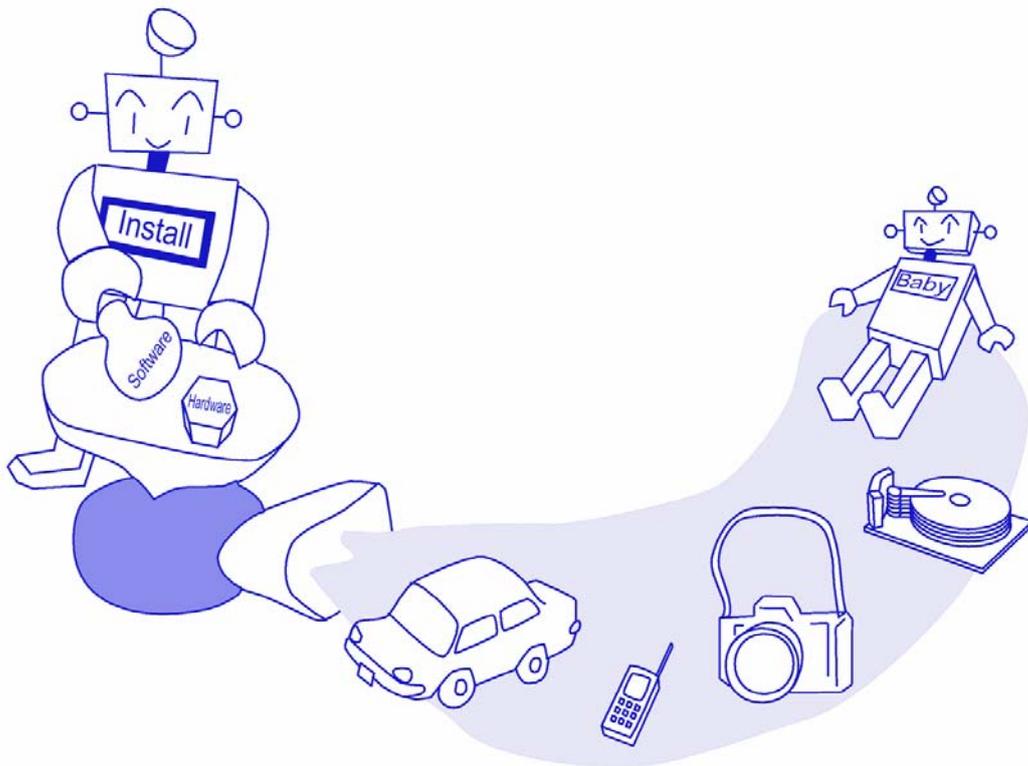


Welcome to the world of TK-78K0.  
This document described as part of the TK-78K0/KF2 evaluation kit  
using sample program.

# TK-78K0/KF2 Evaluation kit Tutorial Guide



Date published October, 2006 (3<sup>rd</sup> Edition)

© NEC Electronics Corporation 2005,2006

# Introduction

Target Reader	This document intended for users for the 78K0 series microcomputer development tools. A basic knowledge of microcomputers, C language, assembles, and Windows™ operating is assumed.
Purpose	This material has aimed the understanding of the customer of a basic operation method of the development tool for 78K0 series appended to TK-78K0/KF2 evaluation kit. Readers will be able to further their understanding by actually using the development tool while reading this document.
How to use this manual	This PDF file should be read with <b>Adobe™ Acrobat™ Reader5.0 or later.</b>
Organization	<p>This manual consists of the following contents</p> <p><a href="#">Chapter 1 Preparations</a></p> <p>→ <b>This chapter provides an outline of the 78K0 series development used in this guide and describes the sample program installation method.</b></p> <p><a href="#">Chapter 2 Practical</a></p> <p>→ <b>This chapter allows the reader to experience the basic operations of PM plus and the integrated debugger using sample programs.</b></p> <p><a href="#">Chapter 3 Other</a></p> <p>→ <b>It explains the new making method of the PM plus workspace(project), the additional registration method of the source file, and how to use of PG-FPL3.</b></p>
Caution	<ul style="list-style-type: none"><li>- This contents of this document are subject to change without prior notice.</li><li>- No part of this document may be reproduced without the prior written permission of NEC Electronics Corporation.</li><li>- NEC Electronics Corporation does not make any guarantees nor does if grant any license regarding intellectual property rights on any other right of NEC Electronics Corporation or third parties pertaining to the use of the products and the use of the informaiton describes in this document. NEC Electronis Corporation shall not be held liable for problems related to the right of third parties that arise from the above uses.</li><li>- This circuits, software, and appurtenant information describes in this document is provided solely for the sake of semiconductor device operation examples or application examples. Therefore, in case these circuit, software, and/or information items are to be used for a customer's equipment, the design of said equipmentt shall be the responsibility of the customer. NEC Electronics Corporation declines any and all responsibility for damage arising from the use of such circuits, software, and/or information by a customer of a third party.</li></ul>

Trademark

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

Adobe and Acrobat are trademarks of Adobe Systems Incorporated (Adobe System Incorporated).

Other company names and product names that appear in this document are the registered trademarks or trademarks of their respective companies.

# Chapter1 Preparation

This chapter provides an outline of the development tools used in this document and describes how to install the sample programs.

The sample program function only for the development tools included in the TK-78K0/KF2 evaluation kit in this document

# Used development tools

This section outlines the development tools used in this document.

The name of the development tools and their main function are as follows.

- **Device file DF780547 V2.10**

Since the device file contains the device-specific information, this file is required for using the development tools.

The sample used in this document has been created as a 78K0/KF2(uPD78F0547)

- **Integrated development environment PM plus V5.21**

This is an integrated development environment platform that operates on Windows.

An editor with an idea processor function is provided as the Edit window, allowing efficient development linked with development tools such as a compiler and a debugger.

- **C compiler CC78K0 W3.70(size restriction version)**

It is "Object size restriction(32Kbyte)" compiler for 78K0 series.

It is a program that translates the source program written by C language for 78K0 series and C language in accordance with ANSI-C into the machine language.

The source program written by C language is input, and the object program that becomes an input of the source program and the linker that becomes an input of the assembler for 78K0 series is output.

- **Assembler RA78K0 W3.80 (size restriction version)**

It is "Object size restriction(32Kbyte)" assembler for 78K0 series.

The source program written by the assembly language for 78K0 series is input, and the generic name of a series of program that object program of the machine language. The output object program becomes an input of the flash programmer and debugger.

- **78K0 integrated debugger ID78K0-TK V2.02**

This is windows-based software that runs on a host PC.

It is integrated debugger that achieves debug at source level C.

Source debug of the reference, the change in the variable, and the step execution, etc. by such source line can be done easily and efficiently.

- **Program for writing to on-chip flash memory of microcontroller PG-FPL3**

This is a Windows-based software used to write programs to the on-chip flash memory of a microcontroller.

It is possible to write/delete it to a built-in flash memory of 78K0/KF2 by connecting TK-78K0/KF2 with the PC with an attached USB cable.

The above-listed development tools must be installed in order to operate the sample programs described in this document. For the installation method, refer to the TK-78K0/KF2 User's Manual software version (¥DOC¥TK78K0-KF2-SW\_E.pdf). Also refer to the TK-78K0/KF2 User's Manual for the restrictions on the development tools.

The documentation assumes that the name of the installation destination folder is the default name "NECTools32"

# Sample environment

This section describes the preparations for the sample programs used in this document. The preparations consist of installing the sample programs in the customer's environment.

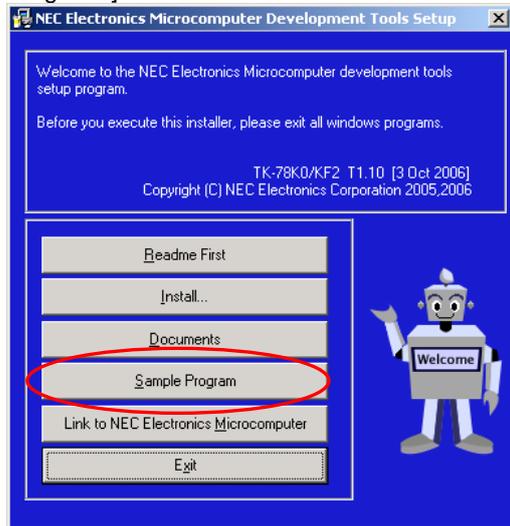
To uninstall the sample programs, delete the folder that was specified as the installation destination.

The installation method and the installation destination are described below.

● Installing the sample programs

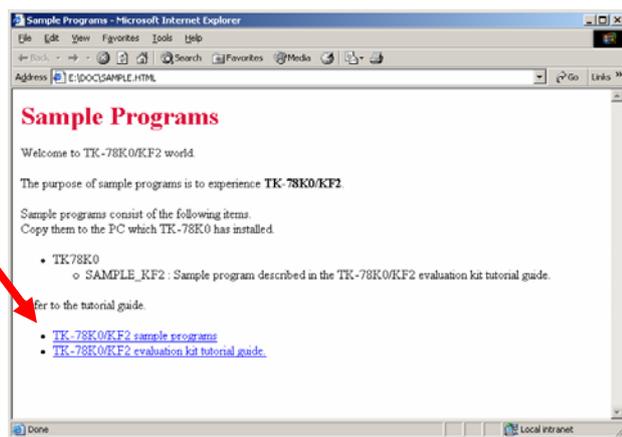
Inset the CD-ROM disk provided as part of the TK-78K0/KF2 products in the CD-ROM drive. The [NEC Electronics Microprocessor Development Tools Setup] screen automatically appears.(if this screen does not appear automatically, start setup.exe from Explorer. etc.)

Press the **Sample Program** button to start the WWW browser, and then click the[TK-78K0/KF2 Sample Programs] link.

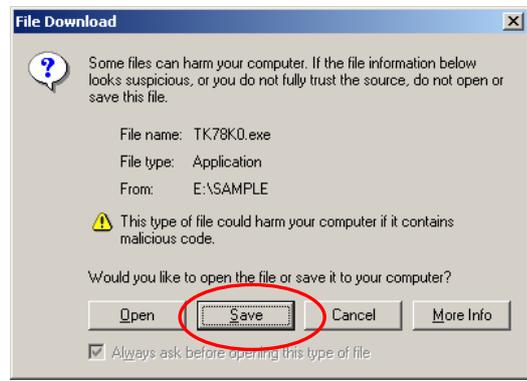


The WWW browser start up.

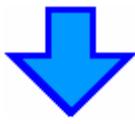
Click the [TK-78K0/KF2 Sample programs] link youcan also download the [Tutorial Guide]

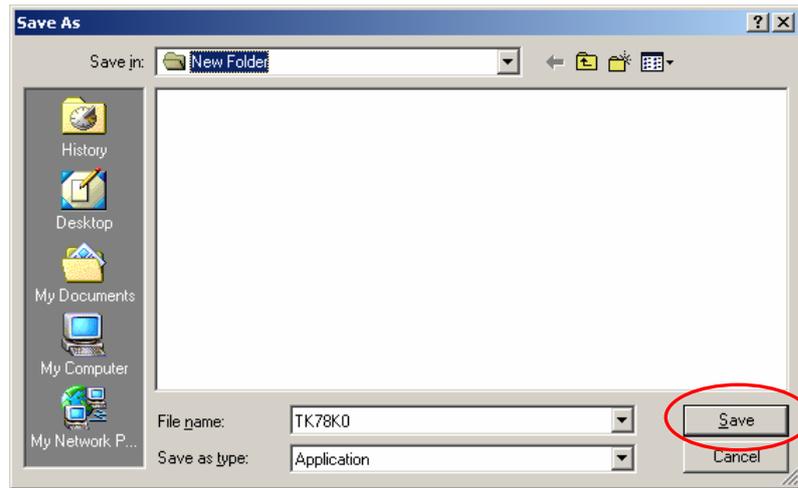


When[TK-78K0/KF2 Sample Programs] is clicked, the following download confirmation window appears.



