SKP16C26
Tutorial 2
Creating A New Project Using HEW

Renesas Technology America Inc.

Overview

This tutorial describes the steps in creating new projects for the M16C using two different methods.

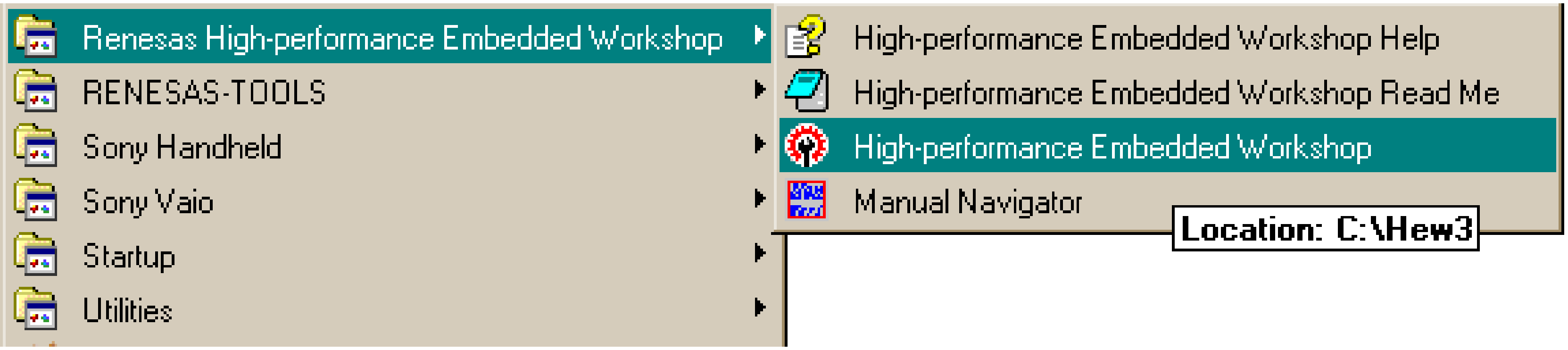
The first method uses the SKP project generator, that will create an empty project.
The second method will provide step by step instructions on creating a project with existing source files.

To get the most out of the SKP including the development tools, check out the references at the end of this tutorial.

Note: ***This tutorial assumes the user has done the following:***

1. Followed the ‘Quickstart Guide’
2. Installed the SKP files, examples, and software tools in the default directories.
3. Reviewed Tutorial 1.

Starting HEW

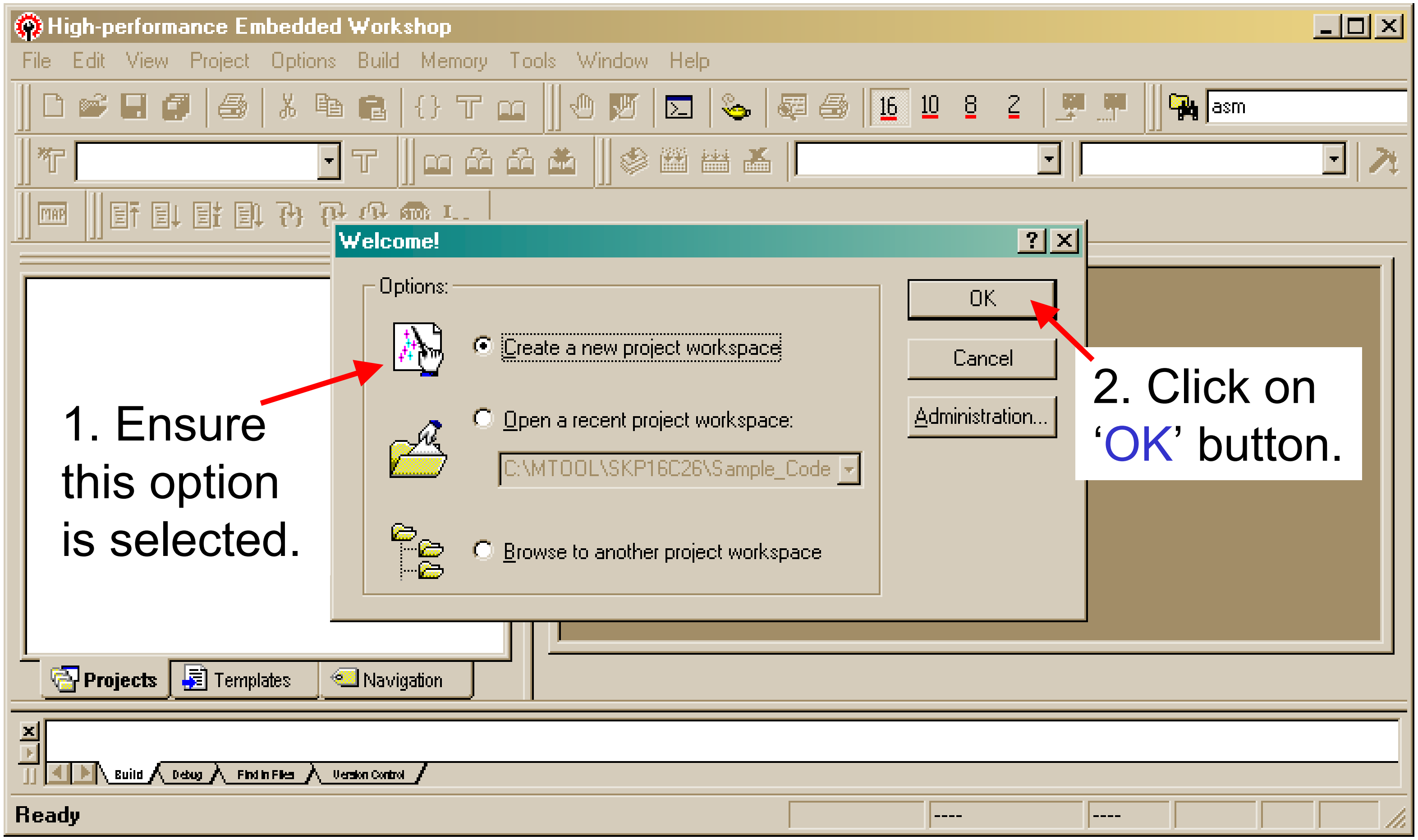


Renesas High-performance Embedded Workshop
RENESAS-TOOLS
Sony Handheld
Sony Vaio
Startup
Utilities
High-performance Embedded Workshop Help
High-performance Embedded Workshop Read Me
High-performance Embedded Workshop
Manual Navigator
Location: C:\Hew3

From the Windows Start menu, click on

Programs > Renesas High-performance Embedded Workshop>
High-performance Embedded Workshop

Creating a Project Workspace



Options:

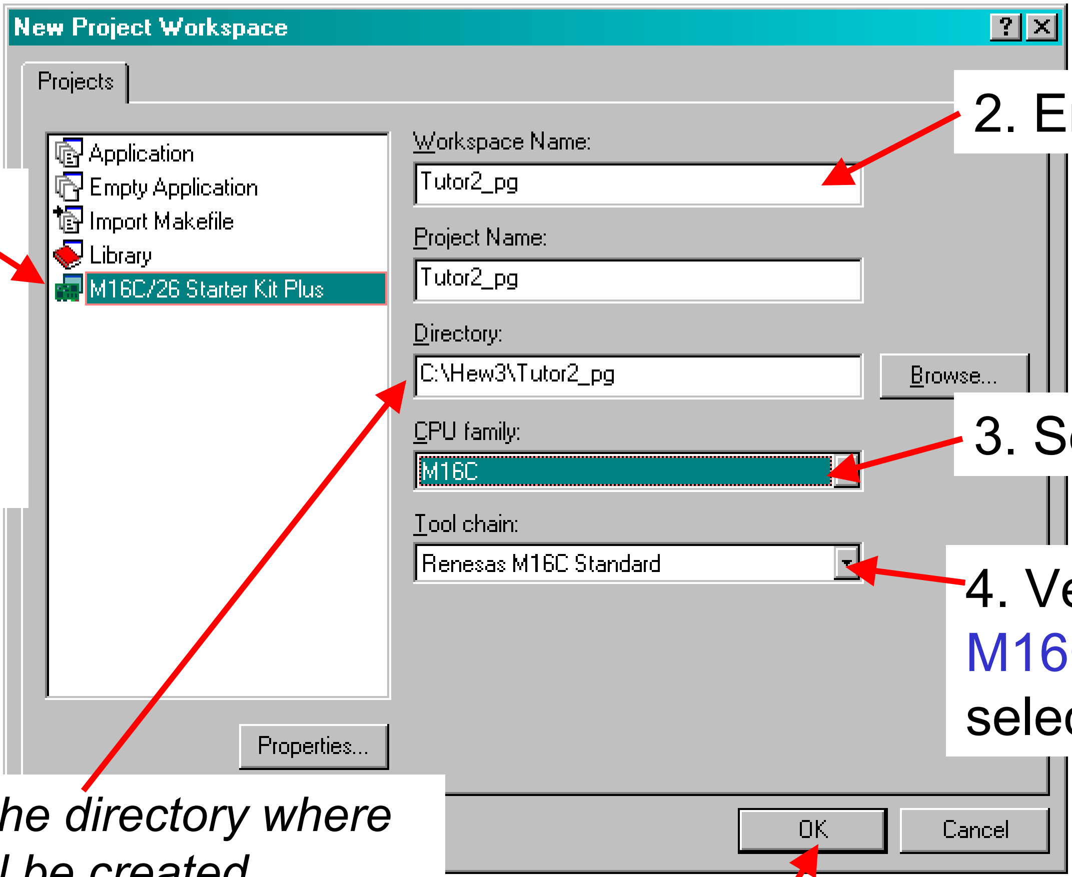
- ● Create a new project workspace
- ○ Open a recent project workspace: C:\MTOOL\SKP16C26\Sample_Code
- ○ Browse to another project workspace

1. Ensure this option is selected.

2. Click on 'OK' button.

When HEW starts up, you will be prompted to select a workspace. By default, 'Create a new project workspace' option is selected.

