



Keil™ MDK-ARM Quick Start for
Holtek's HT32 Series Microcontrollers

Revision: V1.10 Date: August 25, 2011

www.holtek.com

Table of Contents

1 Introduction	5
About the Quick Start Guide	5
About the Keil MDK-ARM	6
2 System Requirements	7
3 Software Installation.....	8
Keil MDK-ARM Installation	8
Installing the Keil HT32 Support Package (For MDK-ARM v4.20 or Below)	8
4 Installing the USB Debug Adapter	11
5 Connecting to the Target Board	12
6 Keil MDK-ARM Quick Start	13
Create and Save New Project	13
Project Options Setup	16
Open Project Options Setup Page.....	16
Device Selection	17
Configure the USB Debug Adapter.....	18
Flash Loader Setup	20
Adding Source Files to the Project	23
7 Compiling the Project.....	28
8 Download and Debug	29
Enter/Exit Debug Mode.....	29
Free Running, Break, and Single Step Operation	31
Breakpoints.....	32
Memory Window	34
9 Mass Erase	36
10 Option Bytes Programming	37
11 Using the CodeSourcery GNU Toolchain with Keil µVision.....	40
Installing the Sourcery G++ Lite Edition	40
Configuring Keil µVision for Sourcery G++	40
12 Conclusion	46

List of Figures

Figure 1. Software Development Flow	5
Figure 2. Keil™ MDK-ARM Software Development Environment.....	6
Figure 3. System Architecture and Requirements	7
Figure 4. Support Package Installation.....	8
Figure 5. Support Package Installation – Destination.....	9
Figure 6. Support Package Installation – Start Install.....	9
Figure 7. Support Package Installation – Completion.....	10
Figure 8. PC, USB Debug Adapter and Target Board Connection	12
Figure 9. Create New Project	13
Figure 10. Create New Project Dialog	14
Figure 11. CPU Database Dialog.....	14
Figure 12. Select Project Device	15
Figure 13. Startup Code Inquiry Window.....	15
Figure 14. Target Menu Options	16
Figure 15. Select Project Device	17
Figure 16. Target Debug Tab Page Options.....	18
Figure 17. ARM Target Driver Setup Page.....	19
Figure 18. Utilities Tab Menu	20
Figure 19. Target Driver Flash Download Tab.....	21
Figure 20. Flash Programming Algorithm Select	21
Figure 21. Flash Programming Algorithm	22
Figure 22. New File Creation	23
Figure 23. Save File.....	24
Figure 24. Add File to Group.....	25
Figure 25. Add File to Group – Specify File Name and Path	25
Figure 26. Components, Environment and Books Function	26
Figure 27. Modify Project Components	26
Figure 28. Project Component List	27
Figure 29. Build All target Files	28
Figure 30. Compiler Messages.....	28
Figure 31. Start Debugging Session.....	29
Figure 32. Debugging Session	30
Figure 33. Run Icon	31
Figure 34. Stop Icon	31
Figure 35. Step Icon	31
Figure 36. Breakpoint Setup.....	32
Figure 37. Breakpoint Program Stop	33
Figure 38. Open Memory Window	34
Figure 39. Address 0x20001000 Memory Contents Window.....	34
Figure 40. Memory Content Examination	35
Figure 41. Mass Erase Function.....	36
Figure 42. Build Output Message Window	36
Figure 43. Option Bytes Setup File.....	37

Figure 44. Add Option Bytes File	38
Figure 45. Add Files to Group Dialog.....	38
Figure 46. Option Bytes Configuration Wizard	39
Figure 47. Open Components, Environment and Books Dialog	41
Figure 48. Folders/Extensions Setup Change	42
Figure 49. "Use GNU Compiler" Warning Message.....	42
Figure 50. Linker Options	43
Figure 51. C Compiler Options	44
Figure 52. GNU Toolchain Compiler Messages.....	45

1 Introduction

The purpose of this quick start guide is to familiarise users with the Keil™ Microcontroller Development Kit (MDK-ARM) for the Holtek HT32 series of microcontrollers. The guide also includes information on the development kit installation and configuration.

About the Quick Start Guide

The guide includes information on how to setup the Keil™ MDK-ARM as well as a guide for installing the HT32 support package for Keil™. Necessary information is also provided about using the Keil μVision4 IDE (Integrated Development Environment) to compile and run software projects. A chapter is also dedicated on how to use the CodeSourcery GNU Toolchain with Keil™ μVision.