Please push the **Save** button.



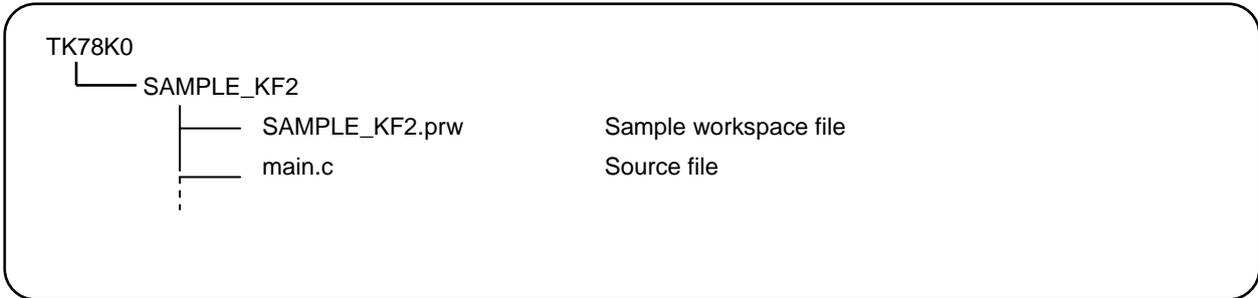


After specifying the download destination folder, press the **Save** button. The self-extraction sample program set(TK78K0.exe) is copied to the specified folder.

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.

## ● Folder structure of sample programs

When the sample program set is decompressed, the files are to the following folder structure.



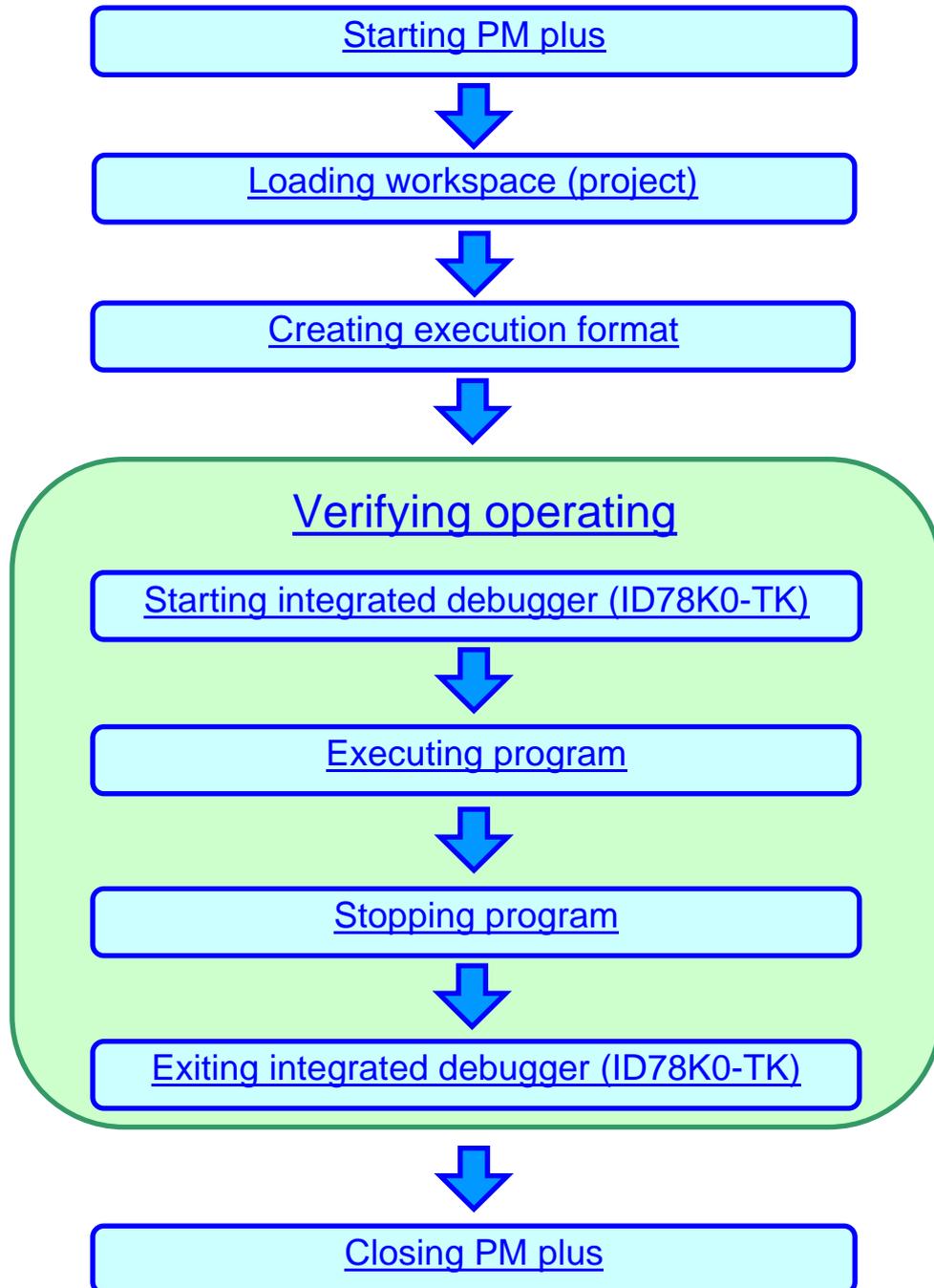
# Chapter2 Practicals

This chapter lets the reader experience the operation of the completed 78K0/KF2 program using the integrated debugger (ID78K0-TK)

Here, the interval when the count improvement is done becomes long if 7seg LED pushes the count improvement doing and SW4 if SW3 on CPU board of TK-78K0/KF2 is pushed as a program for 78K0/KF2, and the stopping program is used.

This chapter is designed to give an understanding of the concrete operation method of the development tools (PM plus, ID78K0-TK) and the outline of the project files required when creating application programs, through building of sample programs and operating ID78K0-TK.

The overall flow is as follows.



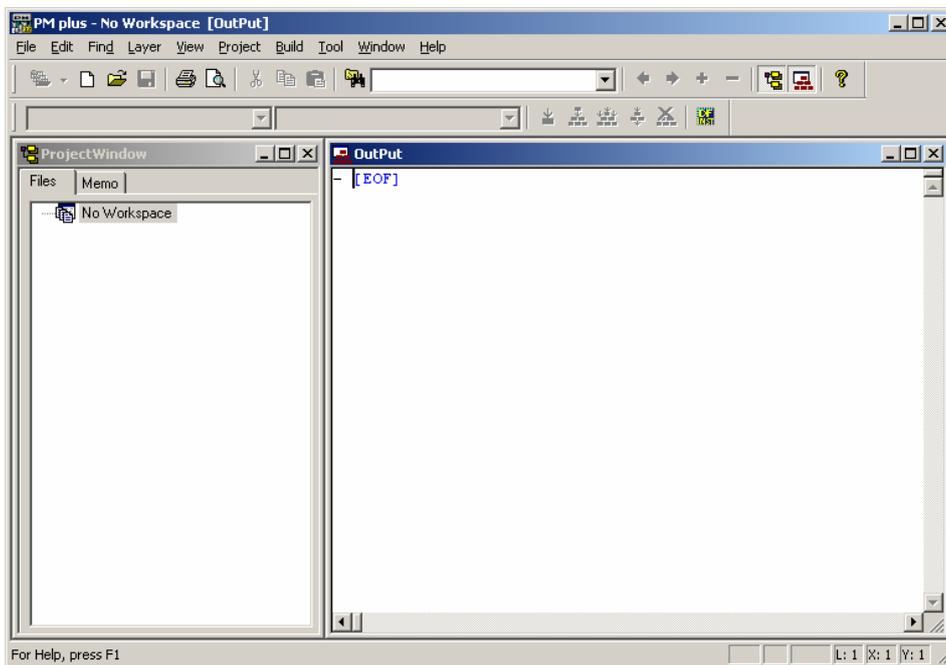
# Starting PM plus

Let's now actually use the development tools.

First, start PM plus.

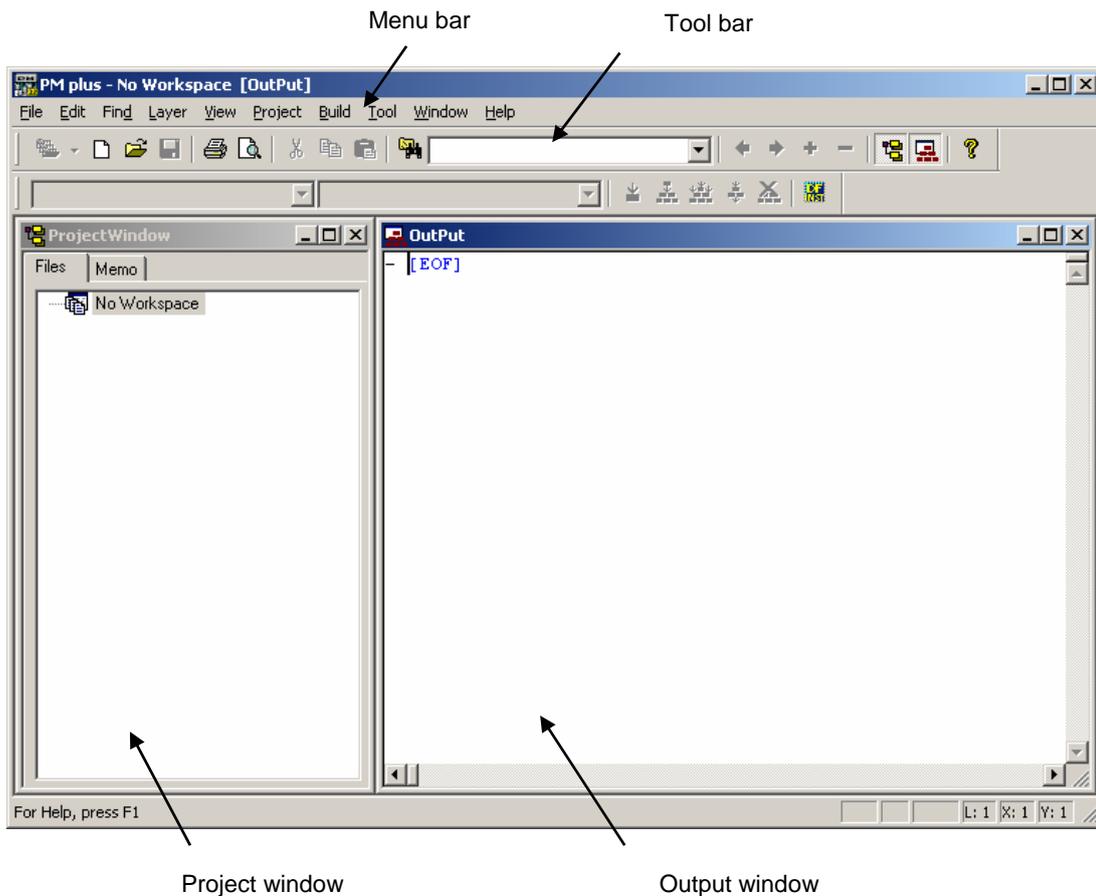
In the Windows Start menu, select [Programs],[NEC Tools32],[PM plus]

PM plus start up



# Introduction to PM plus

In PM plus, 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 or 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 the PM plus User's Manual.

**What is a project?**

A project is the unit in which PM plus executes management, and refers to an application system and environment development based on PM plus.

PM plus compiles project information and saves it in a “project file”, from which it is then referenced.