Creating a New Project with the SKP project generator (1/4)



Workspace Name: Tutor2_pg

Project Name: Tutor2_pg

Directory: C:\Hew3\Tutor2_pg

CPU family: M16C

Tool chain: Renesas M16C Standard

1. Select 'M16C/26 Starter Kit Plus' as the project type.

2. Enter 'Tutor2_pg'.

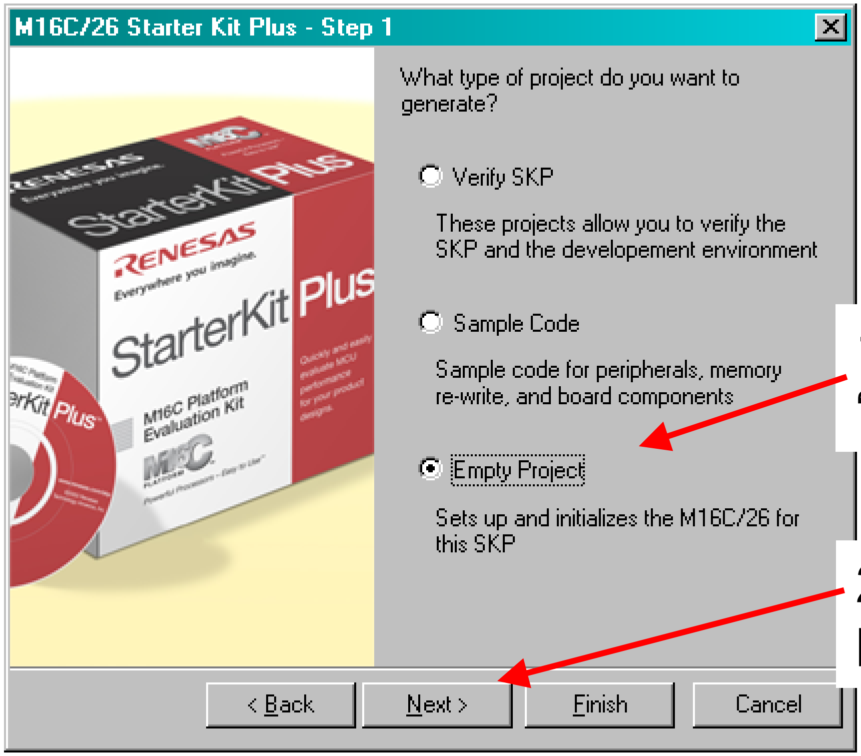
3. Select M16C.

4. Verify Renesas M16C Standard is selected.

5. Click on 'OK' button.

Note: This is the directory where the project will be created, 'C:\HEW3\Tutor2_pg'.

Creating a New Project with the SKP project generator (2/4)



M16C/26 Starter Kit Plus - Step 1

What type of project do you want to generate?

○ Verify SKP

These projects allow you to verify the SKP and the developement environment

○ Sample Code

Sample code for peripherals, memory re-write, and board components

● Empty Project

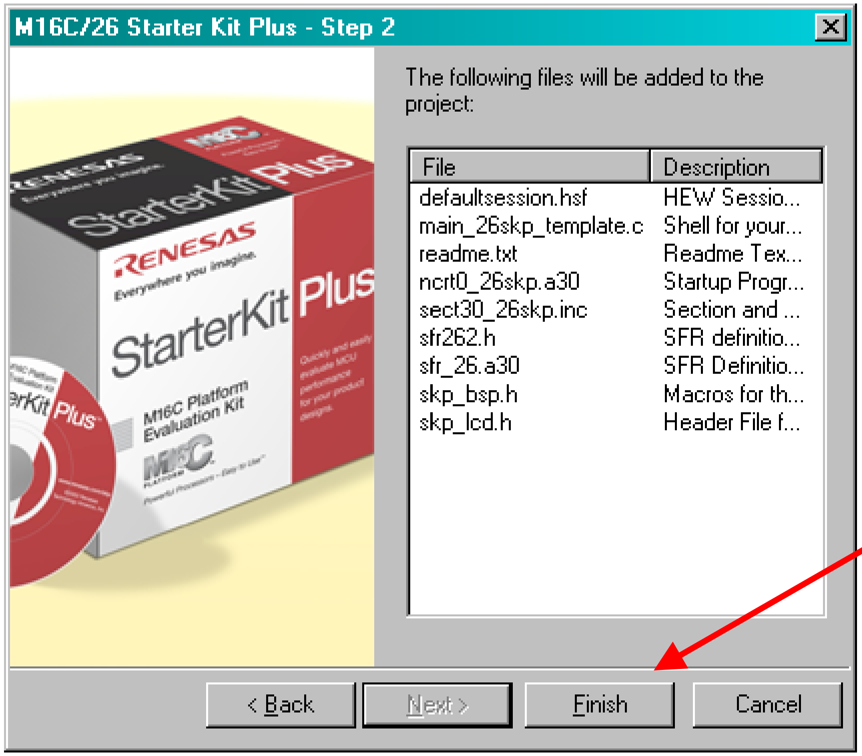
Sets up and initializes the M16C/26 for this SKP

1. Select 'Empty Project'.

2. Click on 'Next' button.

Note: The other project selections (Verify SKP and Sample Code) allow you to test the SKP and evaluate the MCU peripherals and features using pre-built sample code.

Creating a New Project with the SKP project generator (3/4)



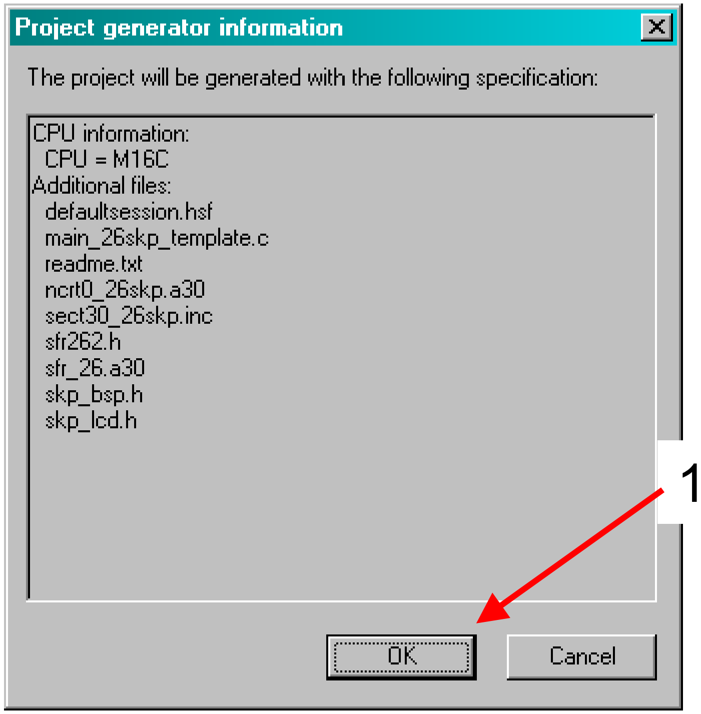
The following files will be added to the project:

File	Description
defaultsession.hsf	HEW Sessio...
main_26skp_template.c	Shell for your...
readme.txt	Readme Tex...
ncrt0_26skp.a30	Startup Progr...
sect30_26skp.inc	Section and ...
sfr262.h	SFR definitio...
sfr_26.a30	SFR Definitio...
skp_bsp.h	Macros for th...
skp_lcd.h	Header File f...

1. Click on ‘Finish’ button.

Note: Step 2 shows all the files that will be added to the new project and their description.

Creating a New Project with the SKP project generator (4/4)



Project generator information

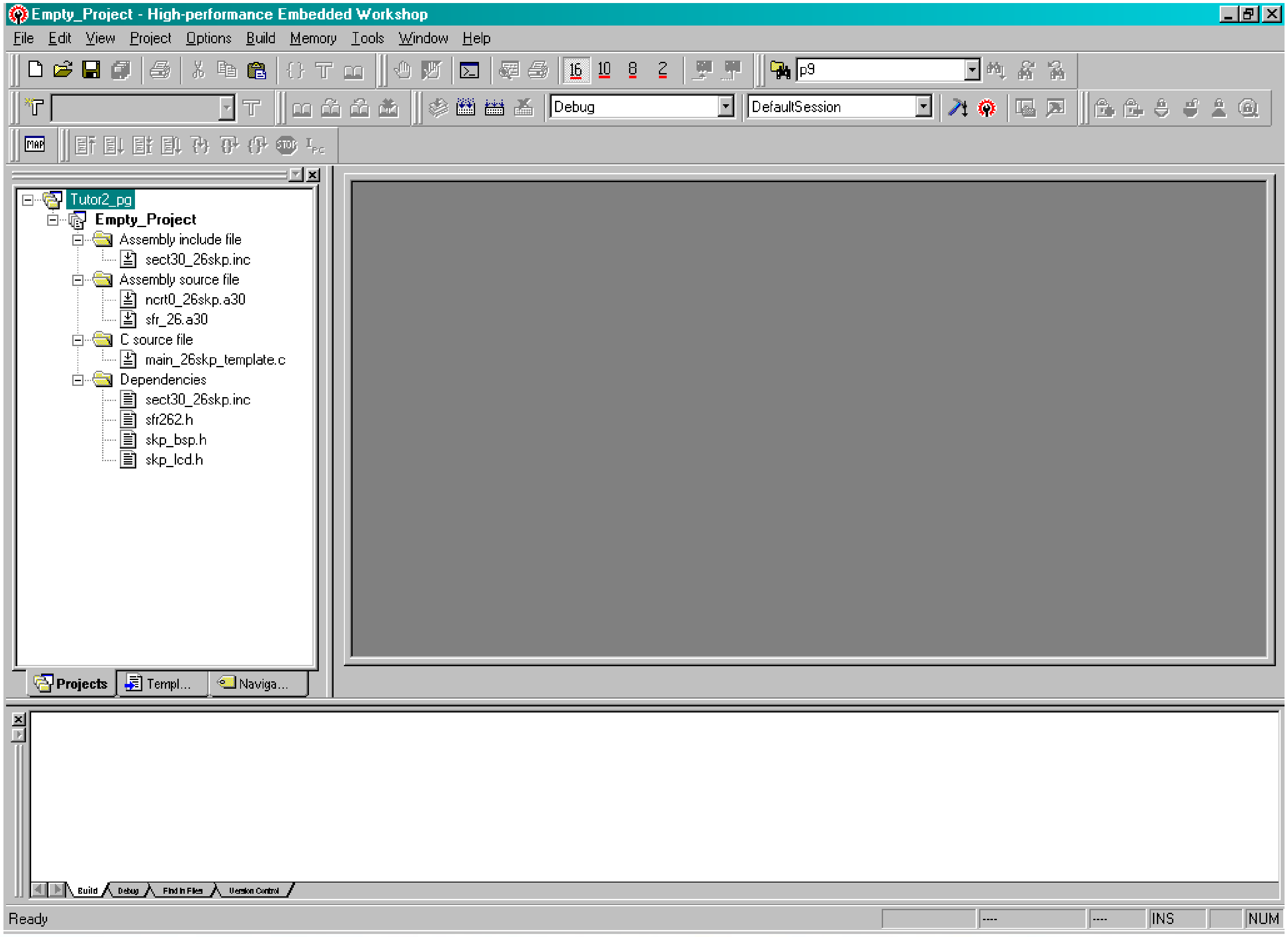
The project will be generated with the following specification:

```
CPU information:
  CPU = M16C
Additional files:
  defaultsession.hsf
  main_26skp_template.c
  readme.txt
  ncrt0_26skp.a30
  sect30_26skp.inc
  sfr262.h
  sfr_26.a30
  skp_bsp.h
  skp_lcd.h
```

1. Click on ‘OK’ button.

The project generator information is shown once the project has been created.