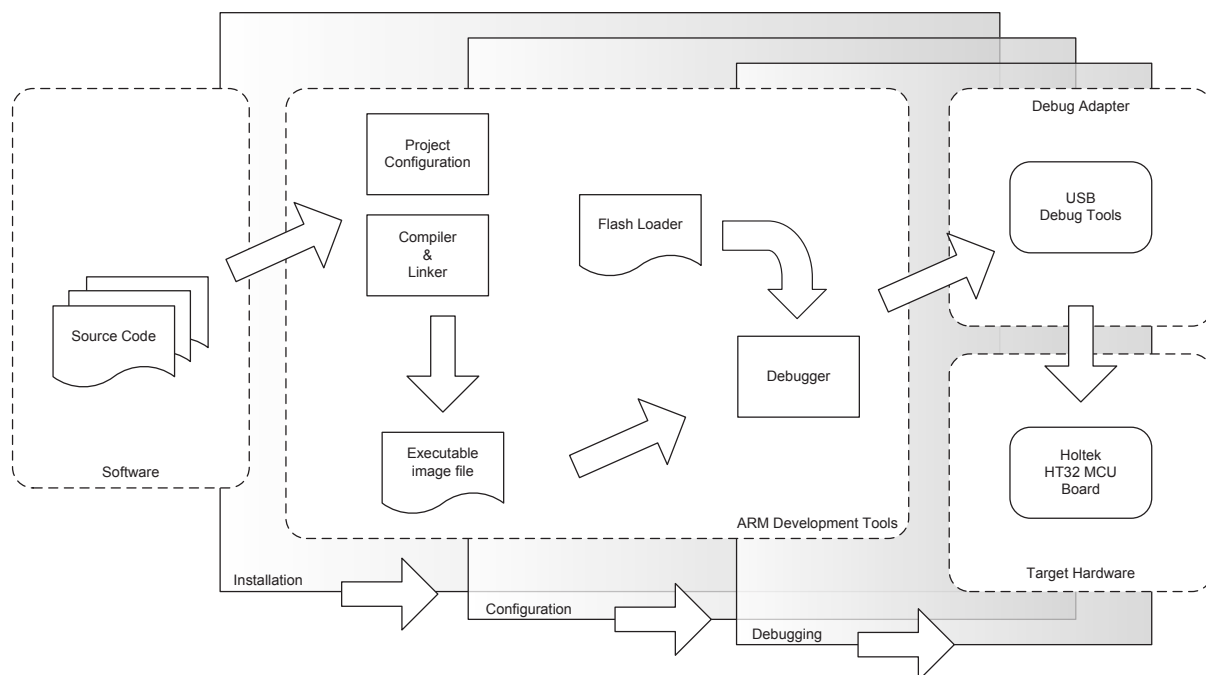
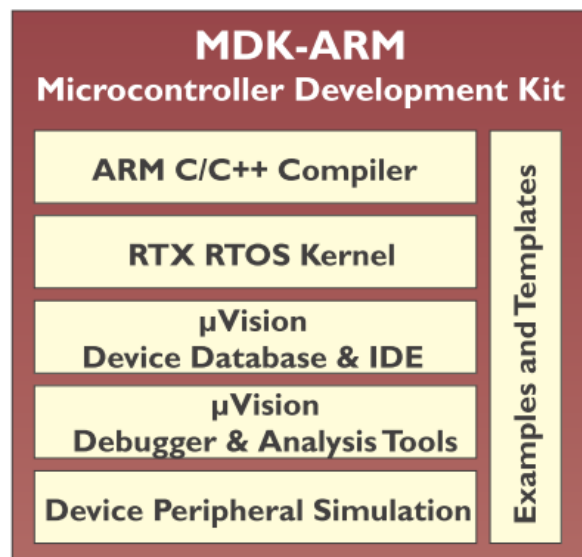


Figure 1. Software Development Flow

About the Keil MDK-ARM

The Keil™ MDK-ARM is a software development environment for ARM-based microcontrollers. The environment includes functions such as source code editor, compiler, assembler, linker, project management, flash programmer and debugger. All of these functions are integrated into μ Vision4 IDE which helps to create and debug C/C++/Assembler source files. The MDK-ARM provides the following key features for embedded applications.

- Supports Cortex-M series, Cortex-R4, ARM7 and ARM9 devices
- μ Vision IDE
 - Device database for all supported devices
 - Editor, project management and compilation
 - Debugger environment including trace and analysis tools
 - Simulation environment
- RealView Compilation Tool (RVCT)
 - ARM C/C++ Compiler (armcc)
 - ARM Macro Assembler (armasm)
 - ARM Linker (armlink)
 - ARM Utilities (Librarian and FromELF)
- Microlib-optimized run-time library
- Keil™ RTX Real-Time Operating System (RTOS)
- Flash Loader for Flash memory programming
- Example code for various boards and devices



NOTE: The above figure was extracted from the Keil website: <http://www.keil.com>

Figure 2. Keil™ MDK-ARM Software Development Environment

2 System Requirements

To get going with this quick start guide, several components are required as listed below:

- A target board with a HT32 series MCU
- A hardware debug adapter such as ULINK2 or Holtek's e-Link32
- A host computer running Microsoft® Windows® XP, Vista, or Windows® 7
 - A recommended 1 GB RAM and 500 MB of available hard-disk space
 - XGA (1024x768) colour monitor or higher resolution display
 - Mouse or other pointing device
 - A CD-ROM drive (optional)
- Keil™ MDK-ARM V3.04 or above