**What is a project file?**

A project file is a file to which information such as the source file to be used in the project, the device name, the tool options for compiling, and the editor and debugger to be used have been saved as “project information”.

The file name format is “ $\Delta\Delta\Delta\Delta$ prj”.

Project files are created in folders that are set when creating new workspaces.

**What is a project group?**

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

The target devices of each project that can be registered to one project group must be the same.

**What is a workspace?**

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

The file name of multiple project files is saved to a workspace file for referencing.

The file name format is “ $\Delta\Delta\Delta\Delta$ prw”.

# Loading workspace (project)

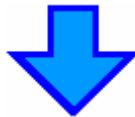
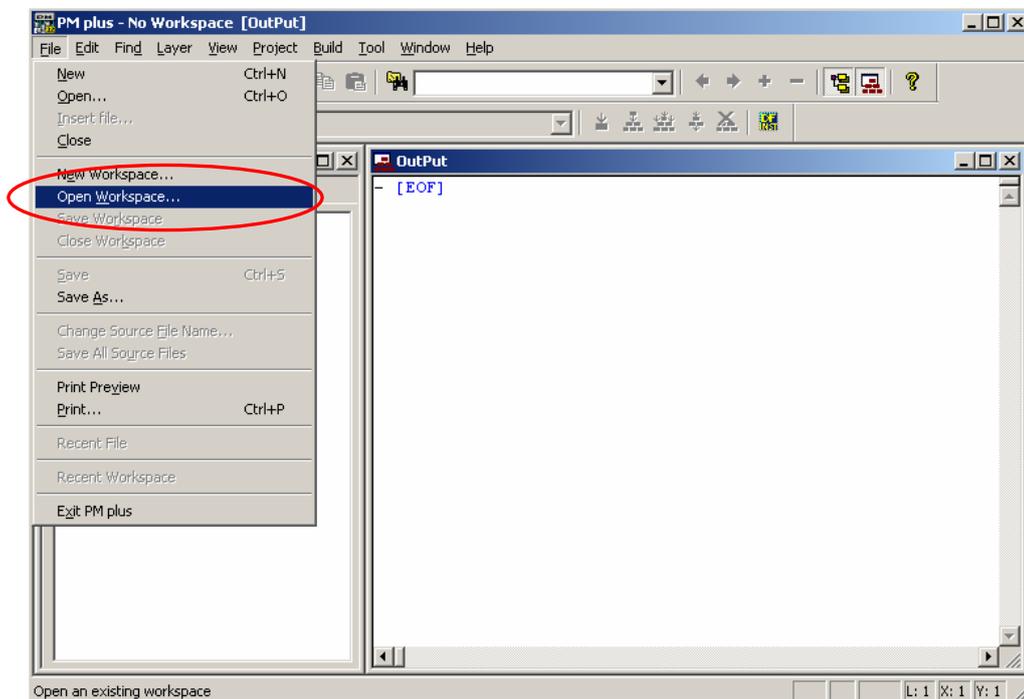
In this chapter, you will use an already created workspace.

➡ The method for creating a new workspace is described in “[Chapter 3 Other](#)”

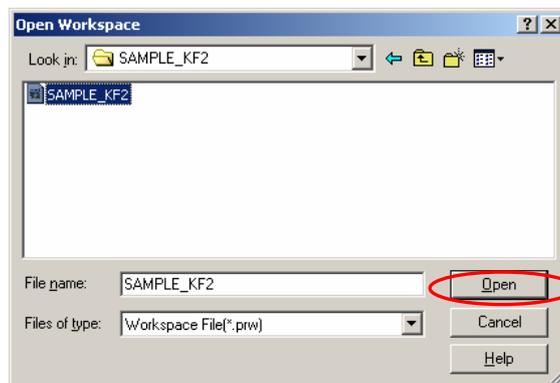
The source file name of the 7seg LED count up program and the 78K0/KF2 setting emulated with ID78K0-TK are saved to the workspace used in this chapter.

In the PM plus menu, select [File(F)]→[Open Workspace(W)...] and specify “SAMPLE\_KF2.prw”

➡ If you have not set an environment, refer to “[Sample Environment](#)”.



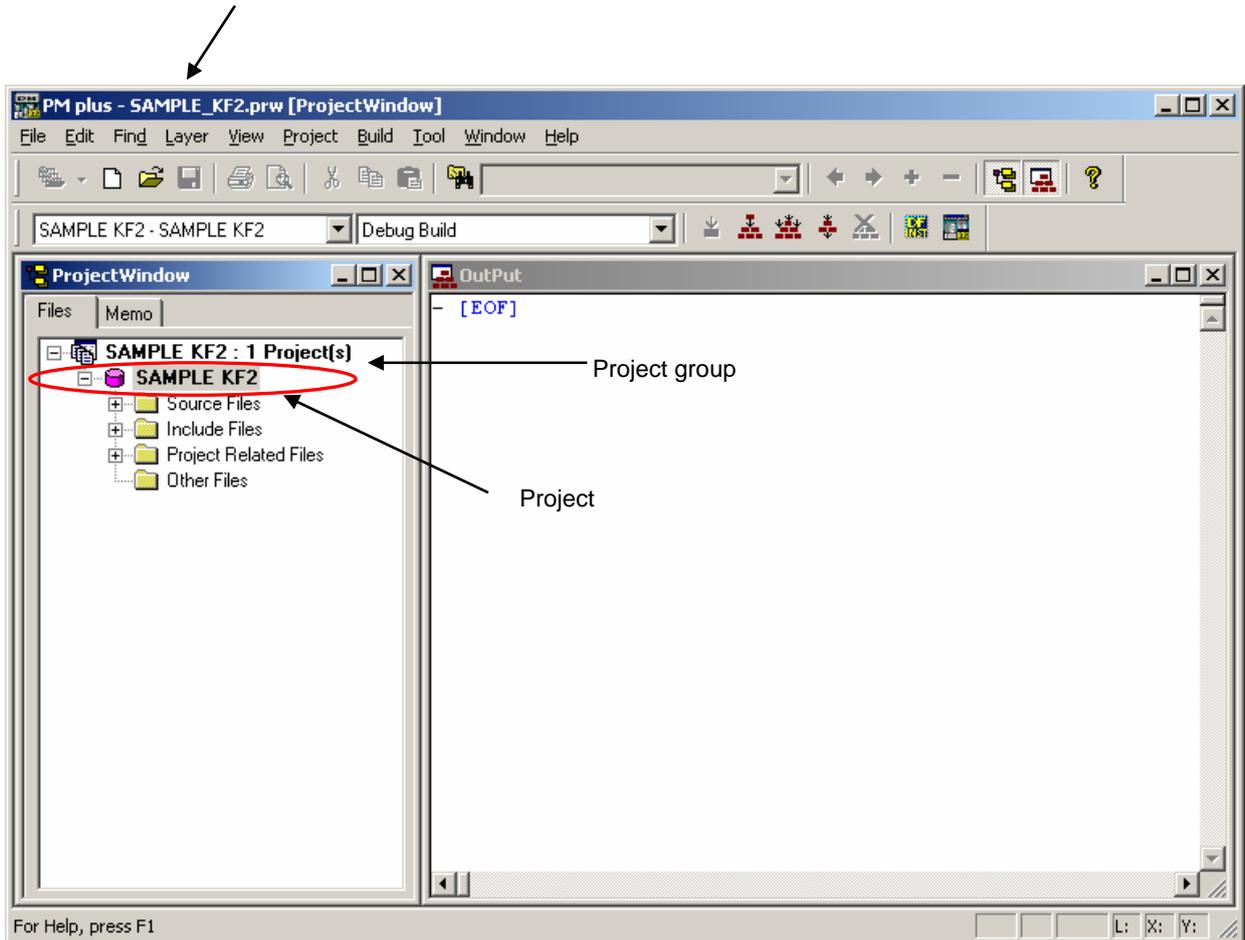
Open the folder in which the sample program is located.



Click “SAMPLE\_KF2.prw” and then press the **Open** button.



Workspace file name :SAMPLE\_KF2.prw



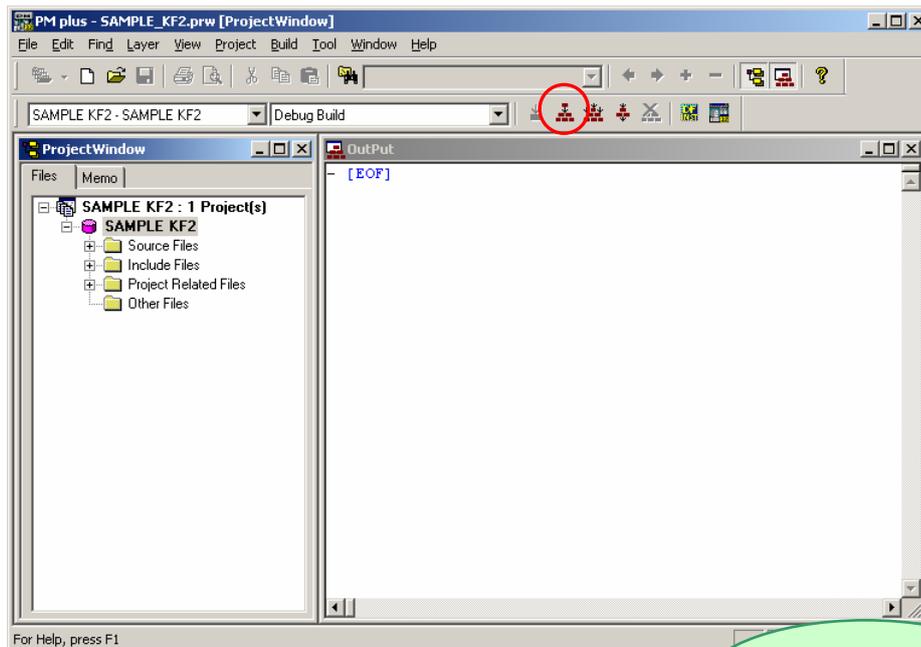
Only one “SAMPLE\_KF2” project is included in the “SAMPLE\_KF2. prw” workspace file.

The operations described below are done in relation to this “SAMPLE\_KF2” project.

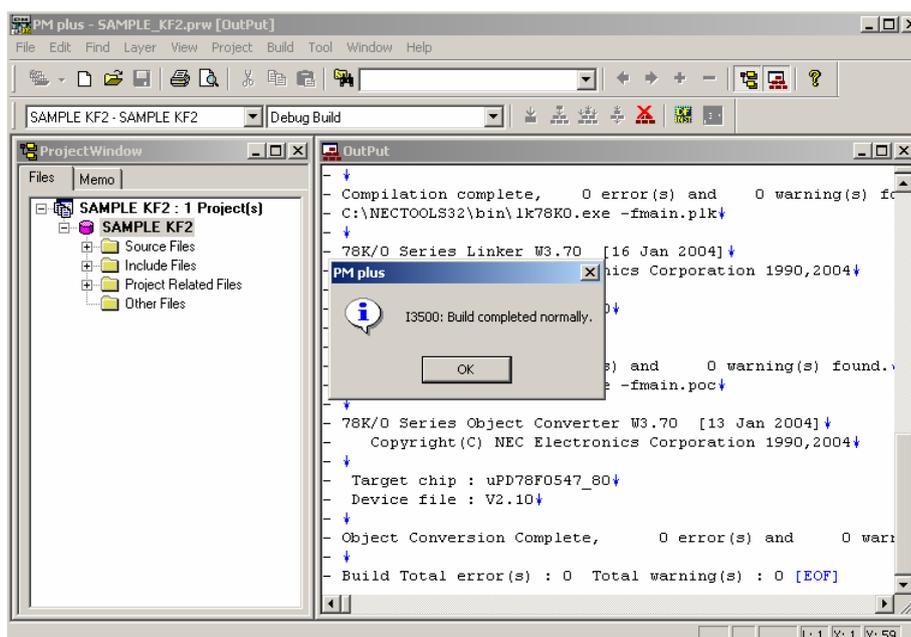
# Creating execution format

Let's create the project's format. This action is called [build](#).