HEW after Creating a New Project using the SKP project generator



Empty_Project - High-performance Embedded Workshop

- Tutor2_pg
  - Empty_Project
    - Assembly include file
      - sect30_26skp.inc
    - Assembly source file
      - ncrt0_26skp.a30
      - sfr_26.a30
    - C source file
      - main_26skp_template.c
    - Dependencies
      - sect30_26skp.inc
      - sfr262.h
      - skp_bsp.h
      - skp_lcd.h

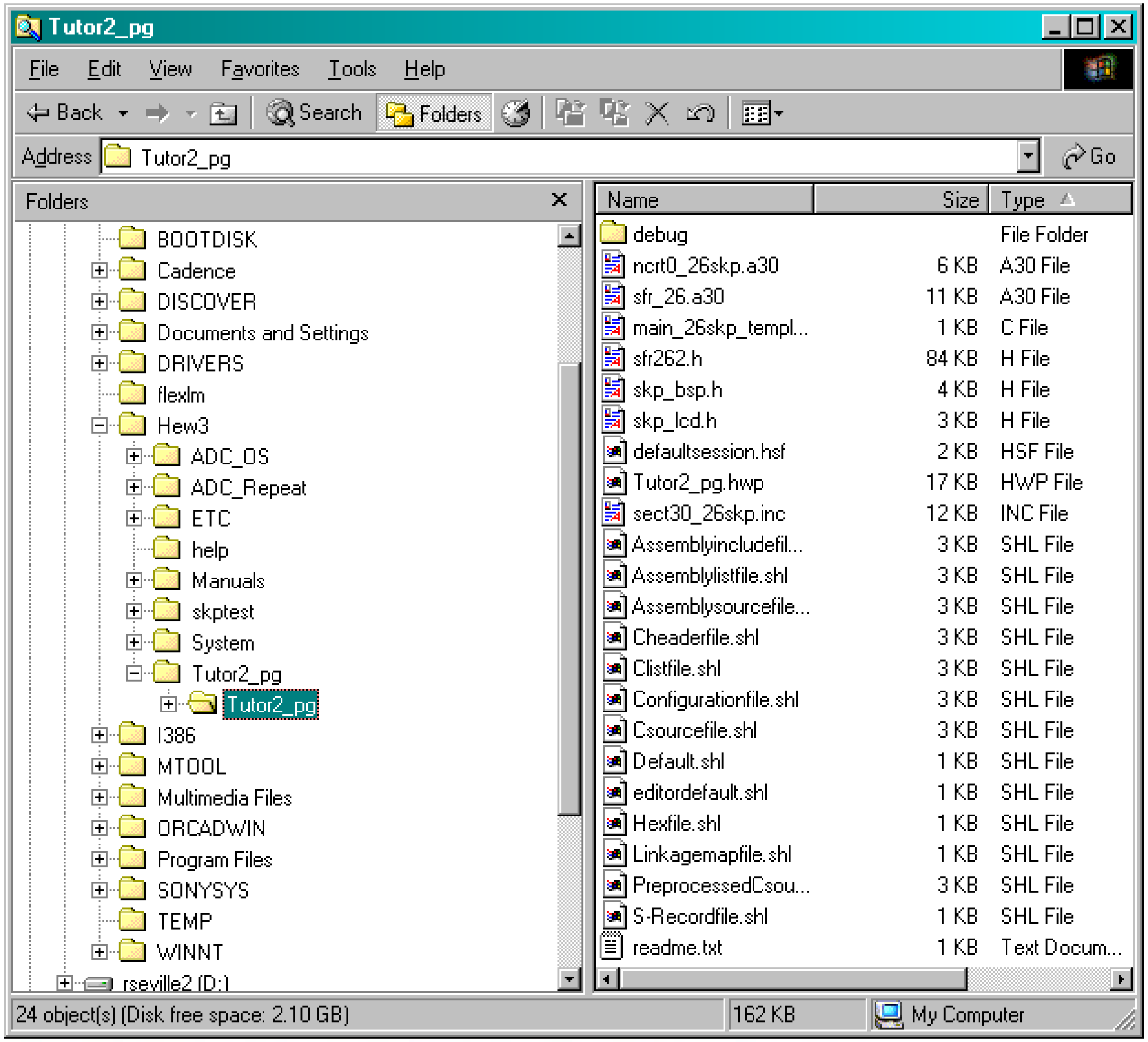
SKP project generator Summary (1/2)

The following is a summary of what happens when creating a new project using the SKP project generator Empty Project:

1. The following files are created in the workspace or project directory (in this example, under C:\Hew3\Tutor2_pg):
   - A copy of the user startup (ncrt0_26skp.a30 and sect30_26skp.inc) files.
   - A .c file (main_26skp_template.c) that contains the main function.
   - Copies of the SKP header files (sfr26.h, skp_bsp.h, and skp_lcd.h) included in the .c file.
   - A copy of the sfr_26.a30 file so SFR labels are visible in KD30.
2. Links the startup files first as shown in the “Linking the startup files first” slides (slides 25 & 26).
3. Configures the external debugger to launch KD30 as shown in the “Launching KD30 from HEW” slide (slide 28).
4. Adds the phase to generate a hex file when building a project as shown in the slide “Generate a hex file for the Programmer” (slide 29).

SKP project generator Summary (2/2)

The project structure and files created when using the SKP project generator are shown here.



```
BOOTDISK
Cadence
DISCOVER
Documents and Settings
DRIVERS
flexlm
Hew3
    ADC_OS
    ADC_Repeat
    ETC
    help
    Manuals
    skptest
    System
    Tutor2_pg
        Tutor2_pg
I386
MTOOL
Multimedia Files
ORCADWIN
Program Files
SONYSYS
TEMP
WINNT
rseville2 (D:)
```

Name	Size	Type
debug		File Folder
ncrt0_26skp.a30	6 KB	A30 File
sfr_26.a30	11 KB	A30 File
main_26skp_templ...	1 KB	C File
sfr262.h	84 KB	H File
skp_bsp.h	4 KB	H File
skp_lcd.h	3 KB	H File
defaultsession.hsf	2 KB	HSF File
Tutor2_pg.hwp	17 KB	HWP File
sect30_26skp.inc	12 KB	INC File
Assemblyincludefil...	3 KB	SHL File
Assemblylistfile.shl	3 KB	SHL File
Assemblysourcefile...	3 KB	SHL File
Cheaderfile.shl	3 KB	SHL File
Clistfile.shl	3 KB	SHL File
Configurationfile.shl	3 KB	SHL File
Csourcefile.shl	3 KB	SHL File
Default.shl	1 KB	SHL File
editordefault.shl	1 KB	SHL File
Hexfile.shl	1 KB	SHL File
Linkagemapfile.shl	1 KB	SHL File
PreprocessedCsou...	3 KB	SHL File
S-Recordfile.shl	1 KB	SHL File
readme.txt	1 KB	Text Docum...

24 object(s) (Disk free space: 2.10 GB) | 162 KB | My Computer

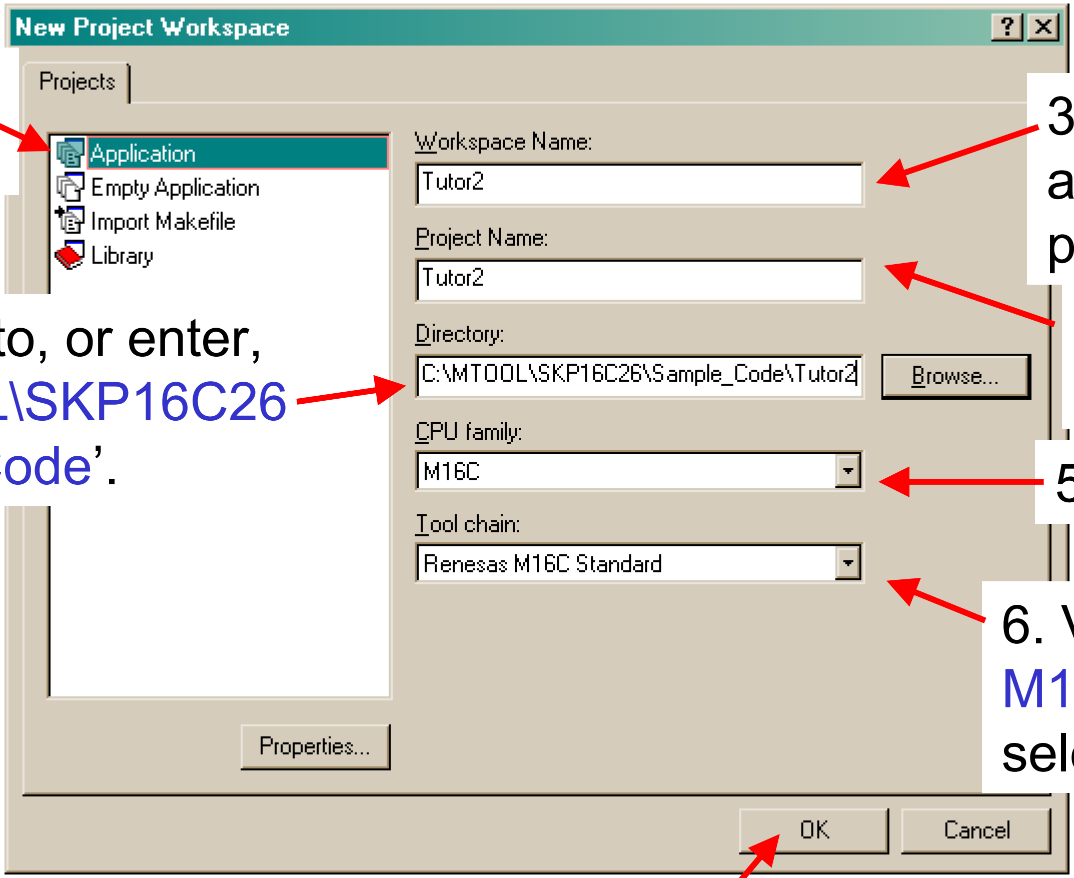
Creating a New Project Environment

The SKP project generator simplifies several steps in creating a new project but what if you already have source files? The following slides will walk you through the following:

- Creating a new workspace from scratch
- Adding source files
- Setting up the software development environment including linker, build, debugger, and programming configuration.