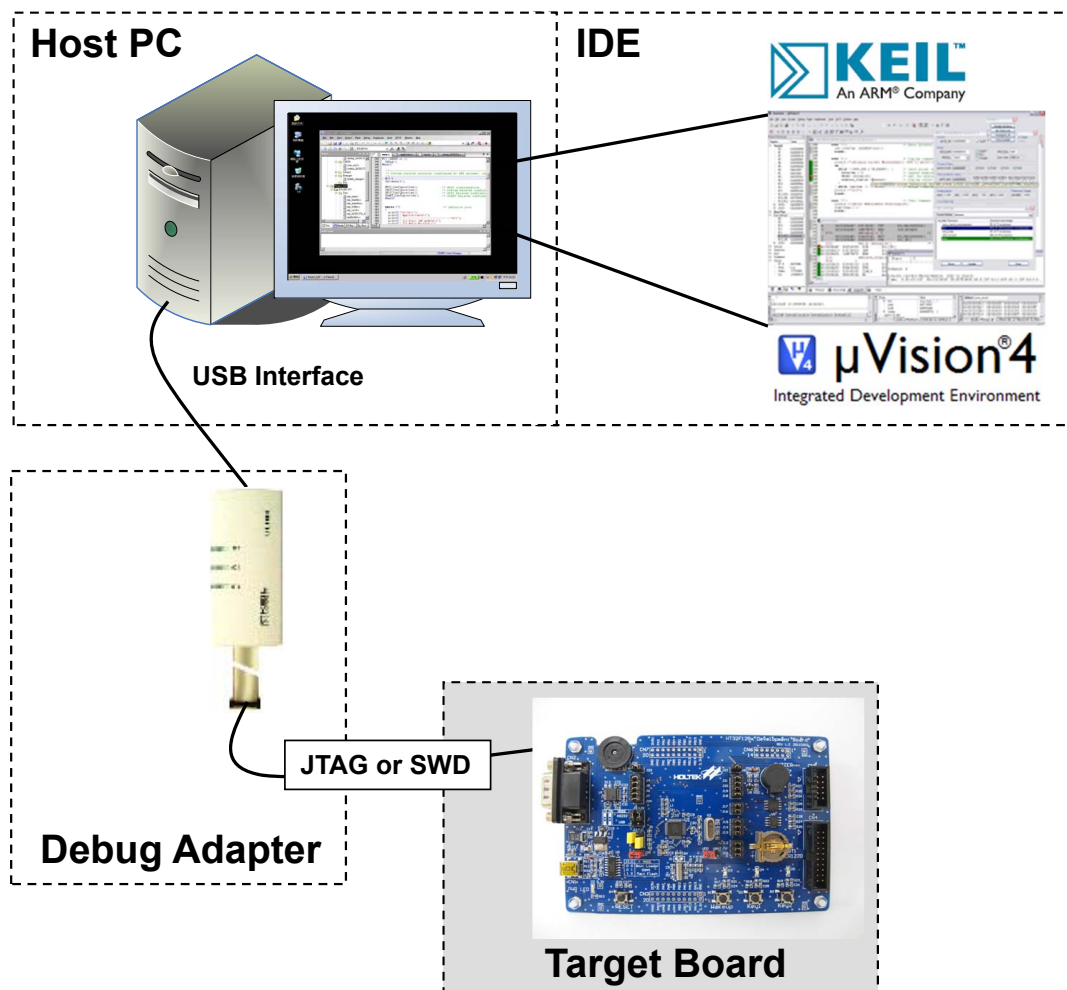


Figure 3. System Architecture and Requirements

3 Software Installation

Keil MDK-ARM Installation

The Keil™ MDK-ARM evaluation software can be downloaded from <http://www.keil.com> or from a CD-ROM provided by Holtek. The latest version of the MDK-ARM is regularly available at <http://www.keil.com/update>. Note that the evaluation version has a limitation on its 32 KB of image size. For more information about the setup process, please refer to the “Read Me First” document from the Keil’s website at: <http://www.keil.com/product/brochures/readmefirst.pdf>.

Installing the Keil HT32 Support Package (For MDK-ARM v4.20 or Below)

After the Keil™ MDK-ARM has been installed, the HT32 Support Package for Keil™ also has to be installed. This support package installs the device database, flash programming algorithms and all other files required for HT32 series MCU program development.

The following steps show how to install the support package into the Keil™ MDK-ARM.

- Step 1:** Obtain the latest version of the Support Package from the Holtek website or from the CD-ROM provided by Holtek. The filename is “HT32_Keil_Package_Vnnn.exe” where “nnn” represents the version number.
- Step 2:** Execute the support package installation program by double-clicking on “HT32_Keil_Package_Vnnn.exe”. Press the “Next” button to continue when the screen below appears.

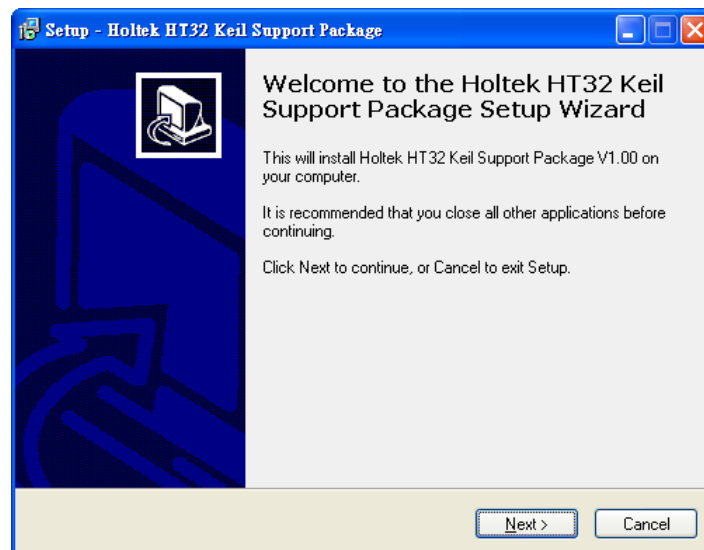


Figure 4. Support Package Installation

Step 3: The support package will detect the last installed path of the Keil MDK-ARM automatically. If the path found is not correct, press the “Browse” button to manually specify the required installed path for the MDK-ARM. Press the “Next” button to continue.