Enter press PM plus's Build button  , or select [Build(B)]→[Build(B)] from the menu.



Build processing is executed



Build processing has been normally completed.

**What is build ?**

Build is a function that creates an executable file, etc., from a source file registered to a project.

PM plus automatically performs compiling, linking, and other processing actions.

On the second and subsequent build, PM plus also automatically detects files that have been updated from the previous build processing, and compiles and assembles only the relevant files, thereby reducing the time required for build processing.

**What is rebuild ?**

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

When settings, such as compiler options, have been changed, not build, but rebuild, must be Selected.

# Verification operating

The [integrated debugger](#) is provided as user application execution environments (debuggers) in the TK-78K0.

In this document, the integrated debugger (ID78K0-TK) is started and the operation is checked using this debugger.

## **What is the integrated debugger ID78K0-TK ?**

The integrated debugger is Windows-based software used for debugging that performs communication with on chip debug function interface in a development environment where the host machine and the target system are connected.

Debugging can be performed either at the C source level or at the assembler level.

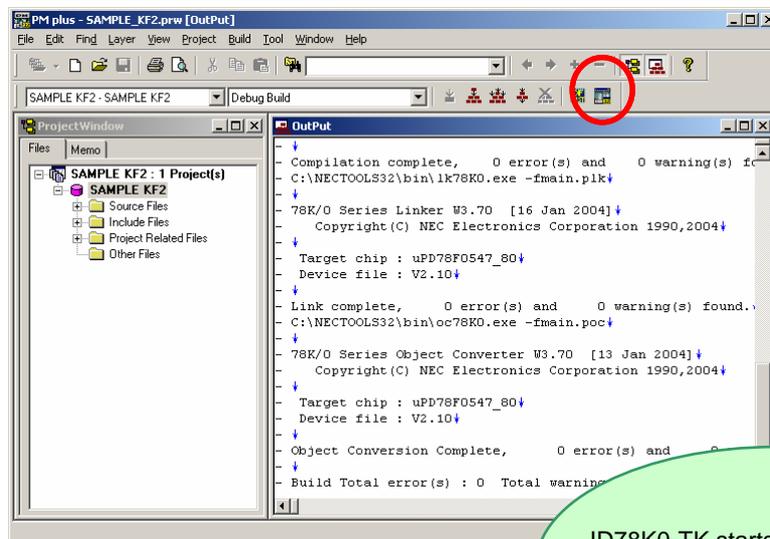
It is possible in real time to execute, and to verify it by using on the chip debug function.

# Starting integrated debugger (ID78K0-TK)

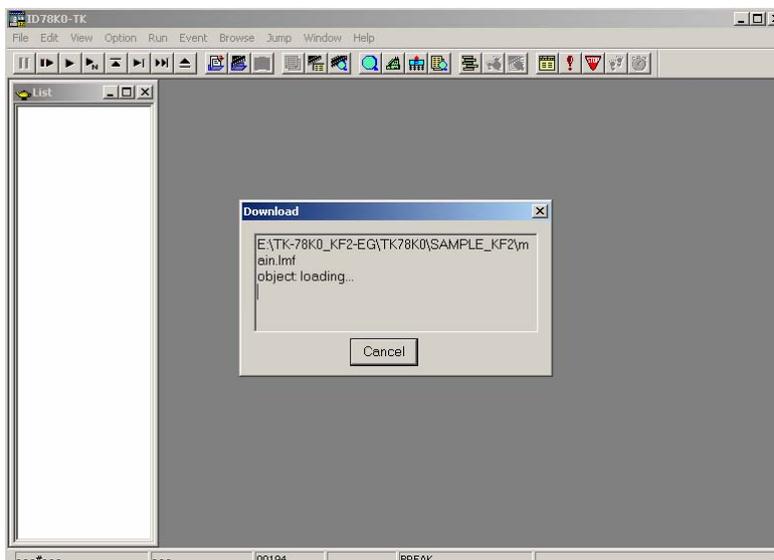
To use ID78K0-TK, it is necessary to connect TK-78K0/KF2's CPU board to the host machine with an USB cable, and power it on. After checking that the CPU board is powered, start ID78K0-TK.

Select the Debug button  , of PM plus, or in the menu, select [Build(B)]→[Debug(D)].

If the debug button is not displayed, in the menu, select [Tool]→[Debugger Setting...] “78K0 Integrated Debugger (ID78K0-TK for TK-78K0)”



ID78K0-TK starts, and the compiled program is downloaded to the flash memory with built-in CPU.



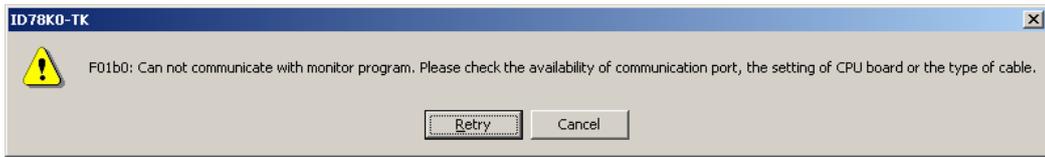


When download ends normally, the program source is displayed.

The screenshot shows a software development environment window titled "ID78K0-TK: SAMPLE\_KF2.prj". The main window displays the source code for "main.c". The code is as follows:

```
21  
22 h_initial();  
23 EI();  
24  
25 while (<1>{  
26  
27     g_start = 0;  
28     IP0H.6 = 0;  
29     MR0H.6 = 1;  
30  
31     IP0L.2 = 0;  
32     MR0L.2 = 0;  
33     PR0L.2 = 1;  
34  
35     while< g_start == 0 >{  
36  
37  
38         P5.7 = 1;  
39         MR1L.5 = 1;  
40         MR0L.2 = 1;  
41         MR0H.6 = 0;  
42  
43         while< g_start == 1 >{  
44             if < !P12.0 >{  
45                 g_start = 2;  
46             }  
47         }  
48  
49         while < g_start == 2 > {  
50             if < g_led_s >= 40 >{  
51                 g_start = 0;  
52                 g_led_s = 0;  
53                 P5.7 = 0;  
54             }  
55         }  
56     }  
57 }  
58
```

Normally, if communication with the CPU, the following error messages are output.



If these errors occur, check the following items.

Verification items:

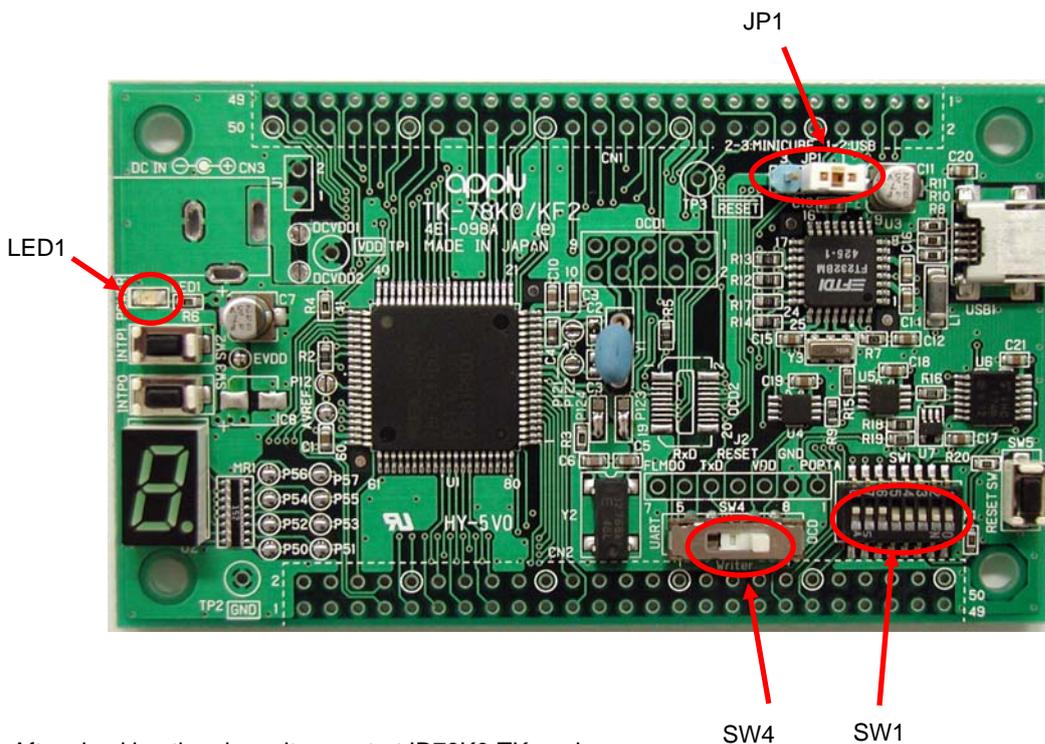
1. Please confirm the switch on the board is set as follows.

SW1

Bit 1	ON
Bit 2	ON
Bit 3	ON
Bit 4	ON
Bit 5	ON

SW4  OCD side

2. Is the departure pendulum mounted on Y1 on the board in the socket?
3. Is JP1 made 1-2 short and does power indication light LED1 light?
4. Is the COM port number set by the COM port and "Portconfig for ID78K0-TK" allocated in TK-78K0 corresponding?



After checking the above items, start ID78K0-TK again.

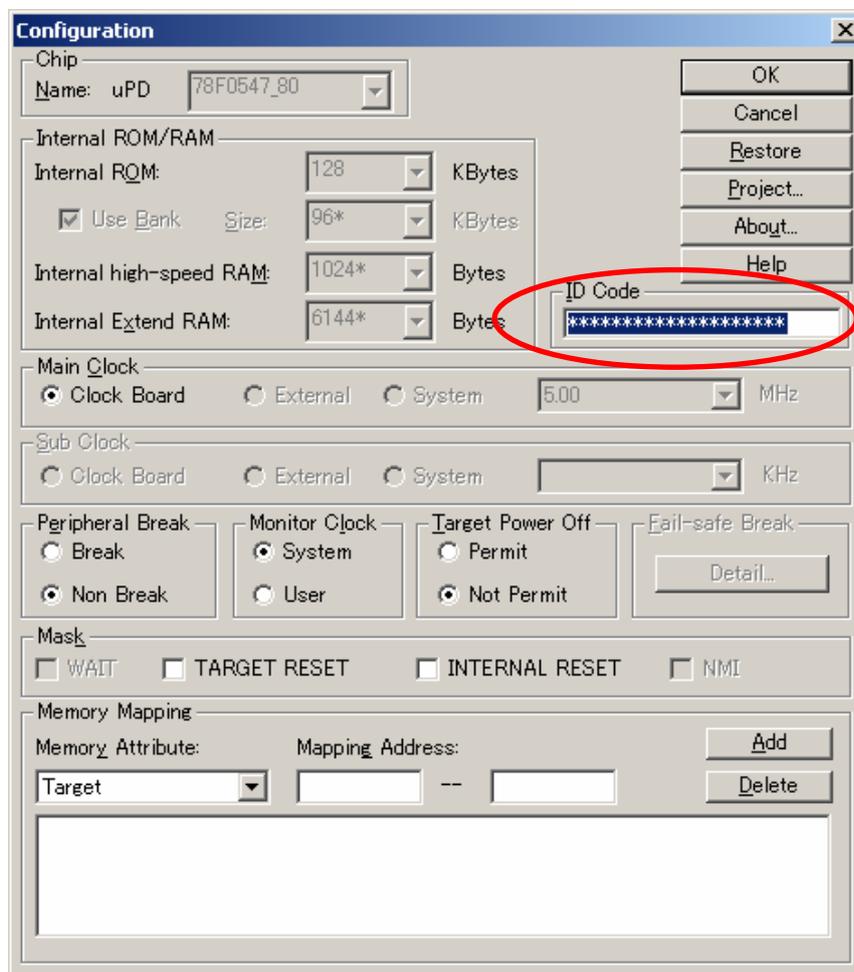
When the ID code is wrong, the following messages are displayed.

