



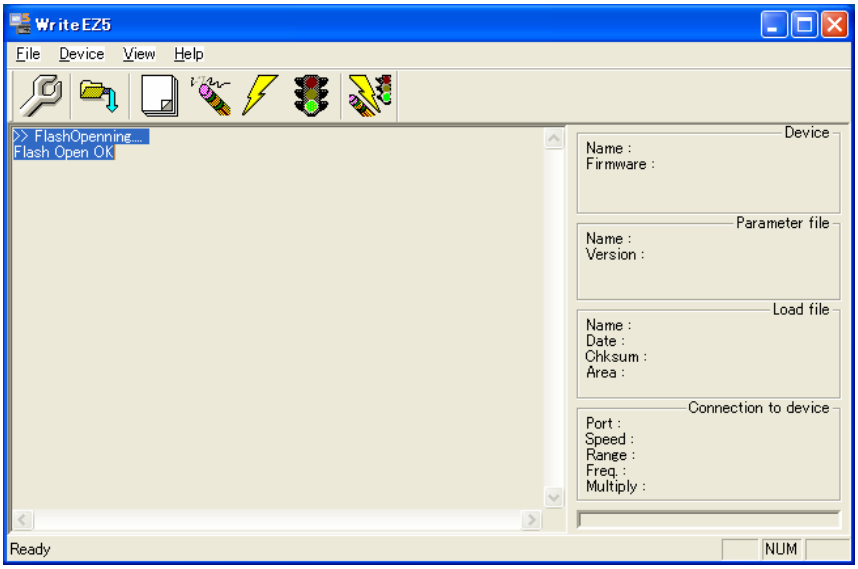
By clicking the button below, you can open the SFR window.



5.4 Erase microcontroller built-in flash memory

WriteEZ5 can erase the flash memory, when you forgot the security ID or you set unexpected value. By erasing the flash memory, you can reset the security ID to "FFFFFFFFFFFFFFFFFFFF".

①Start WriteEZ5 from NEC Electronics Tools.



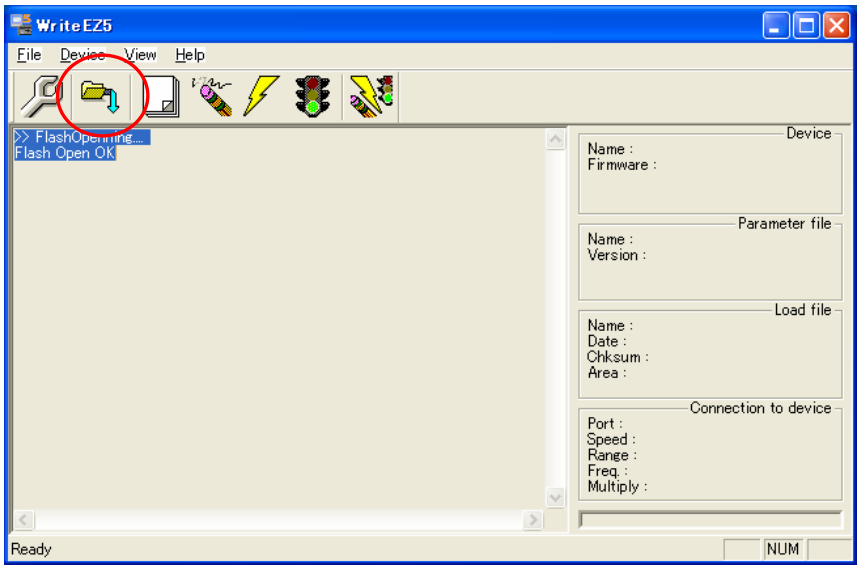
②Set TK-78K0/KE2C switches, and then connect the PC.