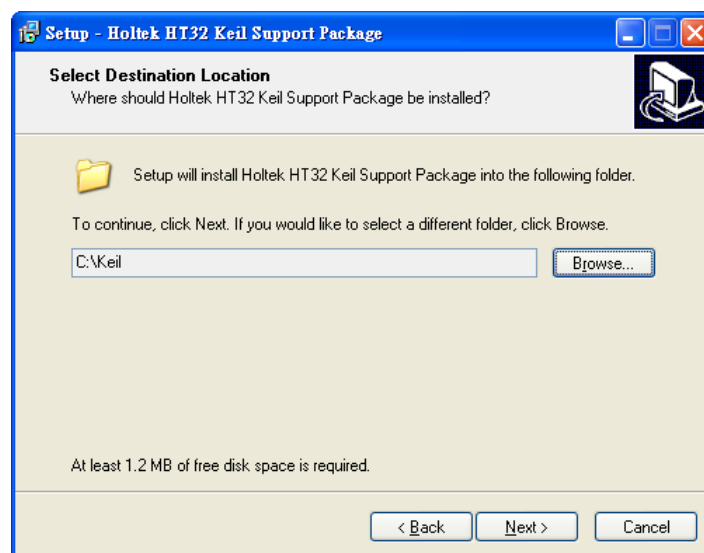


Figure 5. Support Package Installation – Destination

Step 4: After the target path is confirmed, the ready-to-install page will be shown. Press the “Install” button to start the installation.

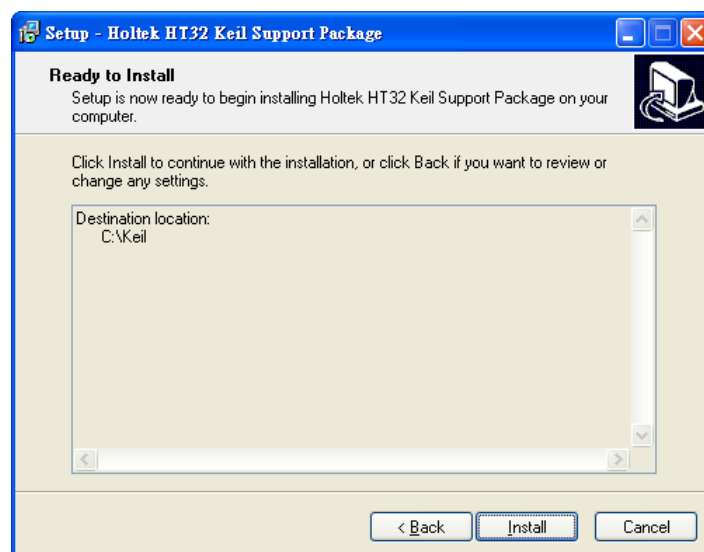


Figure 6. Support Package Installation – Start Install

Step 5: After the installation has completed, a completion page will appear. Choose whether or not to view the release note. Press the “Finish” button to exit the installation program.

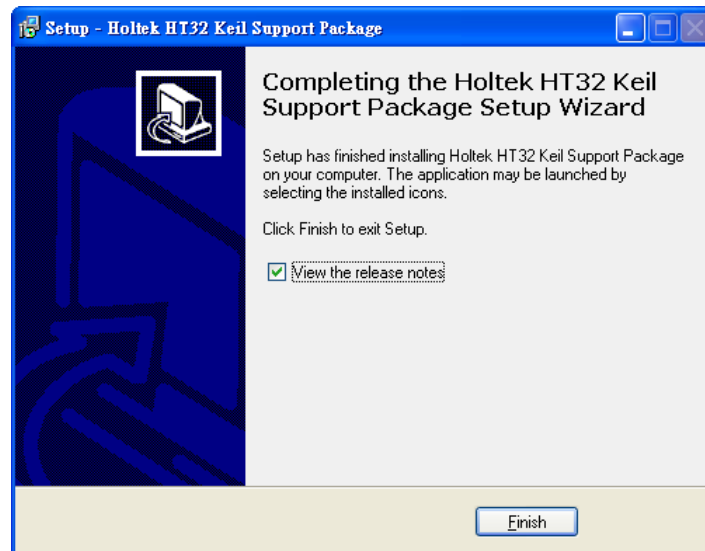


Figure 7. Support Package Installation – Completion

4 Installing the USB Debug Adapter

The e-Link32 drivers can be obtained from the Holtek website or from the CD-ROM provided by Holtek. Below are the configuration steps for the e-Link32 USB drivers.

Step 1: Connect the e-Link32 to the host PC via the USB port.

Step 2: The system will detect a new USB device and will start the driver installation procedure.

Step 3: Specify the driver path manually according to the USB debug adapter.

“C:\Program Files\Holtek HT32 Series\e-Link32 USB Driver\” – for e-Link32

The ULINK2 uses standard Human Interface Device (HID) drivers that are directly supported by Windows 2000/XP/Vista/7 operation systems. Therefore, no additional drivers are required.

5 Connecting to the Target Board

The target board can be powered by the USB port or by an external 5V DC adaptor by changing the on-board jumpers. Refer to the corresponding target board documents for details.