Creating a New Project (1/8)

Select ‘New Workspace’ from the File menu…



Workspace Name: Tutor2

Project Name: Tutor2

Directory: C:\MTOOL\SKP16C26\Sample_Code\Tutor2

CPU family: M16C

Tool chain: Renesas M16C Standard

1. Select ‘Application’.

2. Browse to, or enter, ‘C:\MTOOL\SKP16C26 \Sample_Code’.

3. Enter ‘Tutor2’ as name of the project.

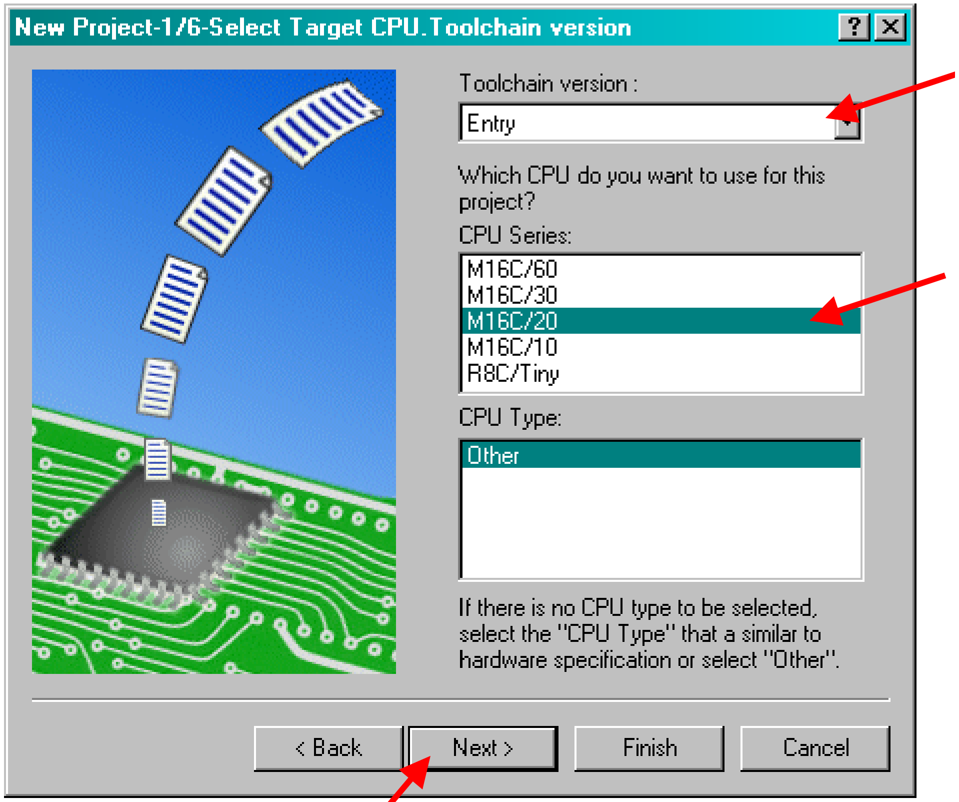
4. Leave as is (i.e. ‘Tutor2’).

5. Select M16C.

6. Verify Renesas M16C Standard is selected.

7. Click on ‘OK’ button.

Creating a New Project (2/8)



New Project-1/6-Select Target CPU.Toolchain version

Toolchain version : Entry

Which CPU do you want to use for this project?

CPU Series:

- M16C/60
- M16C/30
- M16C/20
- M16C/10
- R8C/Tiny

CPU Type: Other

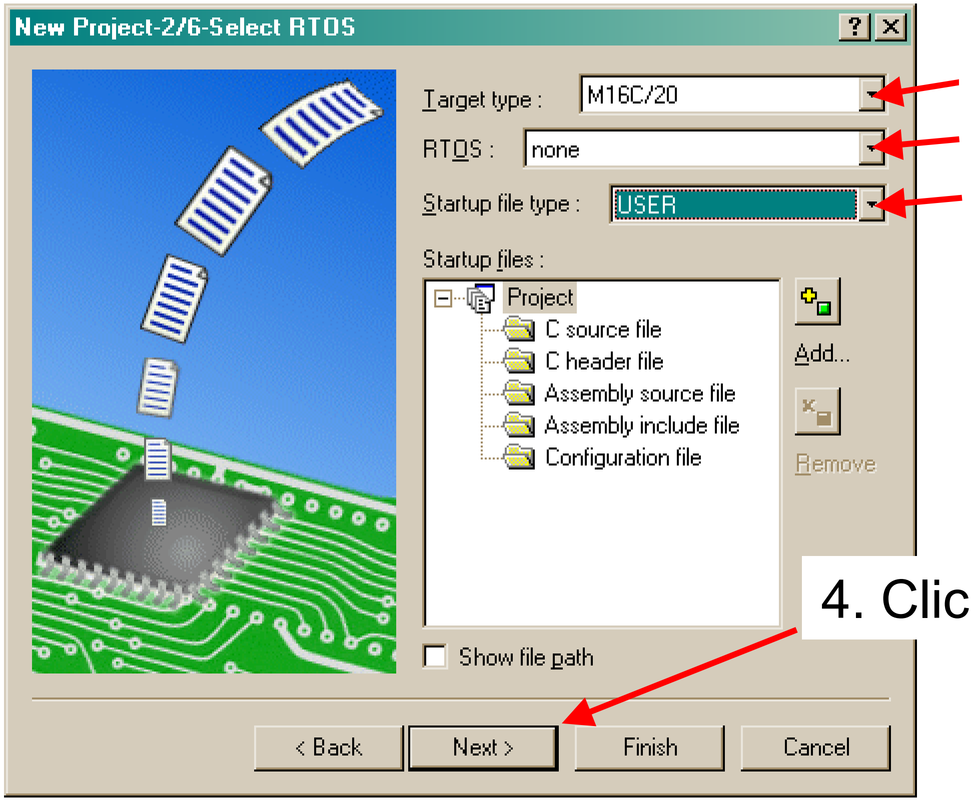
If there is no CPU type to be selected, select the "CPU Type" that a similar to hardware specification or select "Other".

1. Leave as is.

2. Select ‘M16C/20’.

3. Click on ‘Next’ button.

Creating a New Project (3/8)



New Project-2/6-Select RTOS

Target type : M16C/20

RTOS : none

Startup file type : USER

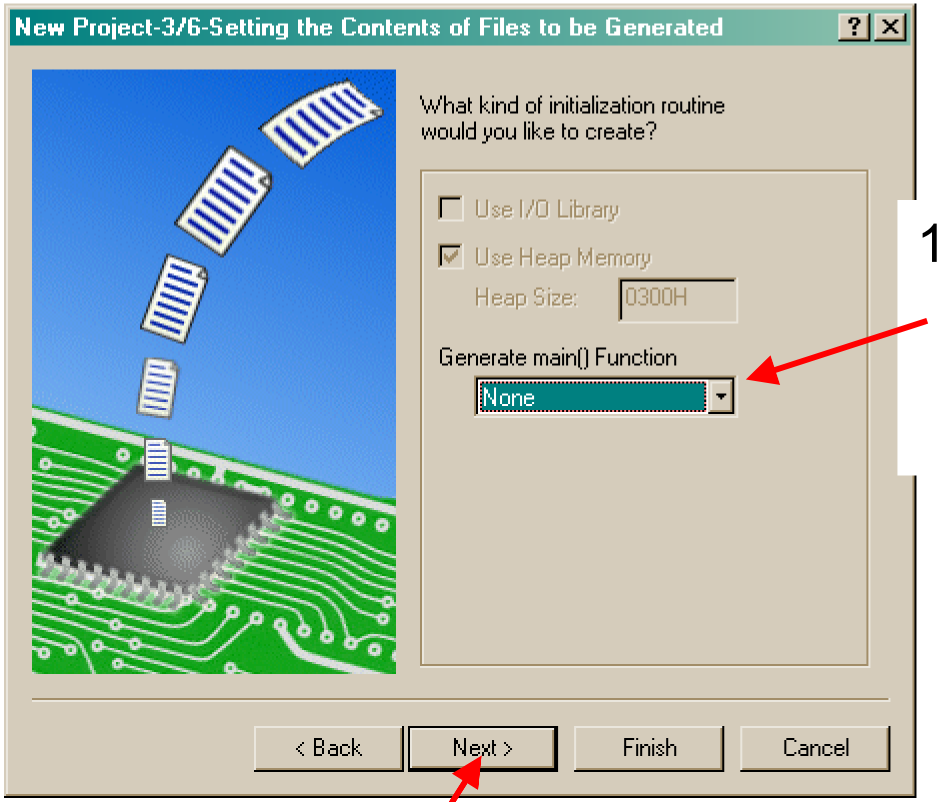
Startup files :

1. Select ‘M16C/20’.
2. Leave as is.
3. Select ‘USER’1.

4. Click on ‘Next’ button.

Note: 1. Differences between Default and USER Startup files is discussed after the project is created.

Creating a New Project (4/8)



New Project-3/6-Setting the Contents of Files to be Generated

What kind of initialization routine would you like to create?