SW3

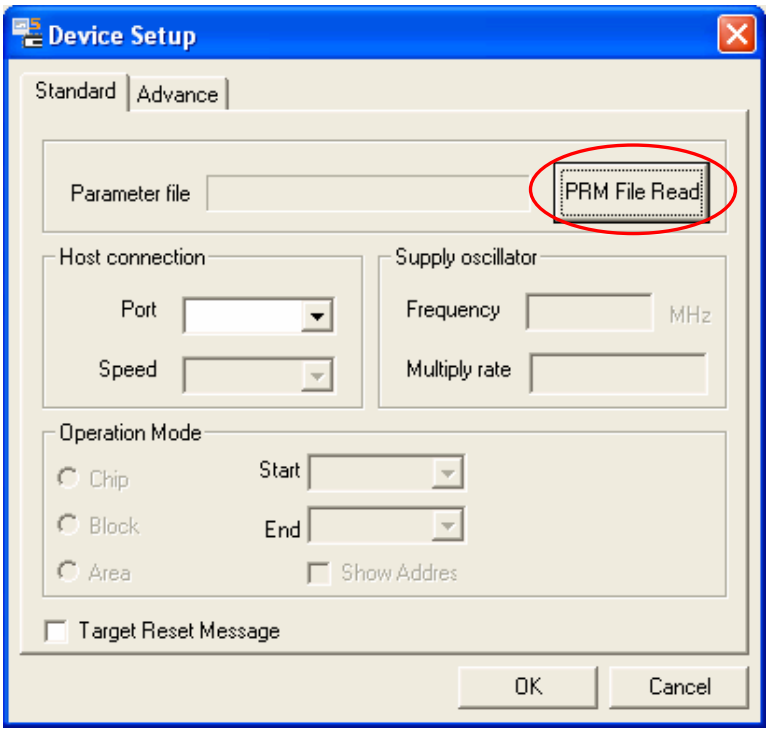
Bit 1	ON
Bit 2	ON
Bit 3	ON
Bit 4	OFF

SW4 UART 側

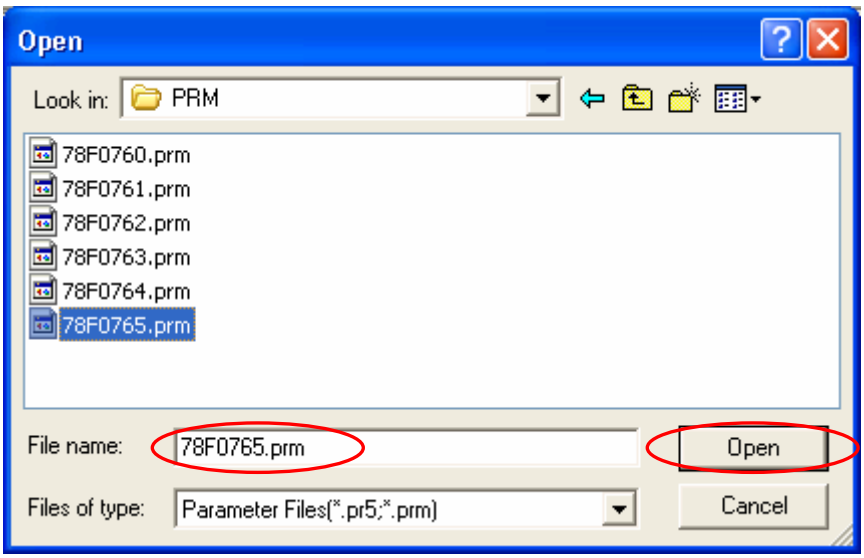
③Click the setup button.