Also, this message can be displayed at first start up.

In that case, Please input "FFFFFFFFFFFFFFFF" [20 of F ] in "ID code".



The following "Configuration" screen is displayed when  is pushed and input a correct ID code, please.



When the ID code is forgotten, it is necessary to delete all flash memories with built-in CPU.

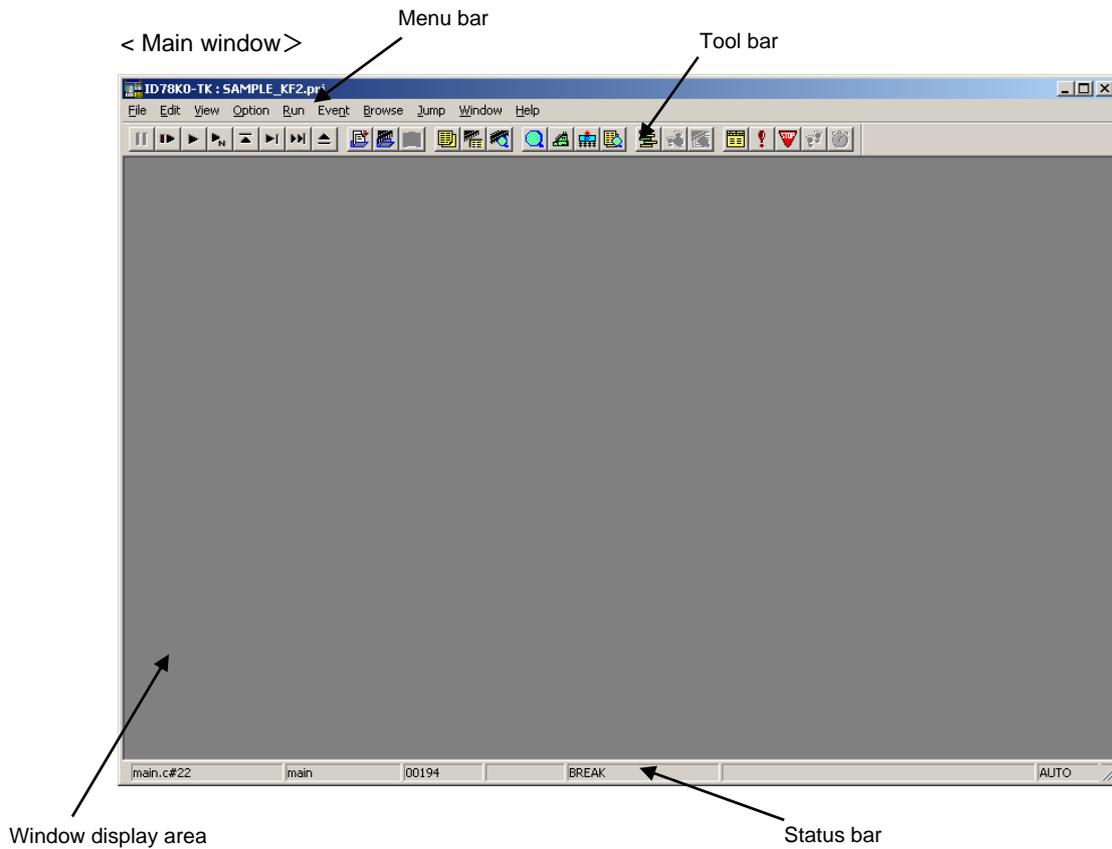
Please delete the flash memory by installing "PG-FPL3" from CD of the attachment.

Please refer to Chapter 3 for the usage of "PG-FPL3".

# Introduction of integrated debugger (ID78K0-TK)

ID78K0-TK displayed the CPU core's internal status and controls monitor program execution in the main window.

The initial screen of ID78K0-TK is as follows.



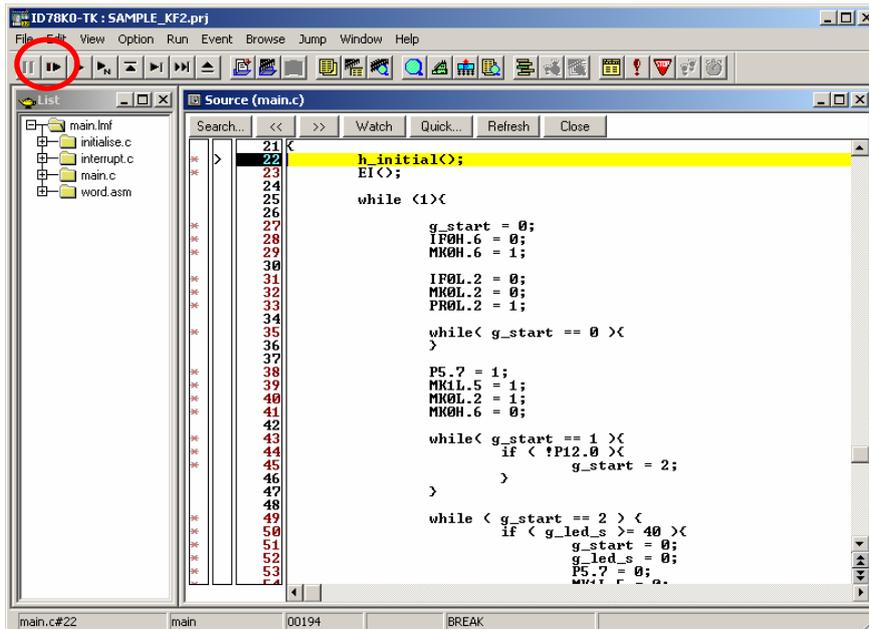
➡ For details about the various areas, the menu bar and the tool bar, refer to the "ID78K0-TK Integrated Debugger Operating Manual"

# Executing program

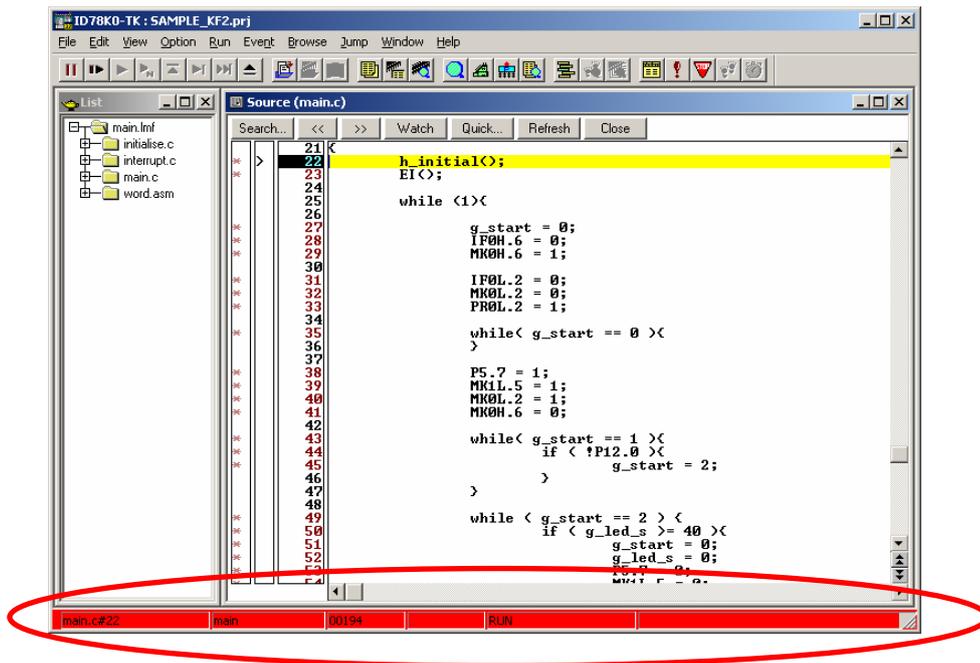
Next, execute the program

Select the ID78K0-TK's restart button  , or select [Run(R)]→[Restart(R)]

The LED count up program is executed.

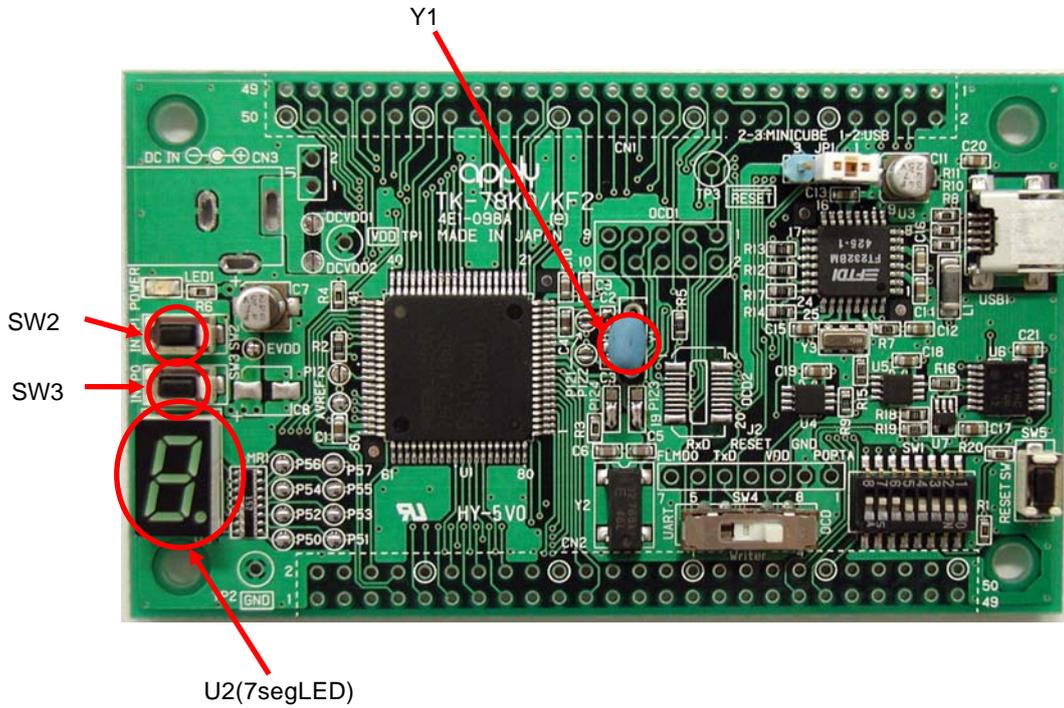


Execute the program



The status bar changes to red during program execution.

Next, when you pushed SW2 on the CPU board of TK-78K0/KF2, 7seg LED starting count up.  
When you pushed SW3, the count up interval becomes longer and confirm the stop.



This program uses a resonator on Y1 on the board in the socket for CPU clock, counter clock.  
Therefore please mount the resonator when execute the program.