□ Use I/O Library

☑ Use Heap Memory

Heap Size: 0300H

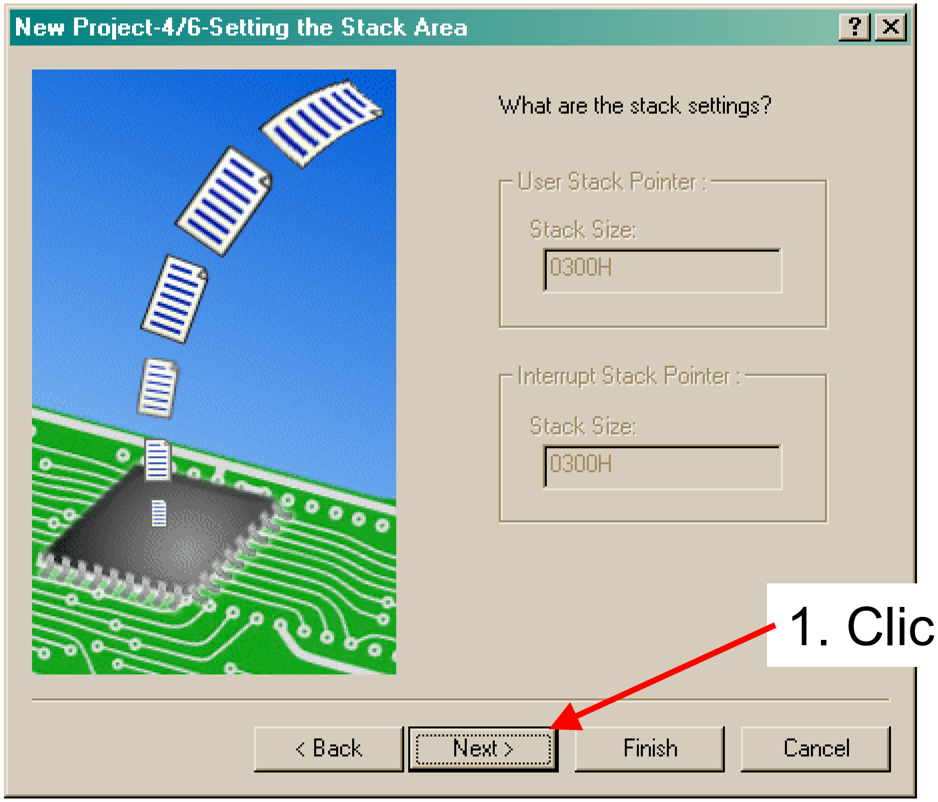
Generate main() Function: None

1. Select ‘None’ as we already have a file with the main() function.

2. Click on ‘Next’ button.

Creating a New Project (5/8)

The stack settings are grayed out because USER startup file was selected (see Creating a New Project 3/8). If DEFAULT startup file was selected, these settings become available.



New Project-4/6-Setting the Stack Area

What are the stack settings?

User Stack Pointer :

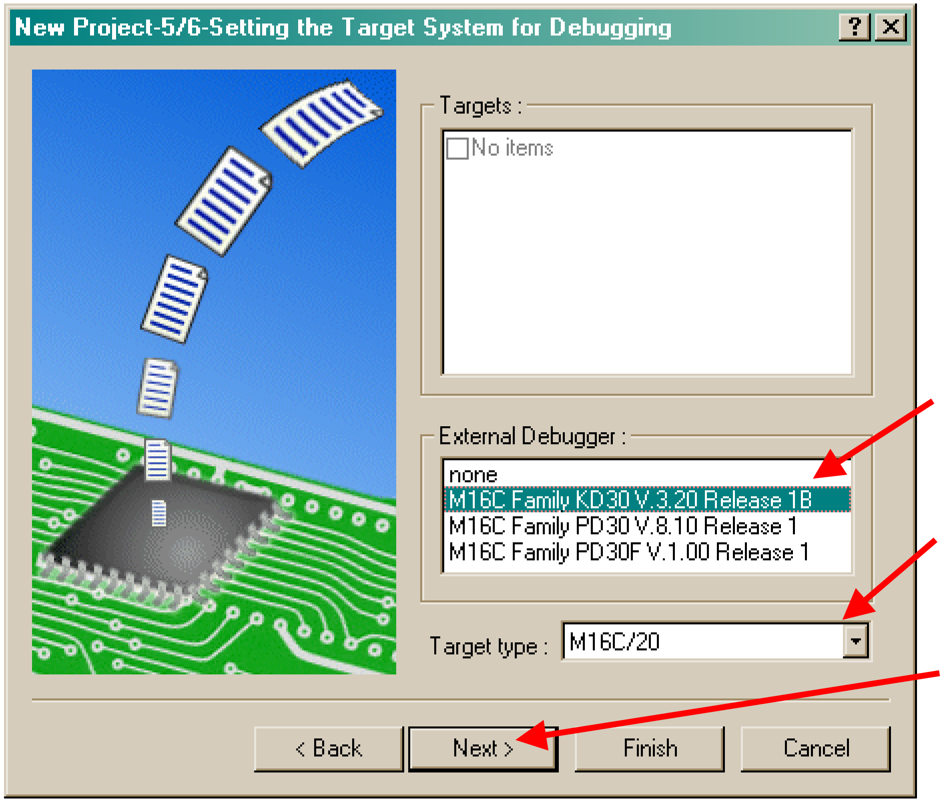
Stack Size: 0300H

Interrupt Stack Pointer :

Stack Size: 0300H

1. Click on ‘Next’ button.

Creating a New Project (6/8)



New Project-5/6-Setting the Target System for Debugging

Targets :

□ No items

External Debugger :

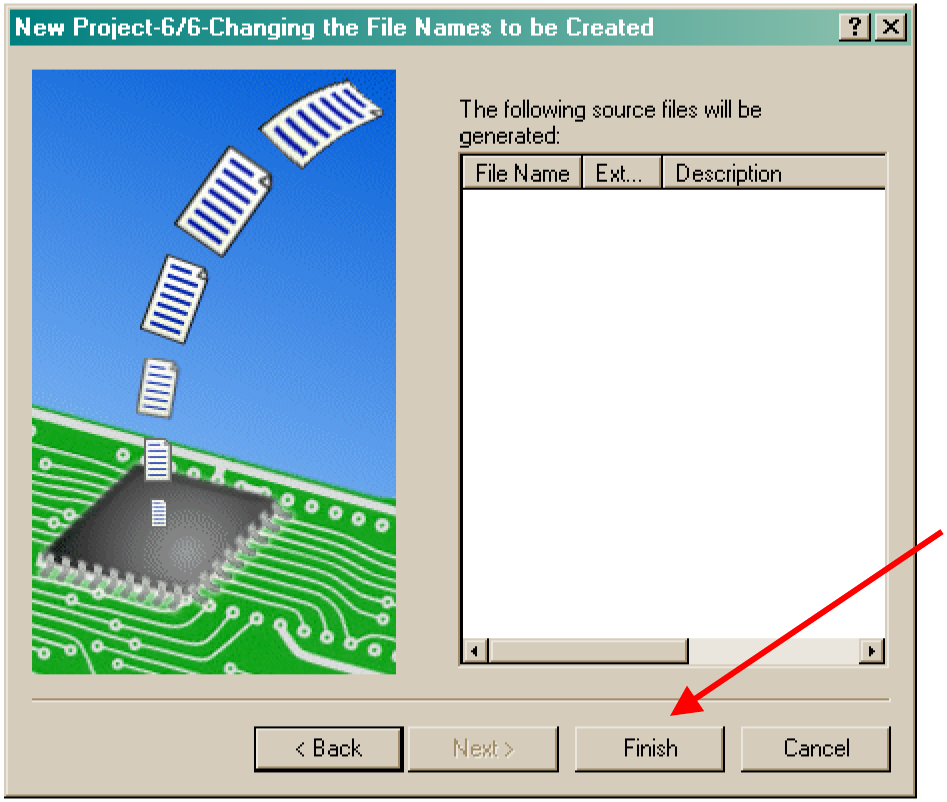
none
M16C Family KD30 V.3.20 Release 1B
M16C Family PD30 V.8.10 Release 1
M16C Family PD30F V.1.00 Release 1

Target type : M16C/20

1. Select ‘KD30’.
2. Leave as is.
3. Click on ‘Next’.

Creating a New Project (7/8)

We already have the source files and selected USER startup files (see Creating a New Project 3/8 and 4/8) and so, there are no source files that will be generated.



New Project-6/6-Changing the File Names to be Created

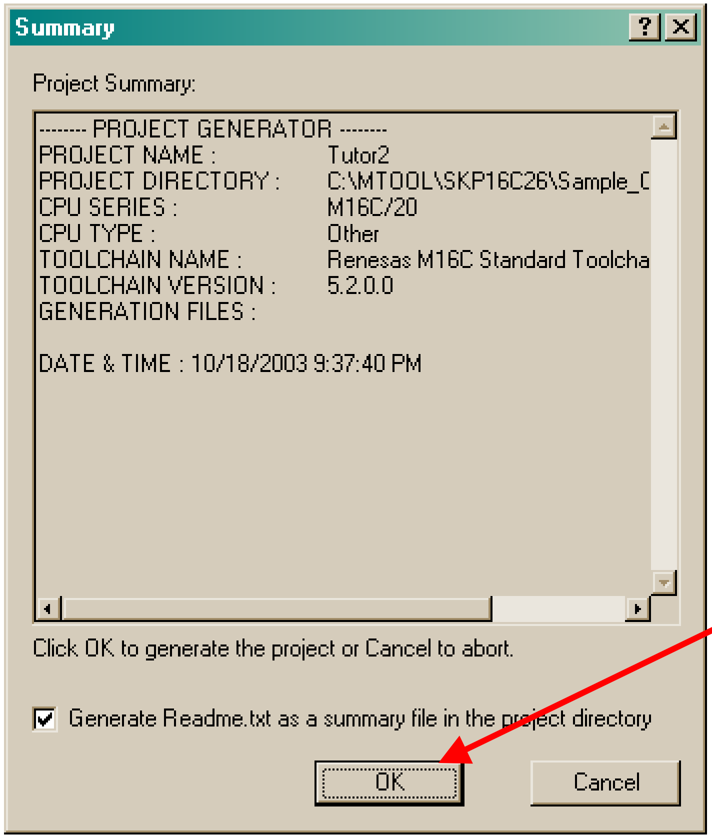
The following source files will be generated:

File Name	Ext...	Description

1. Click on 'Finish' to complete project creation process…

Creating a New Project (8/8)

A project summary is displayed to indicate a successful creation of the new project, tutor2.