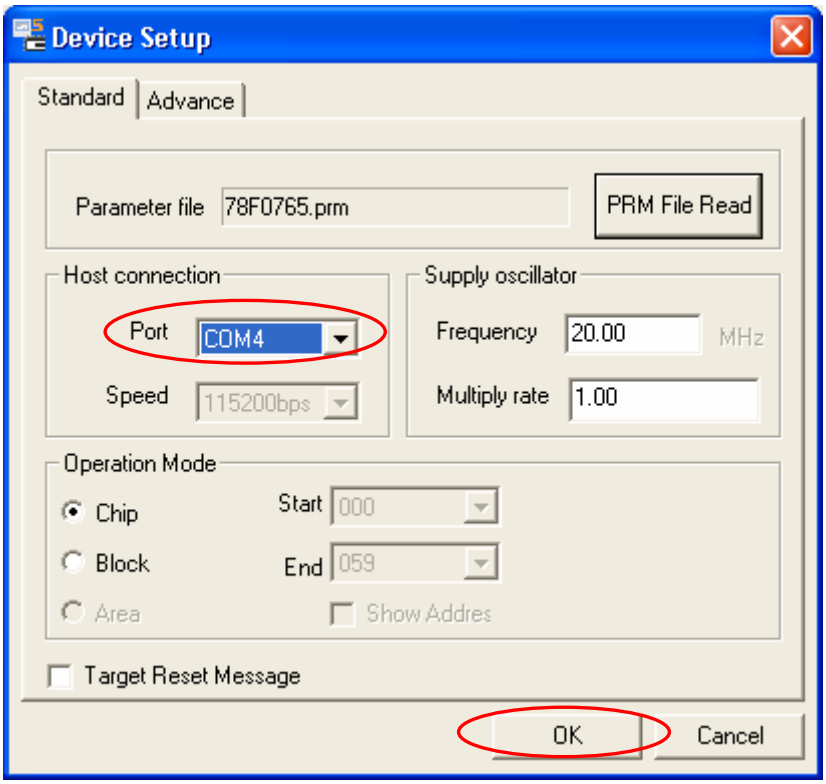
④Click "PRM File Read" button.



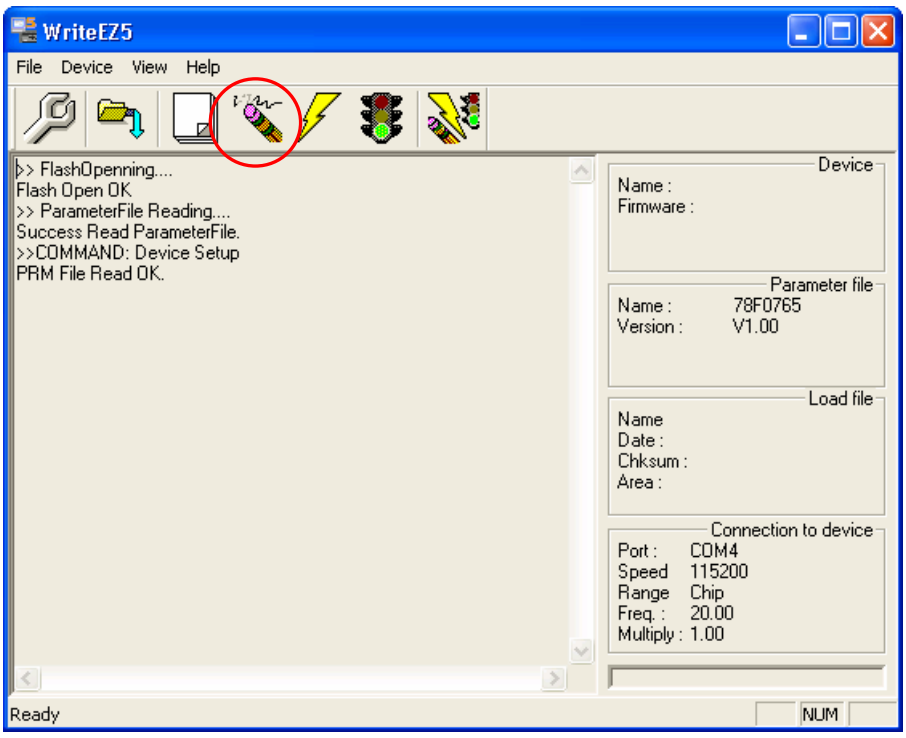
⑤Please select "78F0765.prm" in the directory of "PRM" in the CD-ROM.