The USB debug adapter, ULINK2 or e-Link32, that is connected to the SWD or JTAG interface of the target board via the ARM 20-pin or 10-pin 2.54mm pitch connector, is used to help download and debug the embedded software on the target hardware.

The following figure shows the connection of host PC, USB debug adapter and target board.

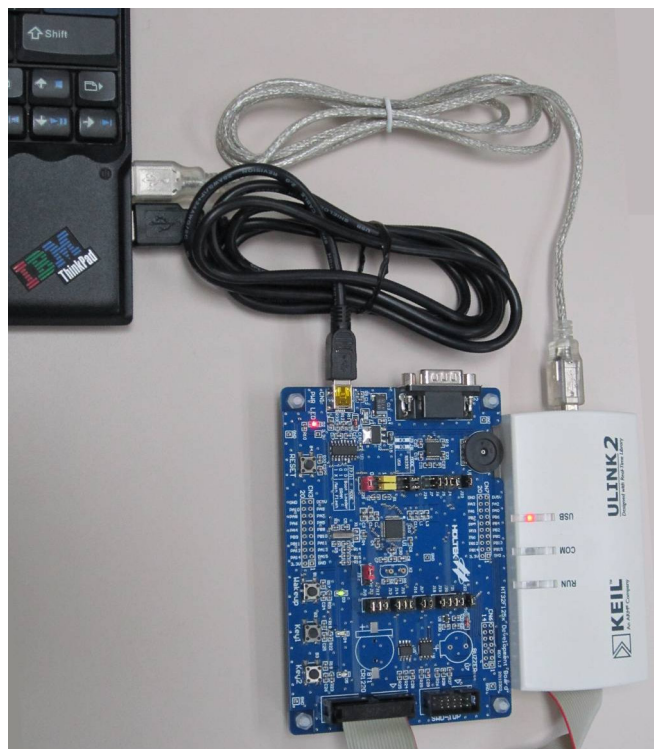


Figure 8. PC, USB Debug Adapter and Target Board Connection

6 Keil MDK-ARM Quick Start

The Keil™ MDK-ARM provides a complete “Keil™ μVision” development tool for project creation. The tool can edit both C and assembly code, set up the development tools, view the assembler code, connect and perform tests. Visit the Keil website <http://www.keil.com> for more information.

Create and Save New Project

To create a new project, follow the steps below:

1. Make sure that the Keil Holtek HT32 Support Package has been installed.
2. Double click on the “Keil μVision” shortcut or click “Start → All Programs → Keil μVision” to run the Keil μVision.
3. Choose “Project → New μVision Project...” to create a new Keil project in the menu.

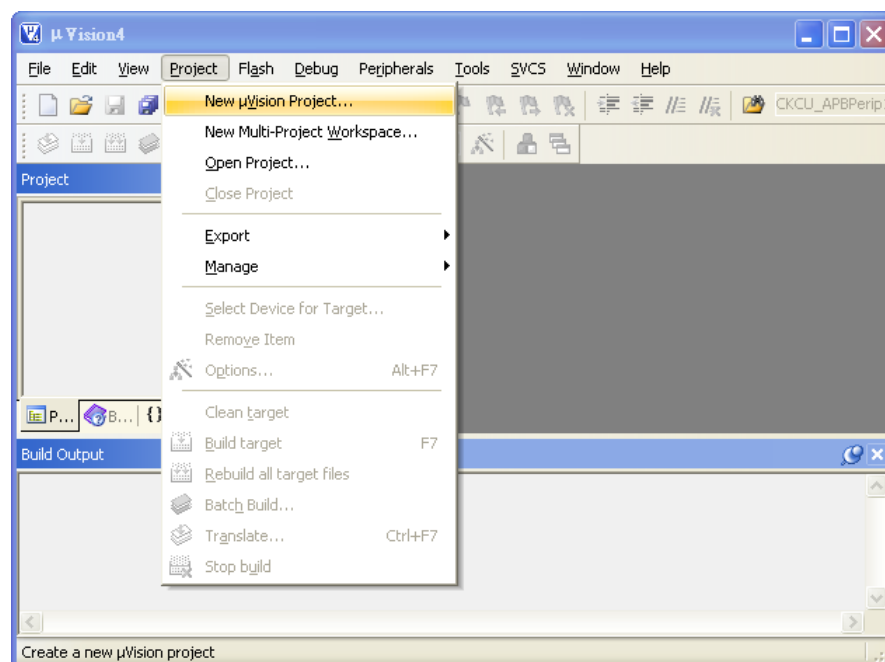


Figure 9. Create New Project

4. A “Create New Project” dialog will pop up. Specify the name and path of the project and press the “Save” button. The related information and files can then be found in the “Project” window.