Project Summary:

```
-------- PROJECT GENERATOR --------
PROJECT NAME :          Tutor2
PROJECT DIRECTORY :     C:\MTOOL\SKP16C26\Sample_C
CPU SERIES :            M16C/20
CPU TYPE :              Other
TOOLCHAIN NAME :        Renesas M16C Standard Toolcha
TOOLCHAIN VERSION :     5.2.0.0
GENERATION FILES :

DATE & TIME : 10/18/2003 9:37:40 PM
```

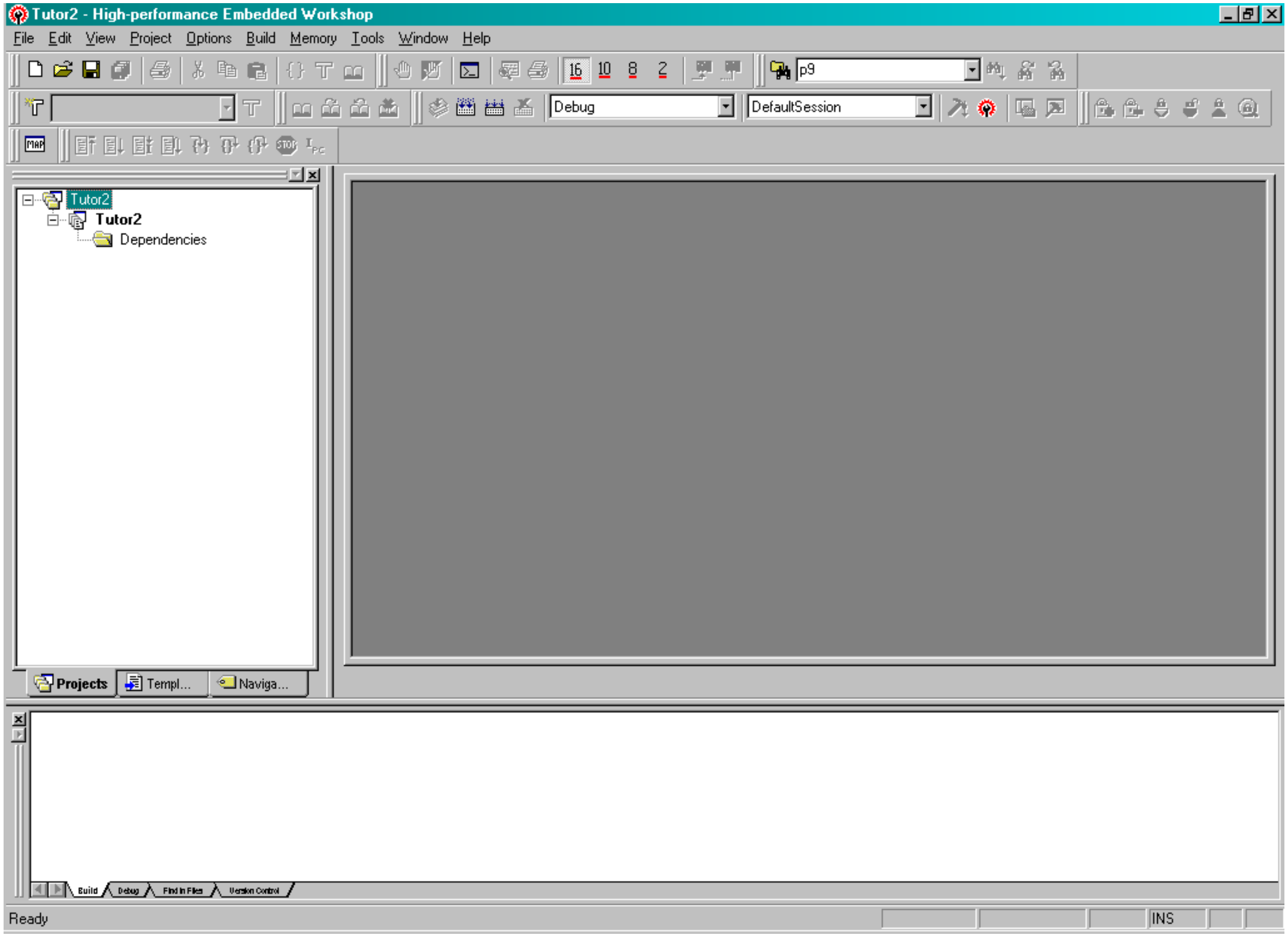
Click OK to generate the project or Cancel to abort.

☑ Generate Readme.txt as a summary file in the project directory

1. Click on 'OK'.

HEW

After creating the project, HEW will look similar to the figure below.



Tutor2 - High-performance Embedded Workshop

p9

Debug

DefaultSession

Tutor2
- Tutor2
  - Dependencies

Ready

Default and User (SKP) Startup Files

Except for comments, the differences between the default (*ncrt0.a30* and *sect30.inc*) and user (*ncrt0_26skp.a30* and *sect30_26skp.inc*) startup files are listed below:

Ncrt0.a30

- **BCLK – Set to f1 (div by 1) of Xin (BCLK = Xin frequency) in user file.**
- **Stack sizes – Smaller (0x100) in user file [default at 0x300].**
- **Heap size – 0 (no heap size) in user file.**
- **Vector Base Address – 0xFF800 for (M16C/26) in user file [default at 0xFFD00 (M16C/62)]**
- **Section Allocation – Calls user sect30 file (sect30_26skp.inc) [default calls default sect30.inc].**
- **Standard I/O Initialization – No initialization in user file (done using mcu_init from main()) [default file calls _init routine].**

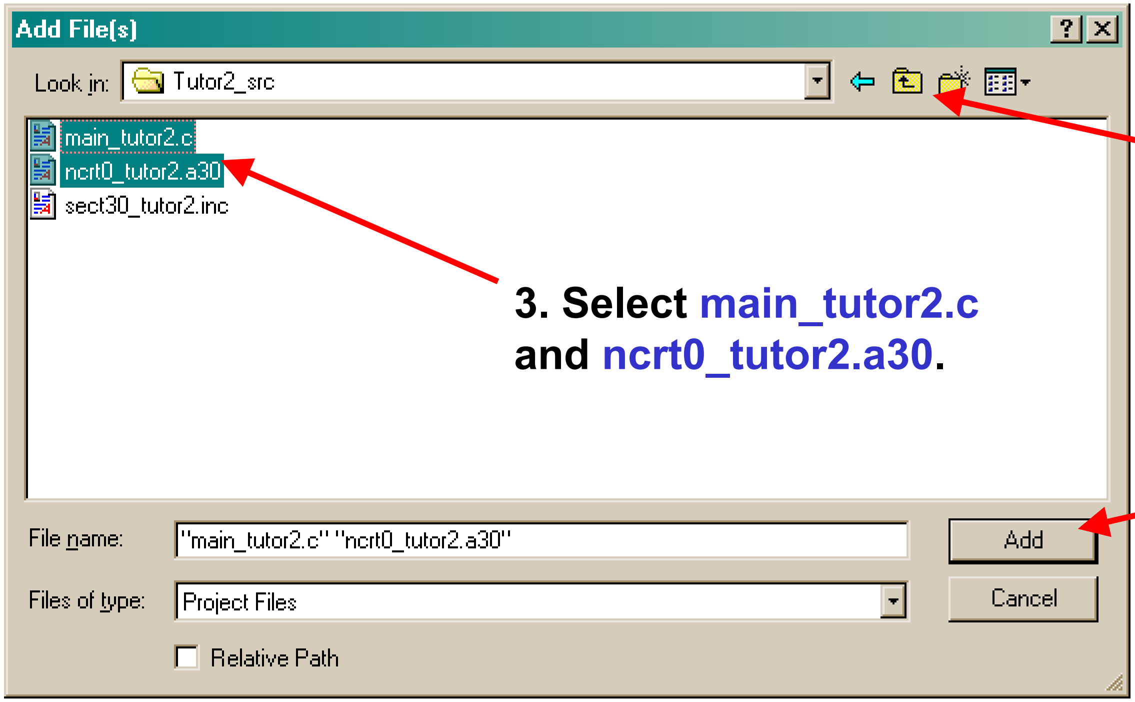
Sect30.inc

- **Variable Vector Definition – Simplified and described in user file**

Note: ***You can use the default startup files but ensure that you understand the how to make the necessary modifications. The customized startup files for the SKP can be found under C:\MTOOL\SKP16C26\Sample_Code\Startup_files directory.***

Adding Source Files (1/2)

1. From HEW's **Project** menu, select **Add Files** and the Add Files dialog box is displayed.



File name: "main_tutor2.c" "ncrt0_tutor2.a30"

Files of type: Project Files

□ Relative Path

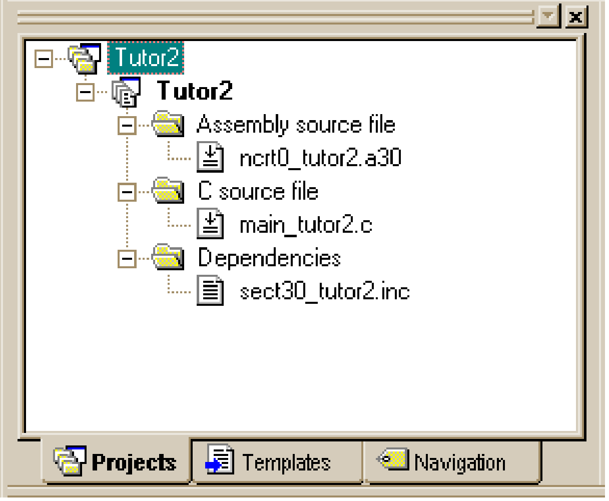
2. Browse to C:\MTOOL \SKP16C26 \Sample_Code\Tutor2_src folder, which contains the source codes for this tutorial.

3. Select main_tutor2.c and ncrt0_tutor2.a30.

4. Click on Add.

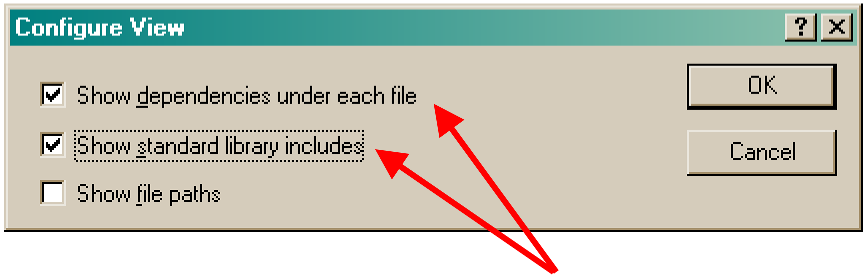
Adding Source Files (2/2)

After adding source files, HEW's workspace will look like the figure below.