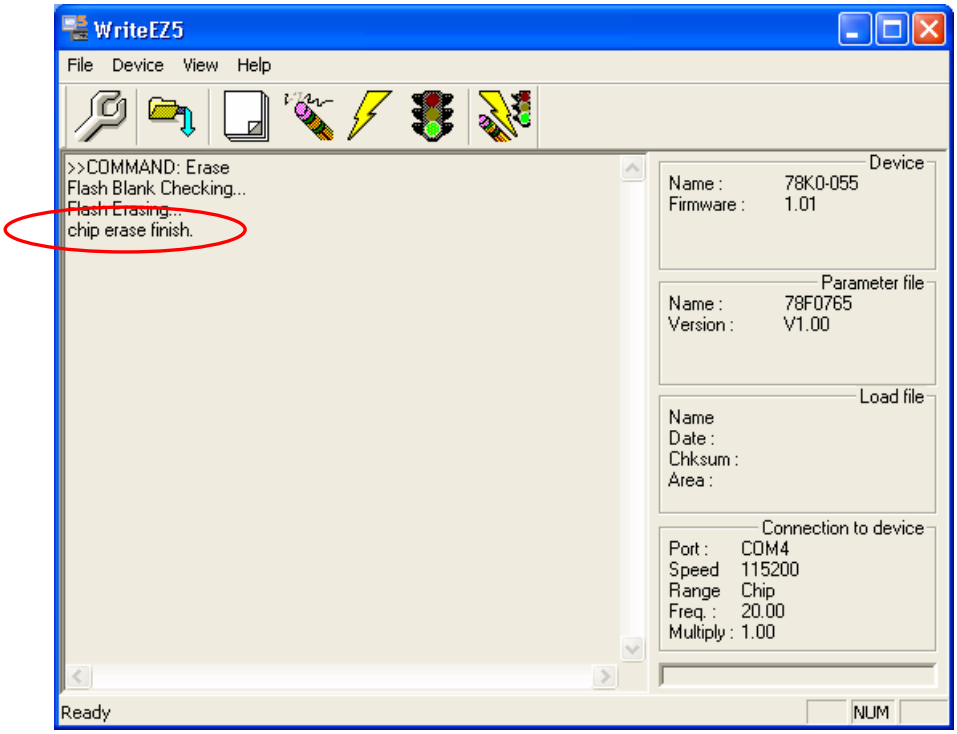
⑥Select the COM port that TK-78K0/KE2C is connected.



⑦Click "Erase" button.