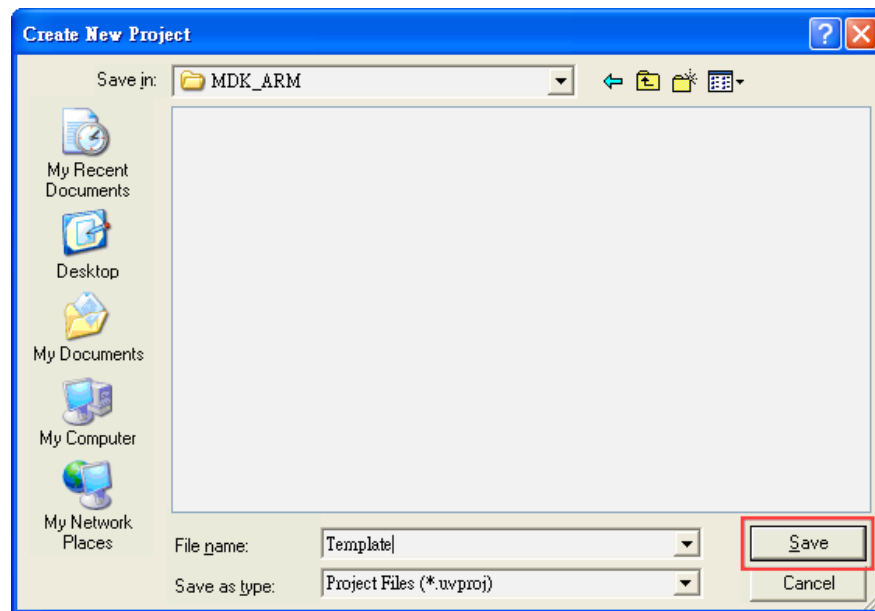


Figure 10. Create New Project Dialog

5. A “CPU Data Base File” dialog will pop up. Choose “Holtek HT32 Device Database” from the drop-down list.

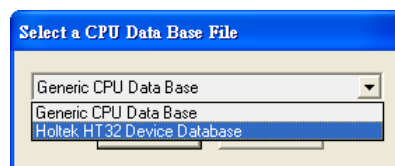


Figure 11. CPU Database Dialog

6. Select the device name. Here, the “HT32F1253” device is used as an example.

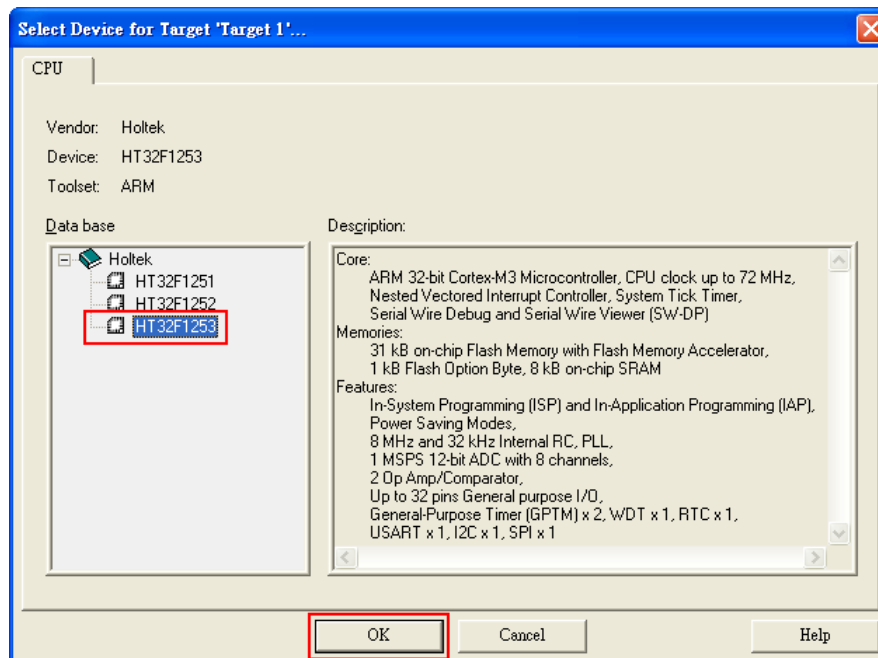


Figure 12. Select Project Device

Click “OK” button to complete the device selection. Decide whether to use the included startup code.



Figure 13. Startup Code Inquiry Window

Project Options Setup

The following section describes how to setup the project options including Device, USB debug adapter and Flash Loader.

Open Project Options Setup Page

1. There are two methods to enter the “Options for Target...” pages:
 - a. Right-click on the target name in the “Project” window to display the Context Menu and choose “Options for Target...” to open the option dialog.
 - b. Click the toolbar button “Options for Target”.