- Tutor2
  - Tutor2
    - Assembly source file
      - ncrt0_tutor2.a30
    - C source file
      - main_tutor2.c
    - Dependencies
      - sect30_tutor2.inc

1. To display dependencies per source file, right-click on the Workspace window, and click on the Configure View.



Configure View

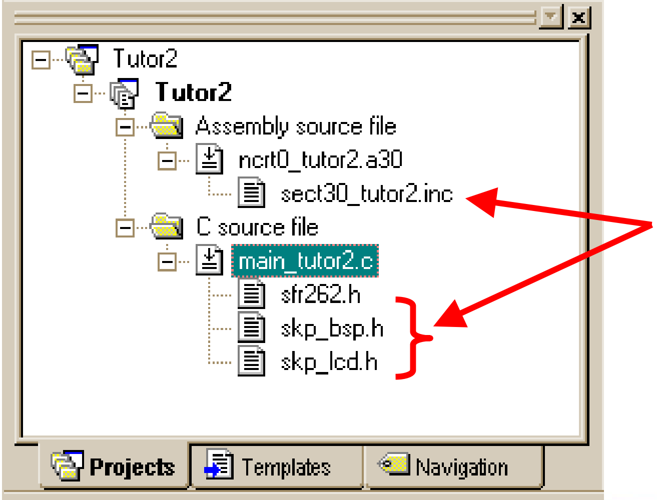
☑ Show dependencies under each file

☑ Show standard library includes

□ Show file paths

2. Click on check boxes in Configure View dialog box.

3. Click on OK button.



- Tutor2
  - Tutor2
    - Assembly source file
      - ncrt0_tutor2.a30
        - sect30_tutor2.inc
    - C source file
      - main_tutor2.c
        - sfr262.h
        - skp_bsp.h
        - skp_lcd.h

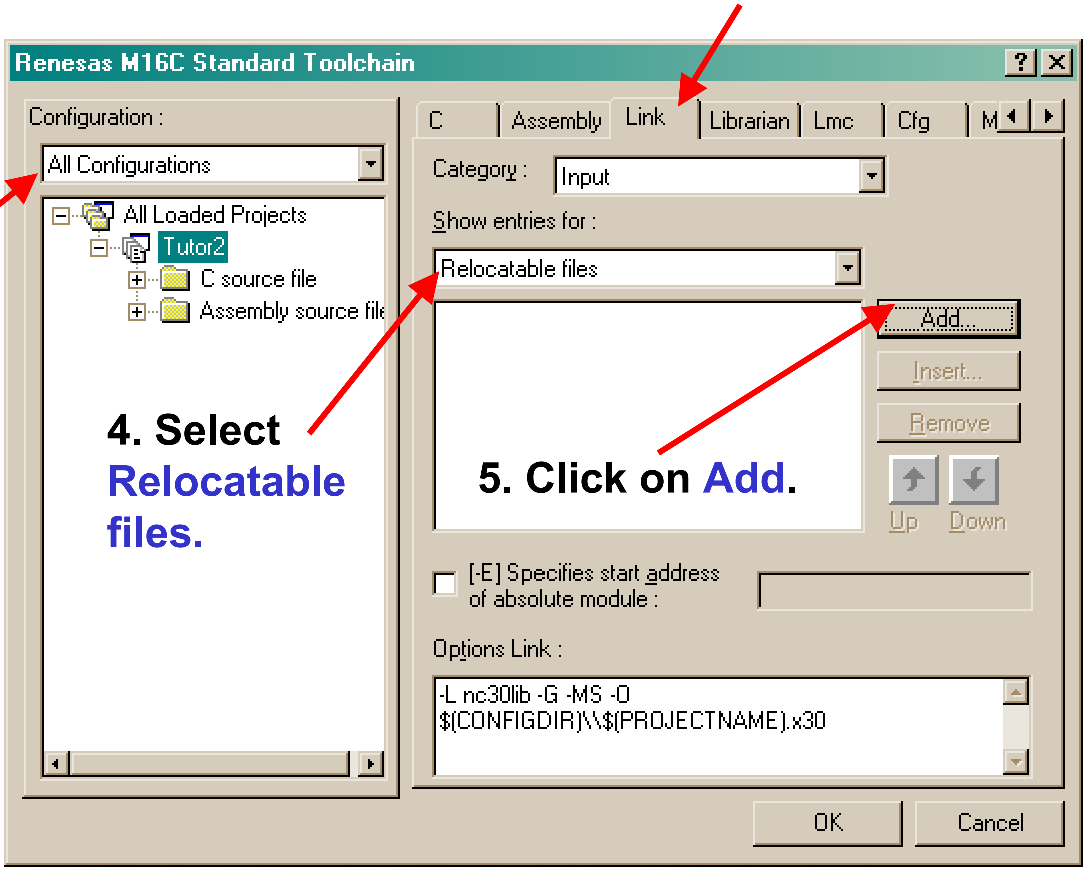
Displaying the source file dependencies this way makes it easier to verify if the header files are included before building the project.

Linking the Startup File First (1/2)

After checking dependencies, NC30 must be setup to compile and link the startup files first. The startup files contain information on memory addresses and sections which the linker uses to place the code at the correct addresses.

1. Select **Renesas M16C Standard Toolchain** from the **Option** menu and this window is displayed.

3. Click on **Link** tab.



Configuration : All Configurations

All Loaded Projects
Tutor2
C source file
Assembly source file

Category : Input

Show entries for : Relocatable files

□ [-E] Specifies start address of absolute module :

Options Link :

```
-L nc30lib -G -MS -O
$(CONFIGDIR)\\$(PROJECTNAME).x30
```

2. Select **All Configurations**.

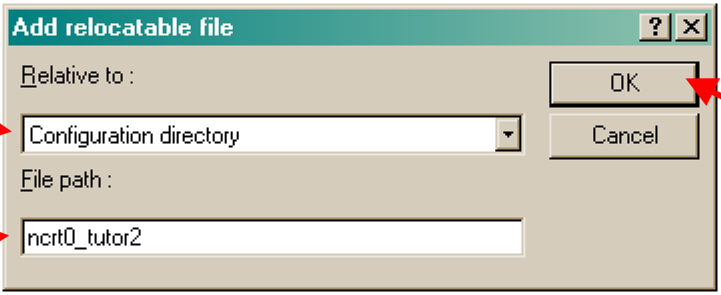
4. Select **Relocatable files.**

5. Click on **Add**.

Linking the Startup File First (2/2)

1. Select **Configuration directory**.

2. Enter the startup source file, i.e. **ncrt0_tutor2**.

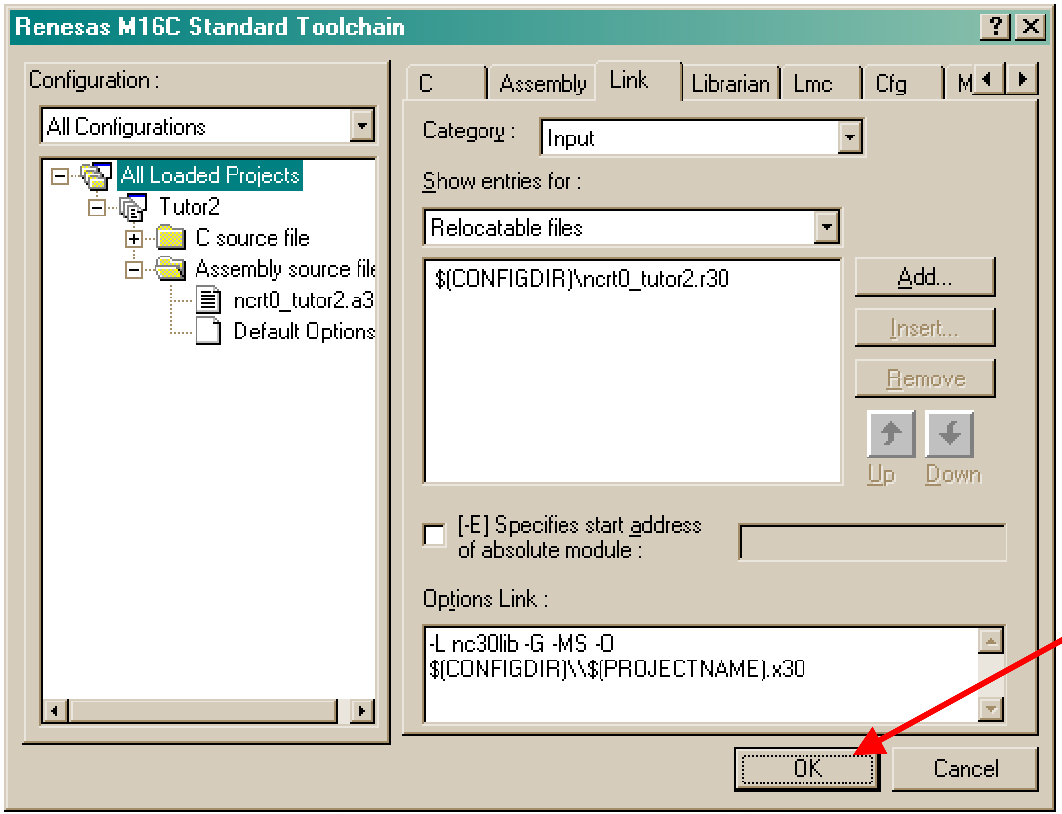


Add relocatable file

Relative to : Configuration directory

File path : ncrt0_tutor2

3. Click on **OK**.



Renesas M16C Standard Toolchain

Configuration : All Configurations

```
All Loaded Projects
  Tutor2
    C source file
    Assembly source file
      ncrt0_tutor2.a30
      Default Options
```