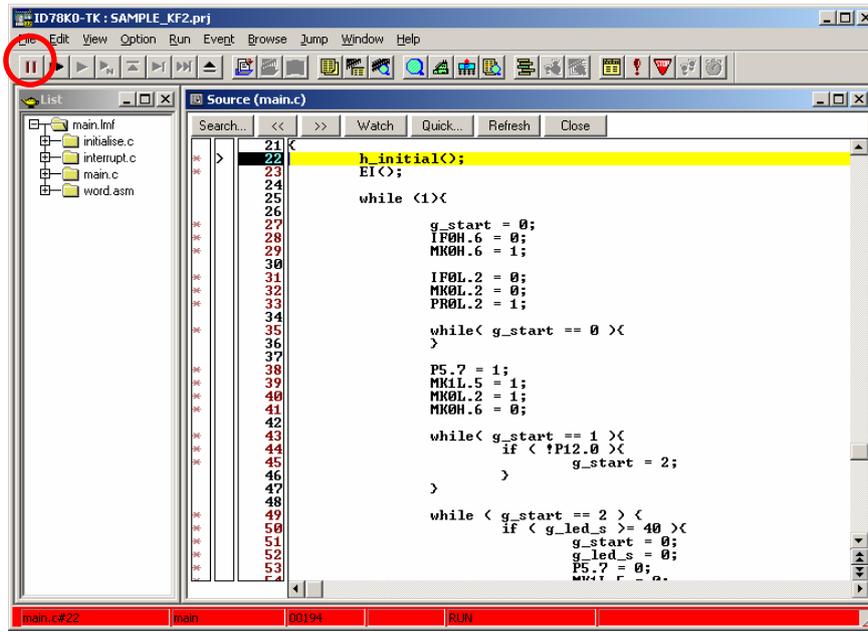


The above-described procedure checks that the 7seg LED count up program functions normally.

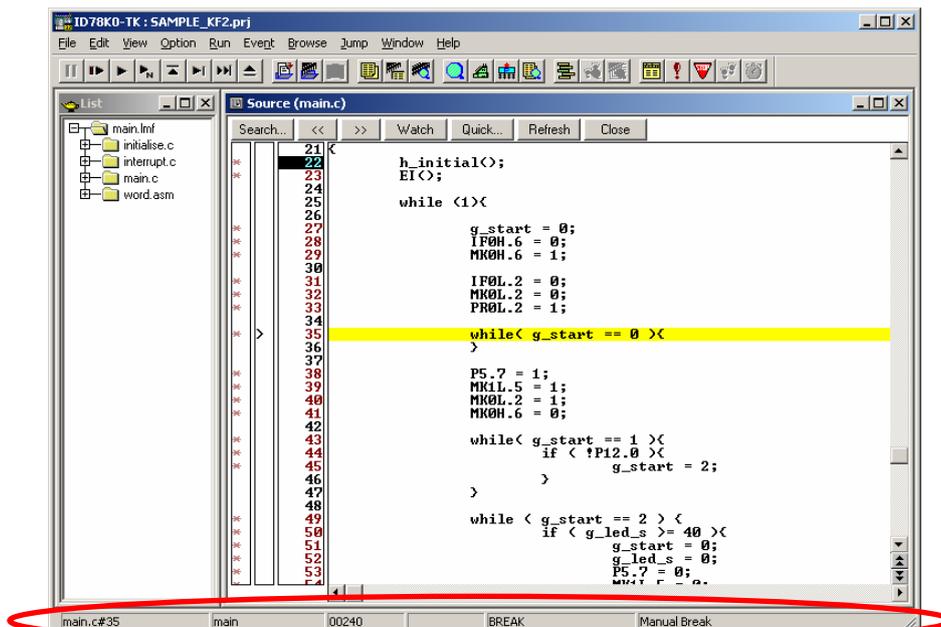
# Stopping program

Stop the program.

Press the ID78K0-TK's stop button , or in the menu, select [R]un→[S]top.



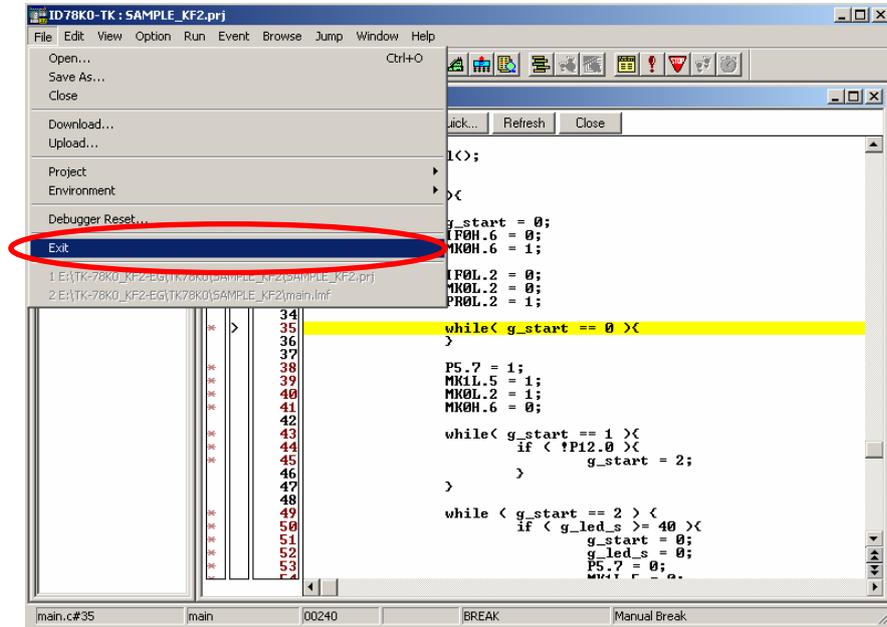
Stop the program



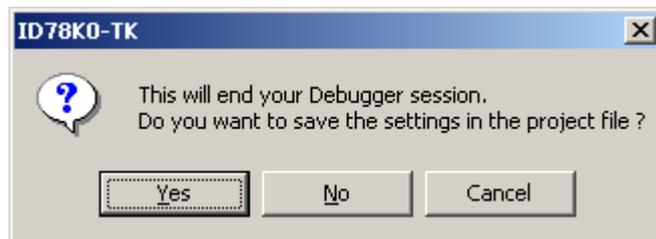
When the program stops, the status bar changes back to its original color.

# Exiting integrated debugger (ID78K0-TK)

In the ID78K0-TK menu, select [File]→[Exit].



The Exit confirmation dialog box is displayed

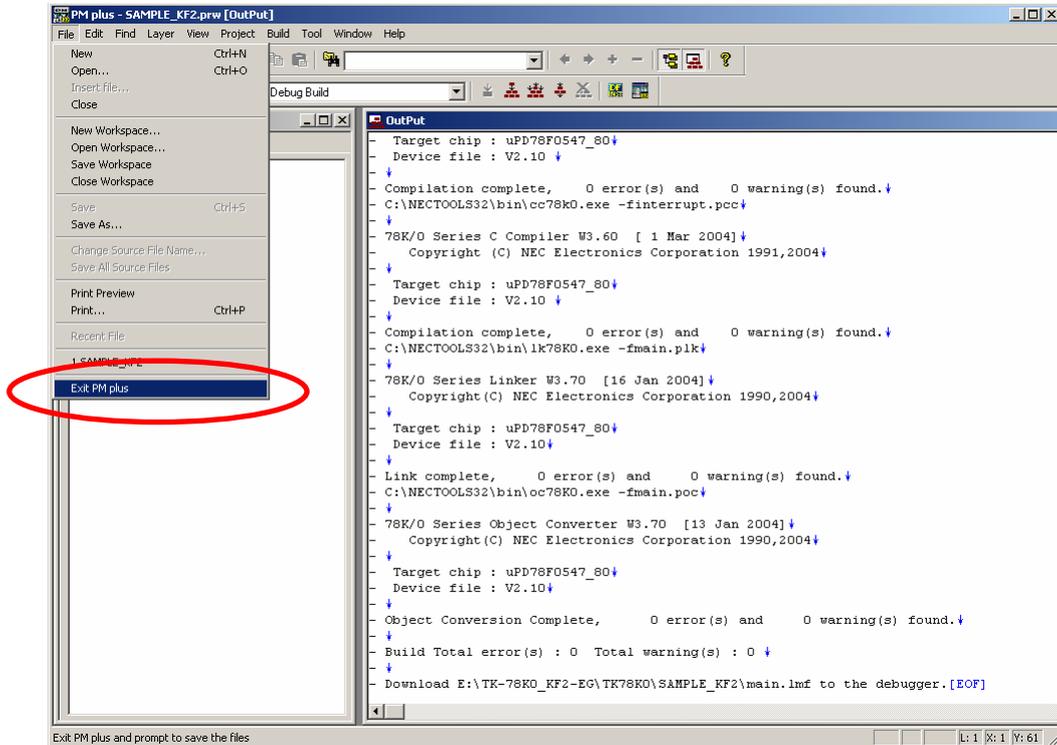


ID78K0-TK ends after preserving a present environment if **Yes** is pushed.

ID78K0-TK ends without preserving a present environment if **No** are pushed.

# Closing PM plus

In the PM plus menu, select [File]→[Exit PM plus].



PM plus ends.

# Chapter3 Other

This chapter describes how to create a new PM plus workspace (project) required for debugging using the integrated debugger (ID78K0-TK), registering additional source file, and how to use the PG-FPL3.

[Creating a new PM plus workspace \(project\)](#)

[Registering additional source file](#)

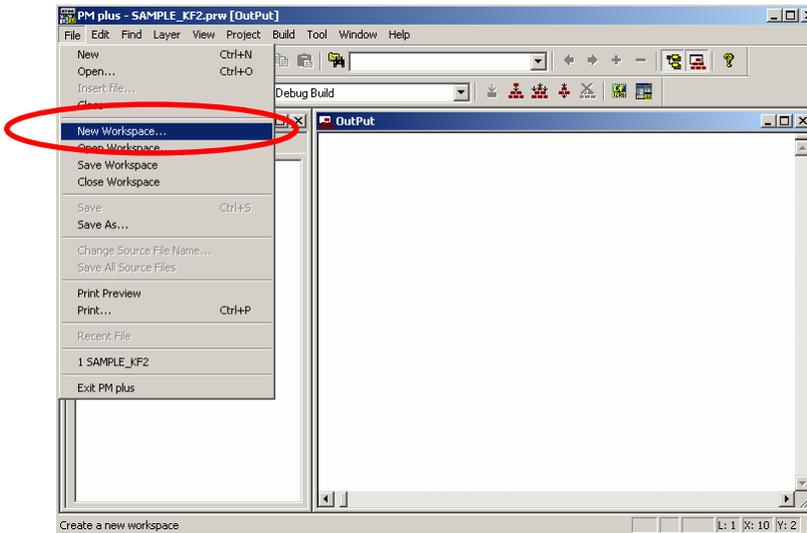
[How to use PG-FPL3](#)

# Creating a new PM plus workspace (project)

Create a new workspace and project.

With PM plus, a project that includes the basic items required for build can easily be created by inputting each of various items step by step.

In the PM plus menu, select [File]→[New Workspace...].



The dialog box for creating New workspace is displayed

## <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:

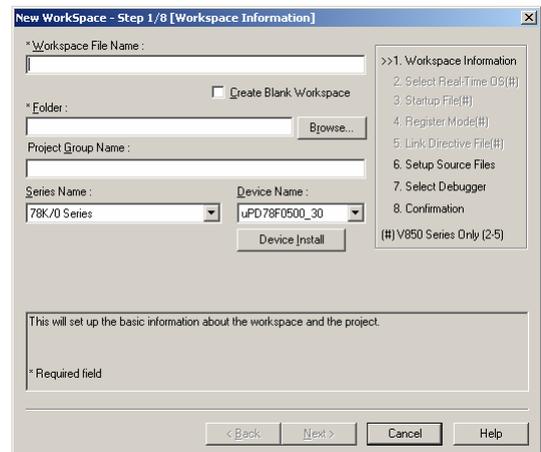
- > Specify this item if wishing to manage multiple projects together in function units.  
If nothing is specified, this item is the same as the workspace file name.

### Series Name:

- > Specify the series name of the device 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  
→ Don't input this item.