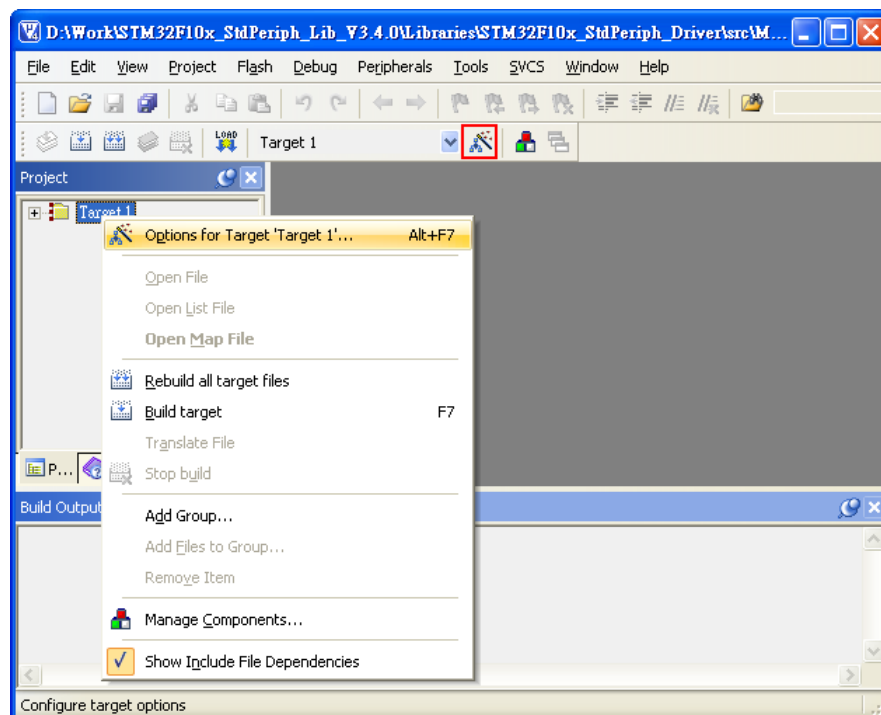


Figure 14. Target Menu Options

Device Selection

1. Open the “Options for Target...” dialog.
2. In the “Device” tab, choose “Holtek HT32 Device Database” from the “Database” drop-down list.
3. Select the device name. For example, “HT32F1253”.

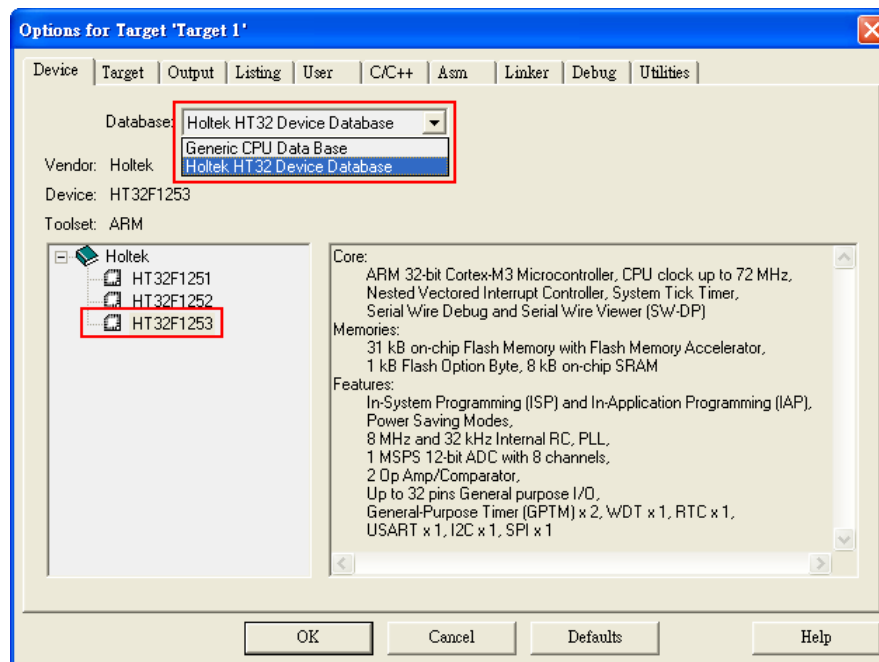


Figure 15. Select Project Device

Configure the USB Debug Adapter

The following shows the method of configuring the Keil µVision USB debug adapter. This example will take the ULINK2 as an example.

1. Connect the ULINK2 to the PC.
2. Open the “Options for Target...” dialog.
3. Click the “Debug” tab. The left side is for the simulator options while the right side is for the USB debug adapter options. Select “ULINK Cortex Debugger”.
4. Check “Load Application at Startup” and the application will then be pre-loaded into the “Debug mode”. If “Run to main()” is checked, the application will first be run to “main()” in the “Debug mode”.