Category : Input

Show entries for : Relocatable files

```
$(CONFIGDIR)\ncrt0_tutor2.r30
```

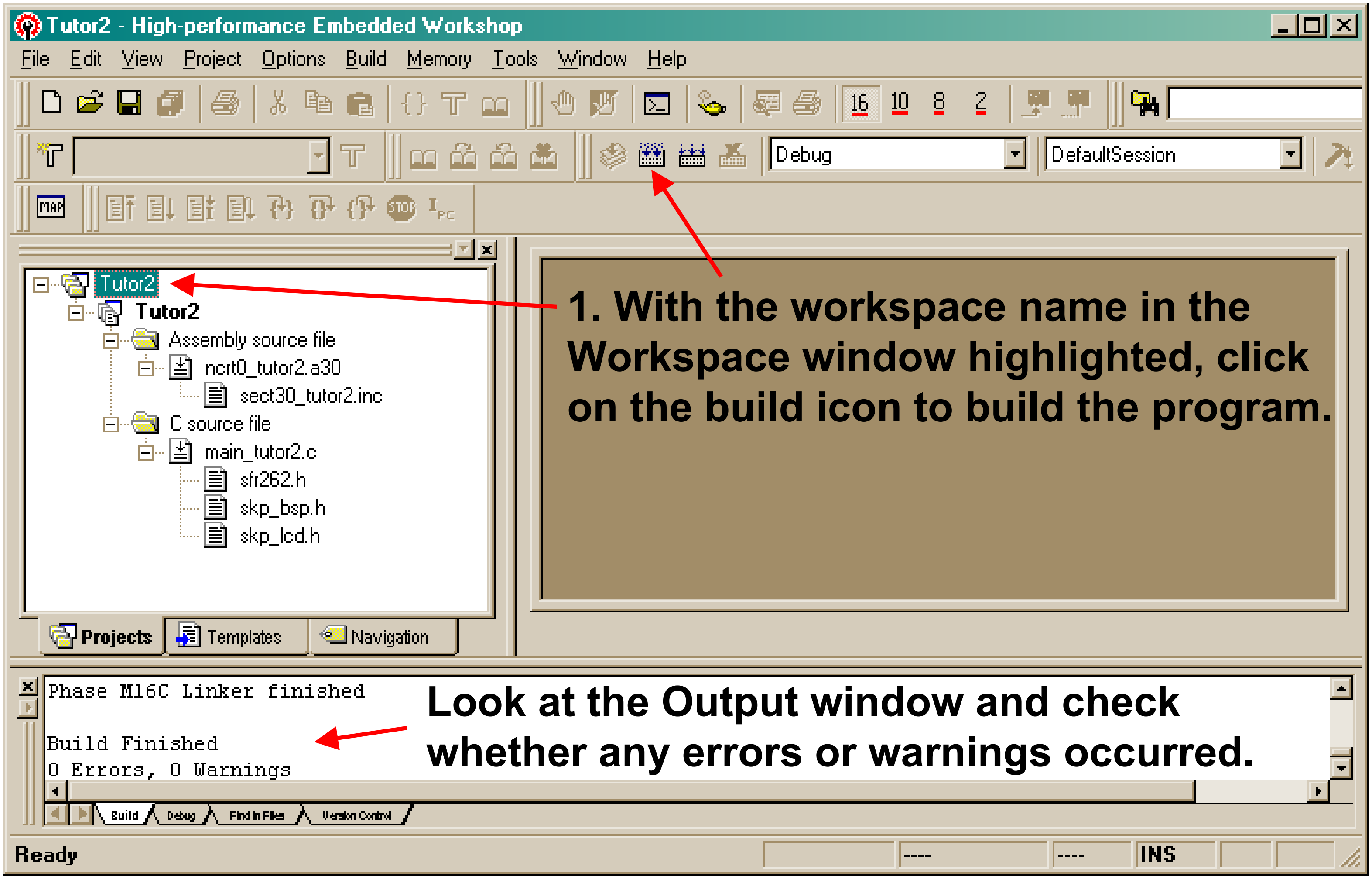
☐ [-E] Specifies start address of absolute module :

Options Link :

```
-L nc30lib -G -MS -O
$(CONFIGDIR)\\$(PROJECTNAME).x30
```

4. Click on **OK**.

Building (compile & link) the Program



```
Phase M16C Linker finished

Build Finished
0 Errors, 0 Warnings
```

1. With the workspace name in the Workspace window highlighted, click on the build icon to build the program.

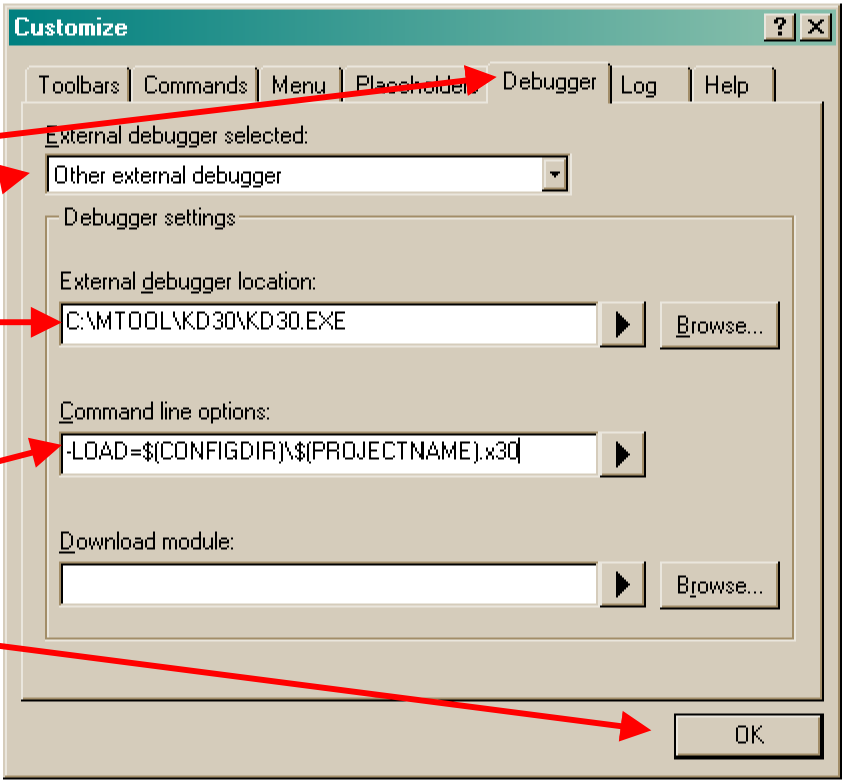
Look at the Output window and check whether any errors or warnings occurred.

If there are no errors, you can try running the program on the SKP16C26 Board using the ICD and KD30 Debugger.

Launching KD30 from HEW

KD30 can be launched directly from HEW, but HEW must be configured as shown below.

1. Select **Customize from** from the **Tools** menu.
2. Click on **Debugger** tab
3. Select **Other external debugger**.
4. Locate **KD30** from **C:\MTOOL\KD30** folder.
5. Enter **-LOAD=$(CONFIGDIR)\$(PROJECTNAME).x30**.
6. Click on **OK**.



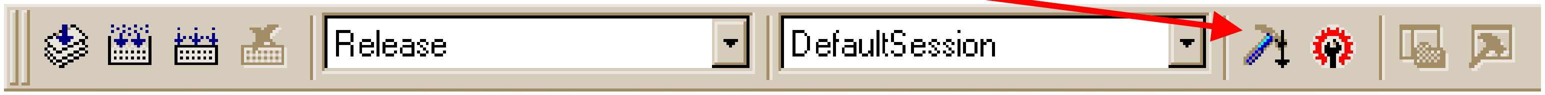
External debugger selected: Other external debugger

External debugger location: C:\MTOOL\KD30\KD30.EXE

Command line options: -LOAD=$(CONFIGDIR)\$(PROJECTNAME).x30

Download module:

To launch (and download .x30 file) KD30, click on this icon.



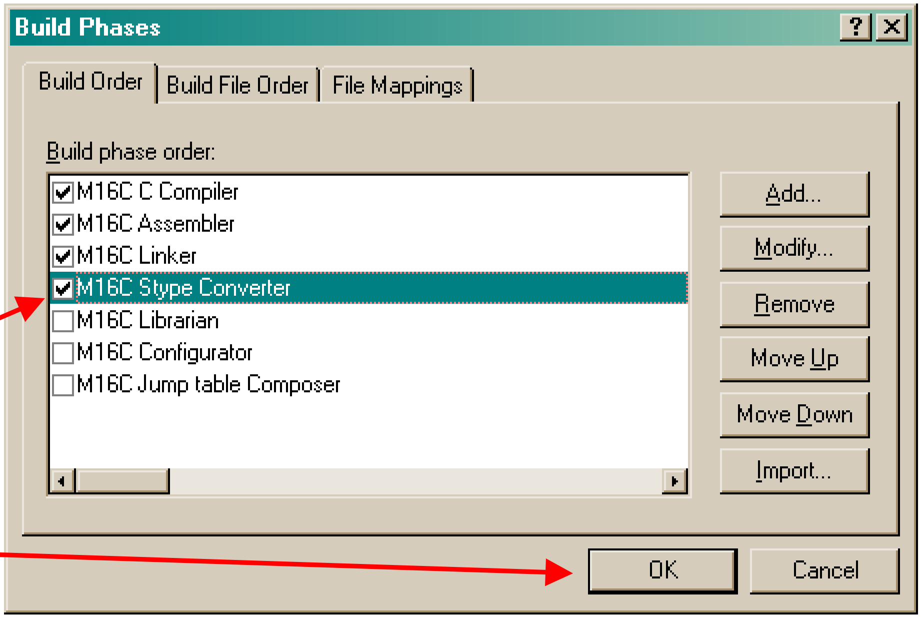
Generate a hex file for the Programmer

After building a program, a file with an .x30 is generated. For this example, 'tutor2.x30' is generated. This .x30 file can be downloaded using KD30. However, device programmers, including the FoUSB Programmer, use hex files in Motorola or Intel format.

1. To generate a hex file during the build process, select Build Phases from the Option menu.

2. Click on M16C Stype converter checkbox.

3. Click on OK.



Build phase order:

- ☑ M16C C Compiler
- ☑ M16C Assembler
- ☑ M16C Linker
- ☑ M16C Stype Converter
- ☐ M16C Librarian
- ☐ M16C Configurator
- ☐ M16C Jump table Composer

With the feature enabled, a hex (.mot) file is created when building the project. Look for a Tutor2.mot file under C:\MTOOL\SKP16C26\Sample_Code\Tutor2\Tutor2\Debug' folder.

End of Tutorial 2

This is the end of the tutorial. You can try modifying or downloading other sample programs from the \Sample_Code directory.

The completely built tutor2 project can be found in the Zip file under C:\MTOOL\Sample_Code\tutor2_src folder.

Be sure to check out the references on the following pages.

Have Fun!!

References and Recommended Reading

All documents that came with the SKP can be found using the “Document Description” from the Start > Programs > Renesas-Tools > SKP16C26 menu.

- **SKP16C26 User’s Manual:** This is a “must read” document! It details all the things you need to know on how to use the Starter Kit.
- **HEW User’s Manual:** To fully understand and get the most out of HEW. This is recommended reading.
- **KD30 Version X.XX Help:** The tutorial only covered the basics of KD30. Check out the Help menu to find out all of KD30’s features.
- **NC30 Version X.XX User’s Manual:** Check this manual out for features specific to the NC30 compiler.
- **M16C/26 Datasheet and SKP16C26 Board Schematic:** These are required to write user application programs.
- **RTA-FoUSB-MON User’s Manual:** Read this manual to understand how the ICD works.

References and Recommended Reading

- **M16C/10/20/60 Series C Language Programming Manual:** This is a great document for any level of programmer. The first chapter is an intro to C programming. The next chapter explains the memory map of C programs on microcontrollers and the role of startup programs.
- **M16C/10/20/60 Series Software Manual:** This document describes the instruction set and timing information for the M16C/20/60 series CPU cores.
- **AS30 Version X.XX User's Manual:** Read this manual if you plan on writing programs in Assembly or when making changes to the startup file.
- **Application Notes and Sample Programs:** Application notes and other sample programs can be accessed from Renesas Technology America's website: http://www.renesas.com.