Series Name  
→ 78K/0 Series

Device Name  
→ uPD78F0547\_80

New Workspace - Step 1/8 [Workspace Information]

\* Workspace File Name : test

\* Folder : C:\TK78K0\test

Project Group Name :

Series Name : 78K/0 Series

Device Name : uPD78F0547\_80

>>1. Workspace Information  
2. Select Real-Time OS(##)  
3. Startup File(##)  
4. Register Mode(##)  
5. Link Directive File(##)  
6. Setup Source Files  
7. Select Debugger  
8. Confirmation  
(#) V850 Series Only (2-5)

This will set up the basic information about the workspace and the project.

\* Required field

< Back Next > Cancel Help

Push the Next > button.



PM plus

Q2011: C:\TK78K0\test

This folder cannot be found. Do you wish to create it?

Yes No

Push the Yes button.



New Workspace - Step 6/8 [Setup Source Files]

Please setup Source Files.

Source File Name :

Add... Remove Remove All Up Down

1. Workspace Information  
2. Select Real-Time OS(##)  
3. Startup File(##)  
4. Register Mode(##)  
5. Link Directive File(##)  
>>6. Setup Source Files  
7. Select Debugger  
8. Confirmation  
(#) V850 Series Only (2-5)

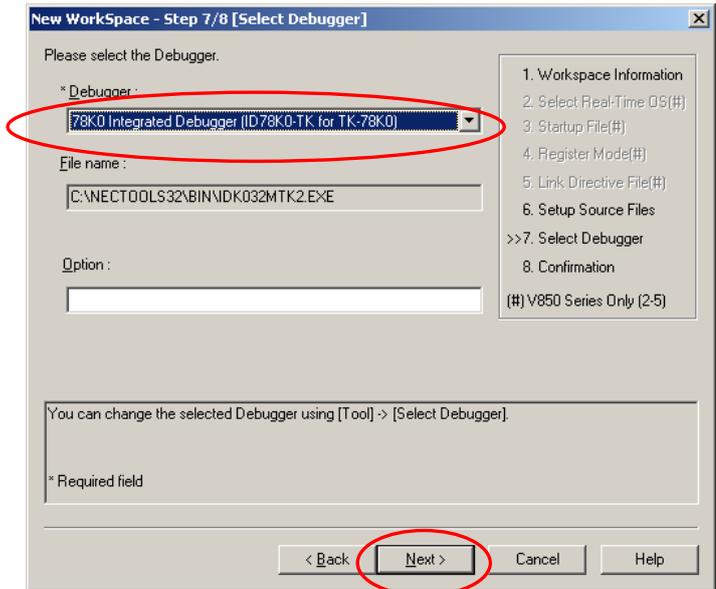
You can also add source files by specifying the list file or the folder.  
You don't need to specify all the files here, and you can setup source files using [Project]->[Project Settings] later.

< Back Next > Cancel Help

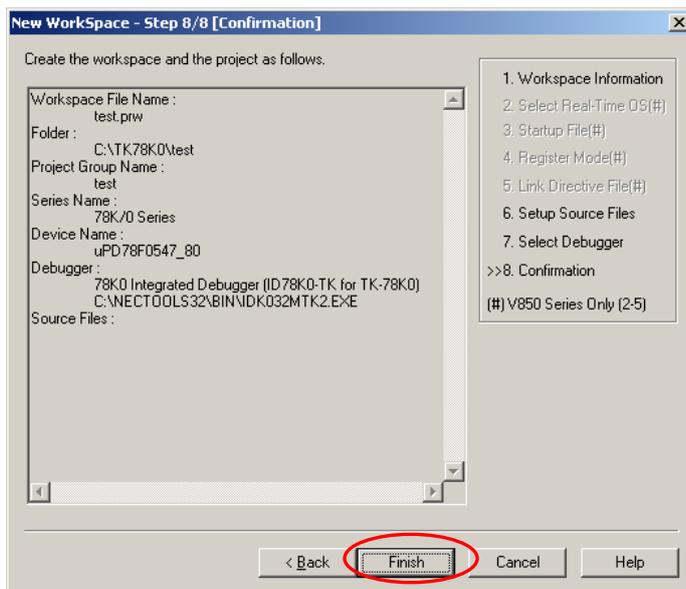
Push the Next > button.



Please select 78K0 Integrated Debugger (ID78K0-TK for TK-78K0)



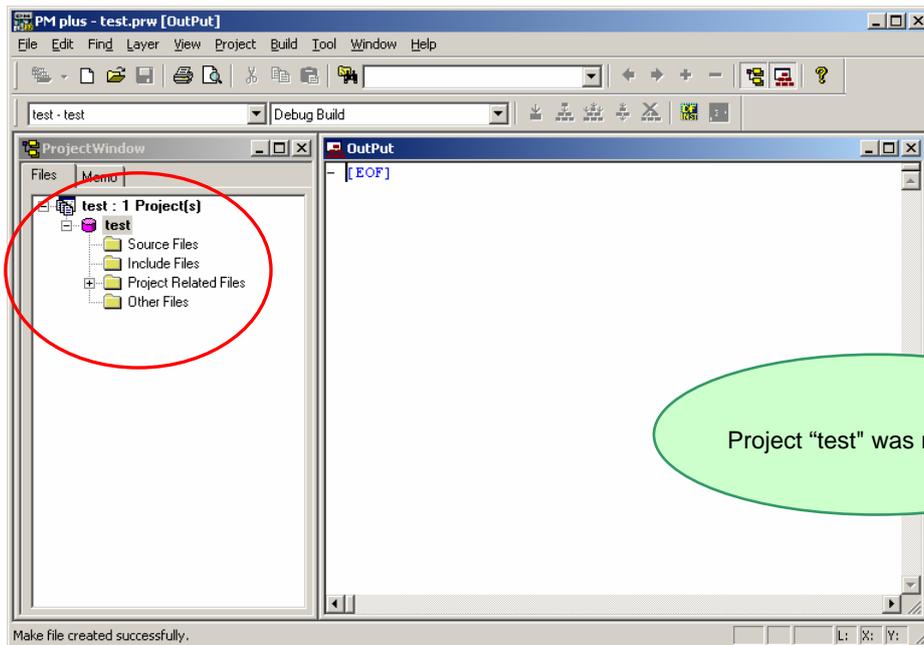
Push the **Next >** button.



Push the **Finish** button.



Check the project information setting contents.



This completes workspace and project creation.

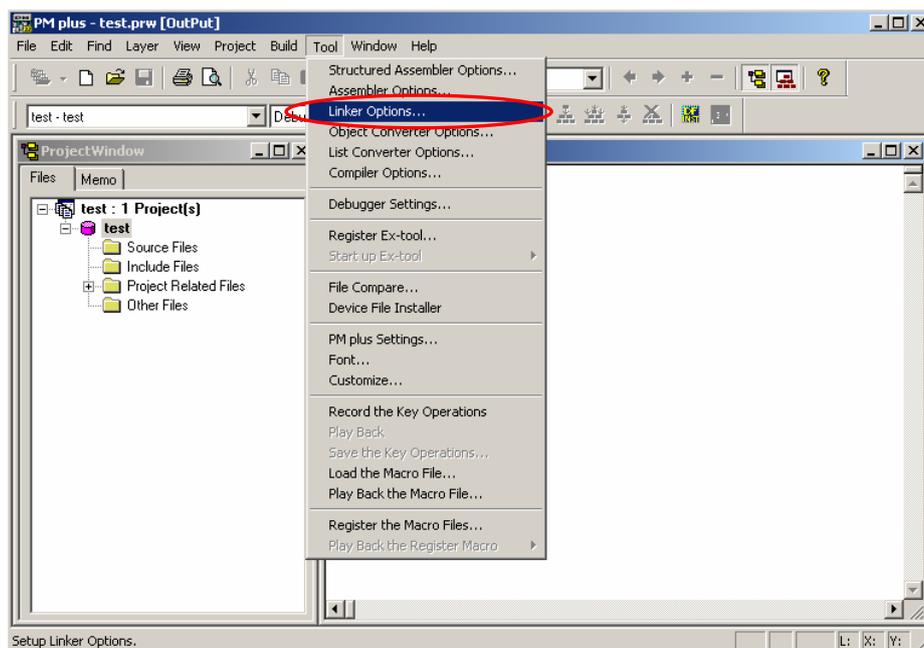
Additional source files can be registered at any time thereafter.

➡ For details, refer to ["Registering additional source file"](#)

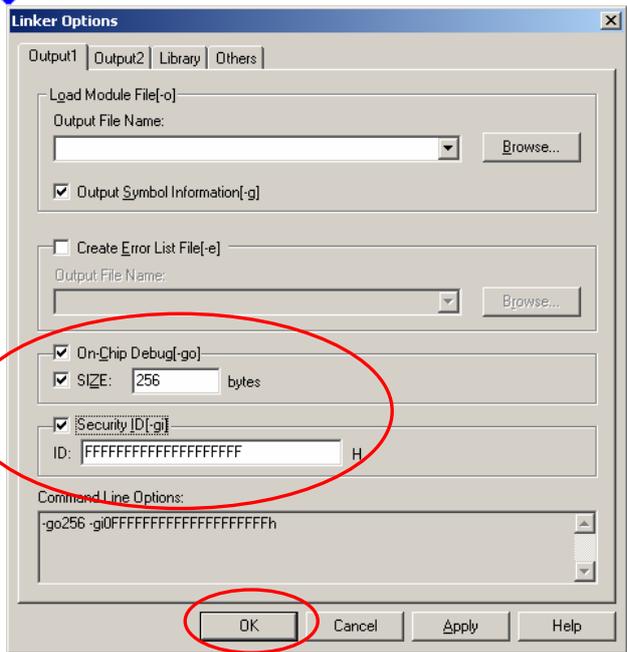


Next, on chip debug function can be used.

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



Checks are put in "On-Chip Debug" and "SIZE". Then input a default value of "256" into the "SIZE". A check is put in "Security ID". The actualities of "FFFFFFFFFFFFFFFF" are taken if there is no problem in the value of ID in security.



Push the **OK** button.

Next, please add the following "word.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 "a notice point in use" of ID78K0-QB for the details of the security ID.

"word.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.

```

ORG 80H          ; setting of option byte
DB 0,0,0,0      ;

ORG 84H
DB 3            ; The authentication of the security ID is indispensable.

end

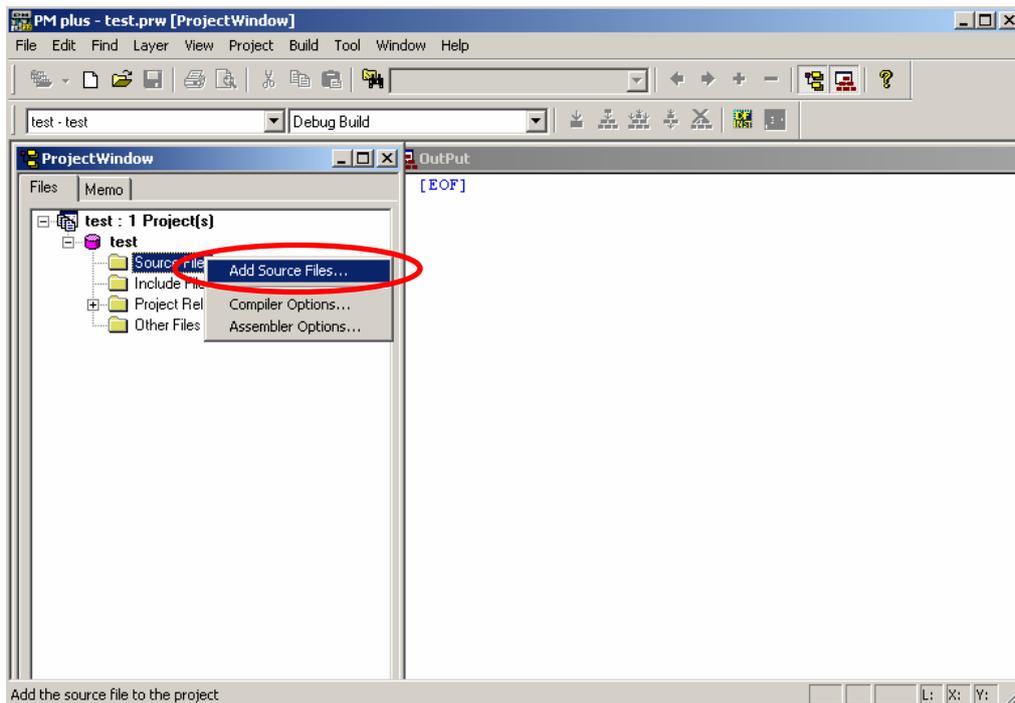
```

# Registering additional source file

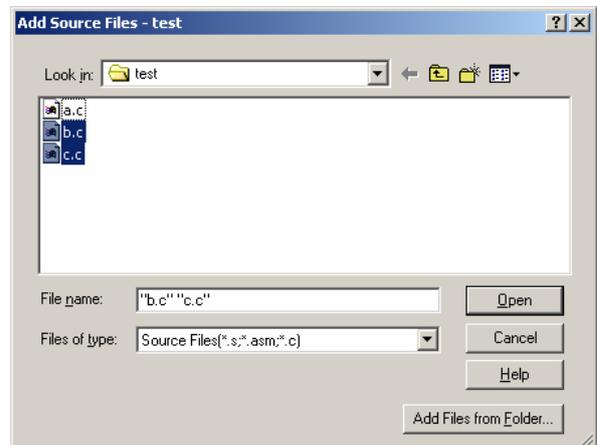
The method for registering additional source files to a project is described below.