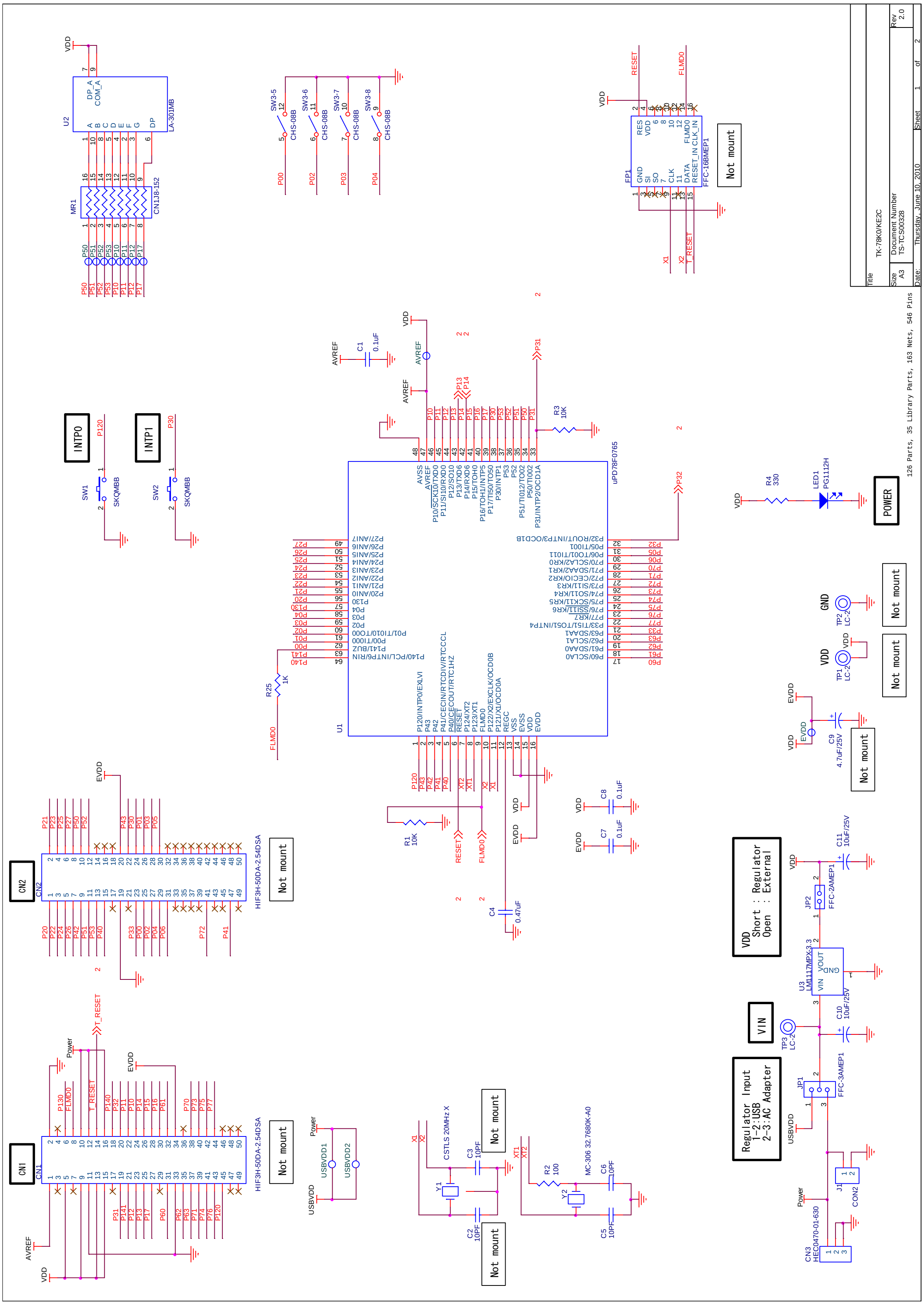


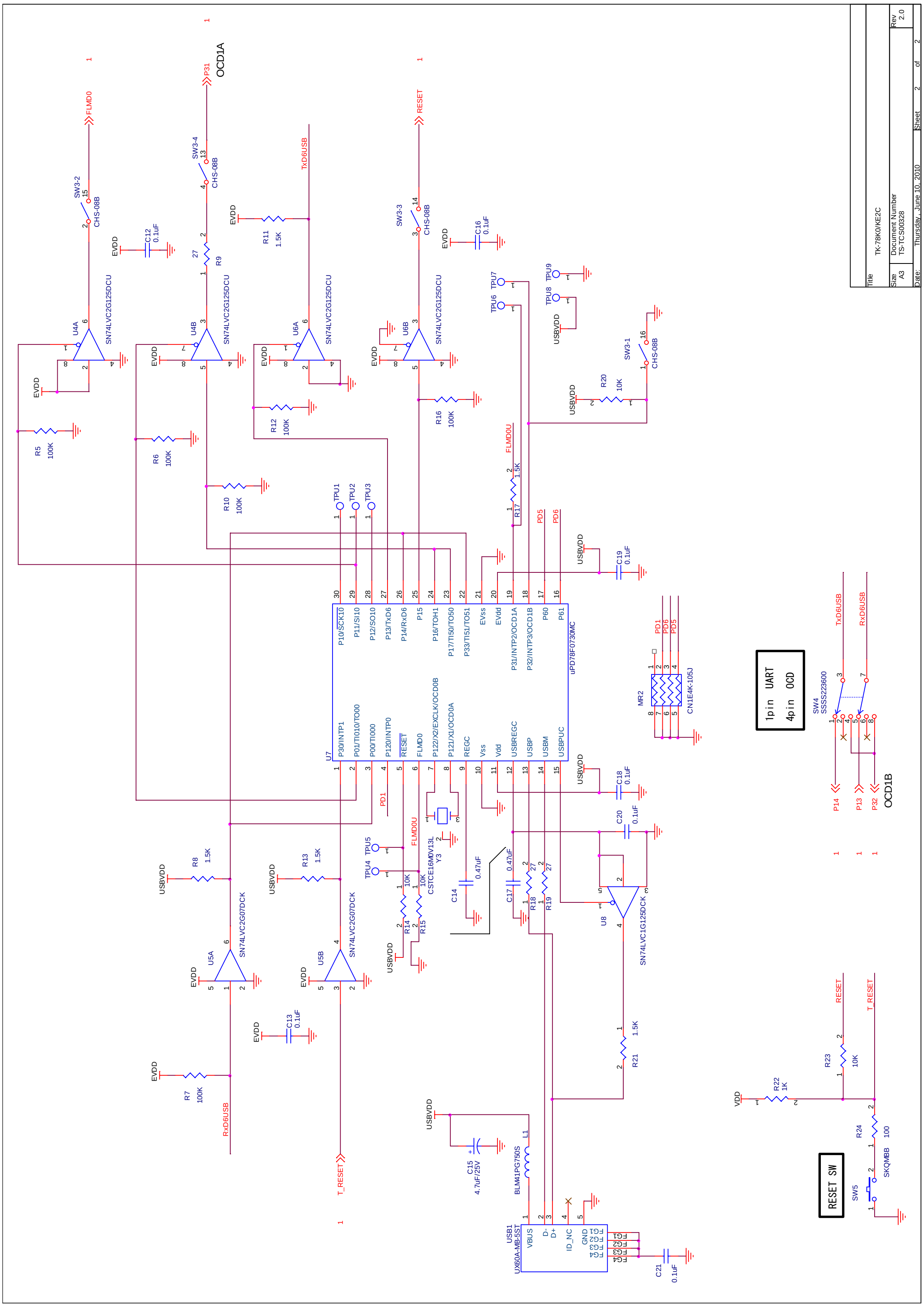
⑧Erasing the flash memory is completed when "chip erase finish" is displayed like below screenshot.



5.5 Circuit diagram

From following page, it shows the circuit diagram of the demonstration kit.


CN1
CN2
INTP0
INTP1
Not mount
POWER
VDD
GND
VIN
Regulator Input
1-2:USB
2-3:AC Adapter
VDD
Short : Regulator
Open : External
TK-78K0/KE2C
TS-TCS00328
Rev 2.0


Title
TK-78K0/KE2C
Size
A3
Document Number
TS-TCS00328
Rev
2.0
Date:
Thursday, June 10, 2010
Sheet
2
of
2
uPD78F0730MC
U7
SN74LVC2G125DCU
SN74LVC2G07DCK
SN74LVC1G125DCK
USB1
UX60A-MB-5ST
CSTCE16M0V13L
BLM41PG750S
SKQMBB
SSSS223600
CN1E4K-105J
1pin UART
4pin OCD
RESET SW
OCD1A
OCD1B
FLMD0
RESET
T_RESET
TxD6USB
RxD6USB
USBVDD
EVDD
VDD