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 plus, and select [Add Source Files...] displayed in the right-click menu



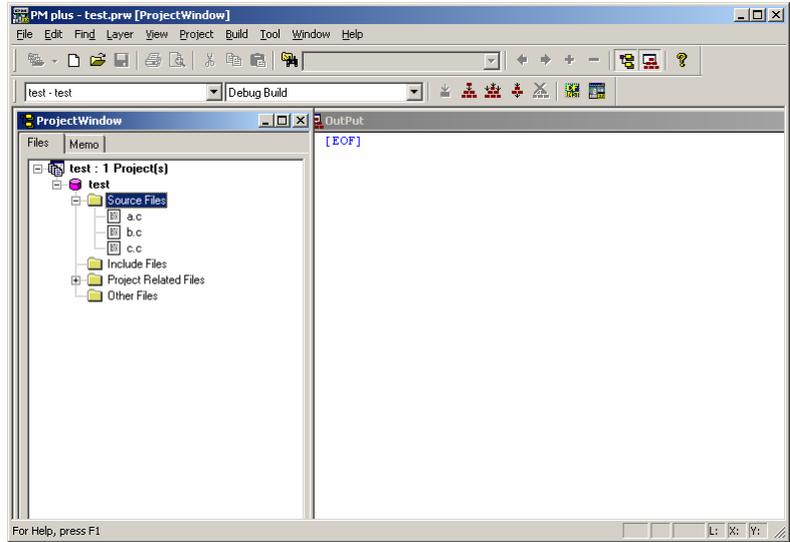
Select source files "b.c" and "c.c" and then press the **Open** button.



Multiple source files can be selected by clicking the desired source files where pressing **Ctrl** key.



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



# How to use PG-FPL3

It explains the method of deleting the flash memory with built-in 78K0/KF2 by using PG-FPL3.

When security ID etc. are forgotten and ID78K0-TK cannot be connected, PG-FPL3 is used.

PG-FPL3 cannot be installed from an integrated installer. Please execute and install

¥FPL3¥FPL3\_V110¥Setup.exe

from the Explorer etc. Please refer to "PG-FPL3 user's manual" for details of the installation method.

However, it is not necessary to install hardware (driver). As for TK-78K0, the hardware of PG-FPL3 is built in.

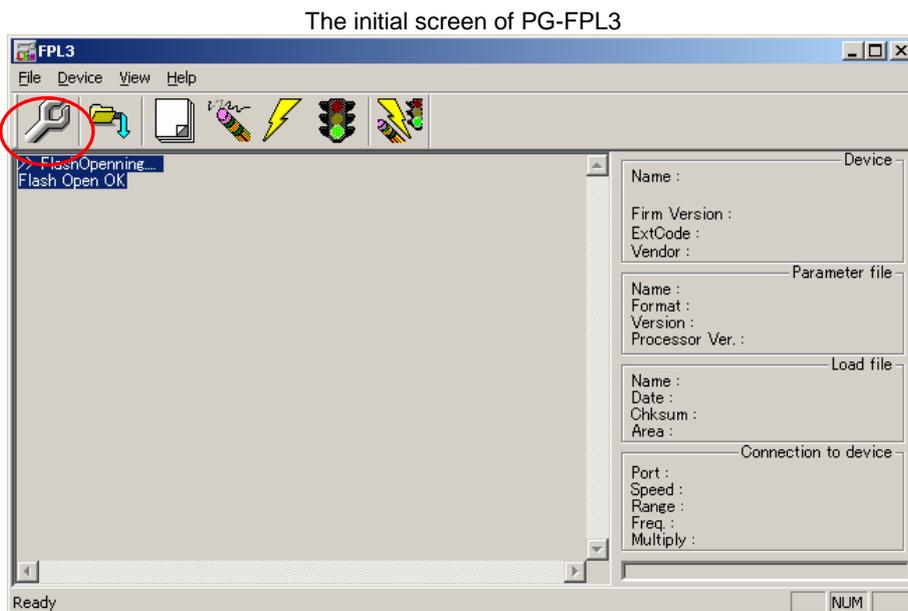
The switch of TK-78K0 is set as follows, and it connects it with the PC.

SW1

Bit 1	ON
Bit 2	ON
Bit 3	OFF
Bit 4	OFF
Bit 5	OFF

SW2     UART side

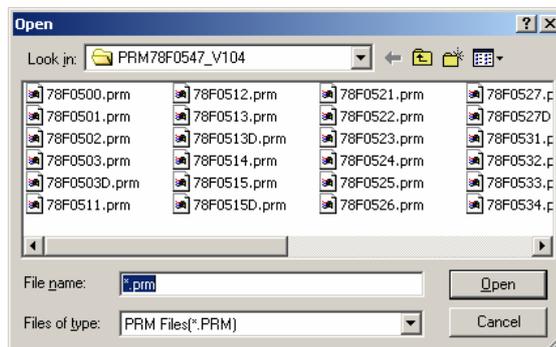
Please select → [Programs] → [NEC Tools32] → [PG-FPL3] of the Windows start menu.



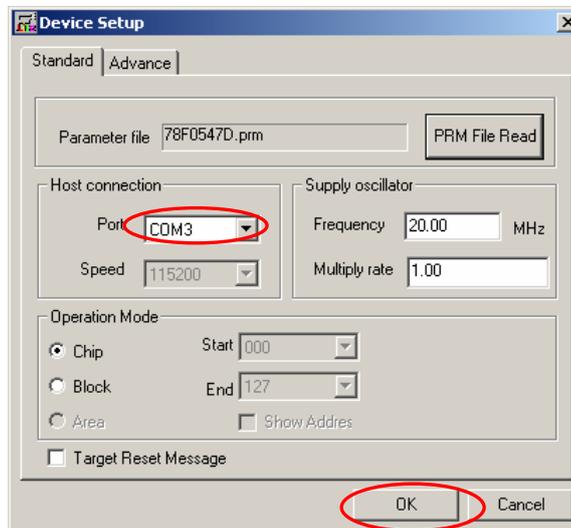
Push the 「Setup」 button.



Push the 「PRM File Read」 button.



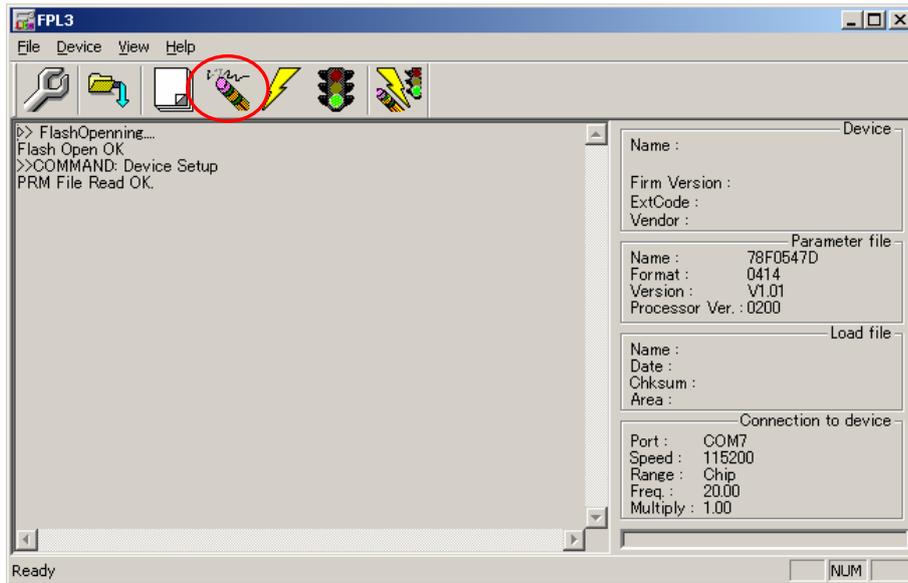
/PRM/PRM78F0547\_V105/78F0547D.prm of attachment CD is selected.



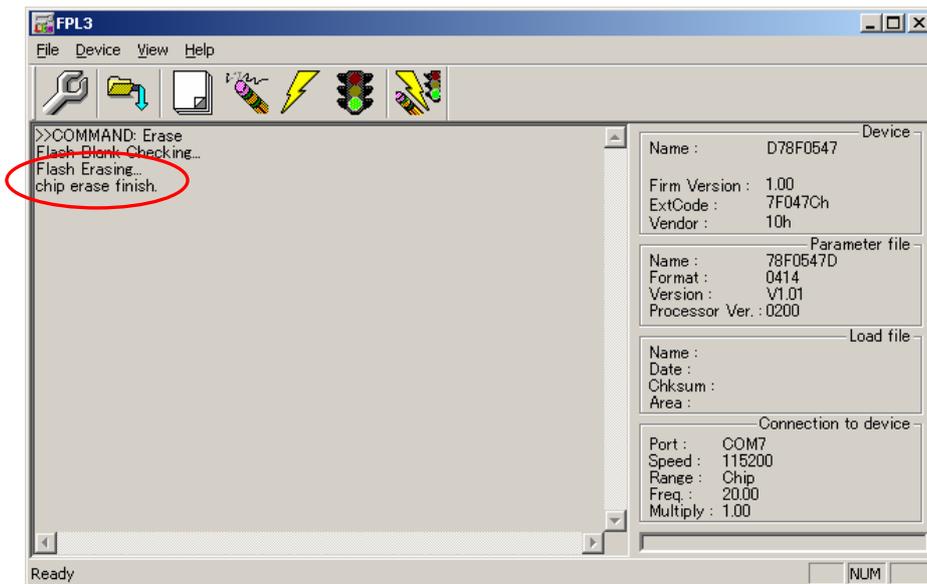
"Port" selects the COM port number where TK-78K0 is allocated.

※Only the COM port number that the personal computer has is displayed in this pull-down menu.

Push the "OK".



"Erase" The deletion of the flash memory begins when the button is pushed.



If "chip erase finish" is displayed, the deletion of the flash memory is completed.

Please end PG-FPL3.

It returns based on the set switch and completion.

SW1

Bit 1	ON
Bit 2	ON
Bit 3	ON
Bit 4	ON
Bit 5	ON

SW2  OCD side