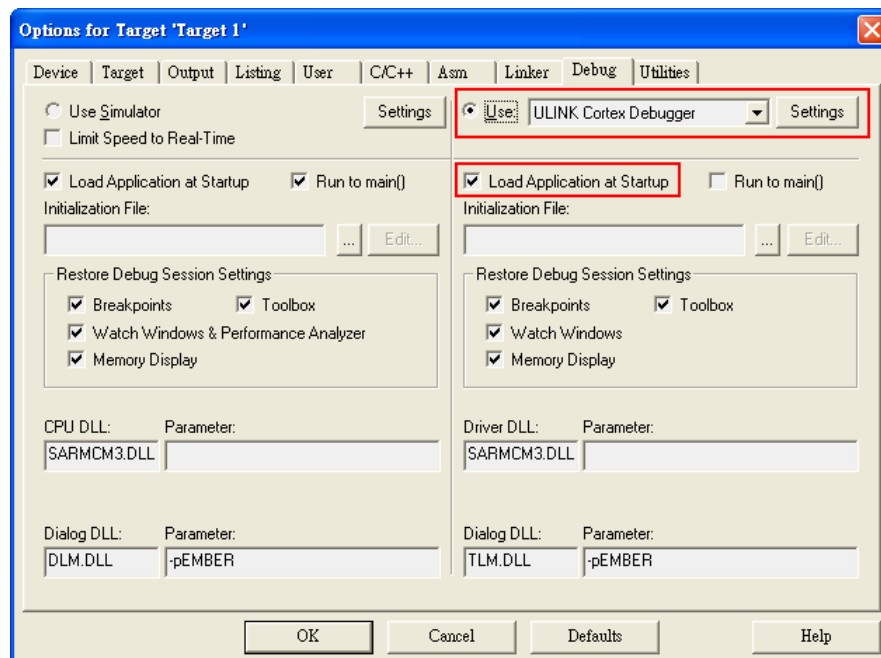


Figure 16. Target Debug Tab Page Options

5. Click the “Settings” button to open the “Cortex-M Target Driver Setup” dialog.
 - a. Select the debug adapter “Serial No” from the drop-down menu.
 - b. Select “Port” as “SW” or “JTAG”. Note that some HT32 devices only support the “SW” interface.
 - c. If “SWJ” is enabled, the ULINK2 will generate the required sequences on the Serial Wire / JTAG debug port (SWJ-DP) to switch between the JTAG and SWD interface. Therefore, enable “SWJ” on devices with SWJ-DP and disable “SWJ” on devices that have only a SW-DP or JTAG-DP interface.
 - d. For details of any other settings, refer to the documents on the Keil website.
http://www.keil.com/support/man/docs/ulink2/ulink2_ctx_debugdrivercfg.htm

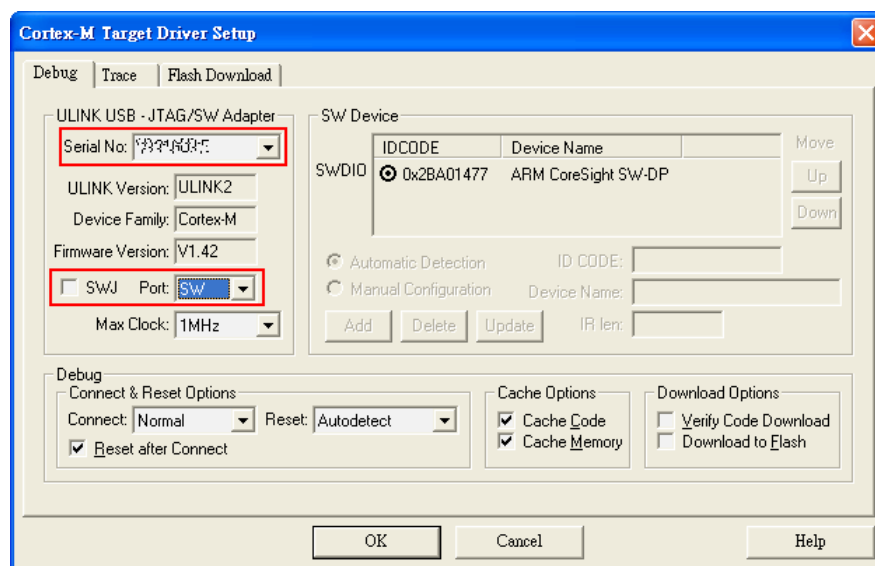


Figure 17. ARM Target Driver Setup Page

Flash Loader Setup

The Flash loader is used to download the program into the device flash memory. It is executed in the SRAM and receives data from the host PC, through the USB debug adapter. It then loads the data into the flash memory.

1. Open the “Options for Target” dialog.
2. Select the “Utilities” tab and open the Settings dialog. In this example, select the ULINK Cortex Debugger. Tick “Update Target before Debugging” to download the image into Flash memory automatically.

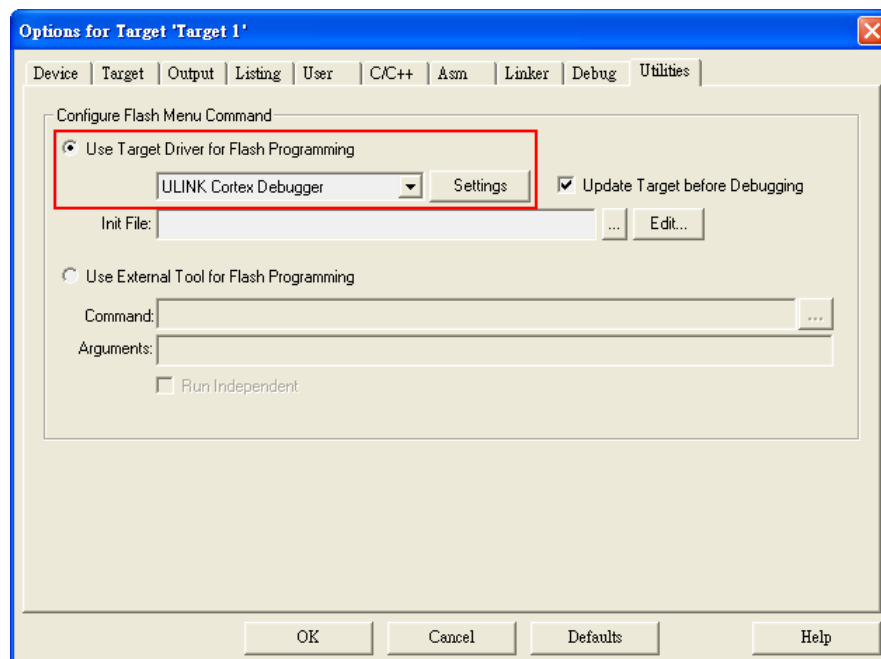


Figure 18. Utilities Tab Menu

3. Setup both the Programming Algorithm start position and memory size in this dialog. The start position is 0x20000000 and the size is 0x0800 (2 kB) for HT32 Series MCUs.
4. Click the “Add” button and choose “HT32 Series Flash” from the list box. To program the Option Bytes, choose the “HT32 Series Flash Options” in the same way.

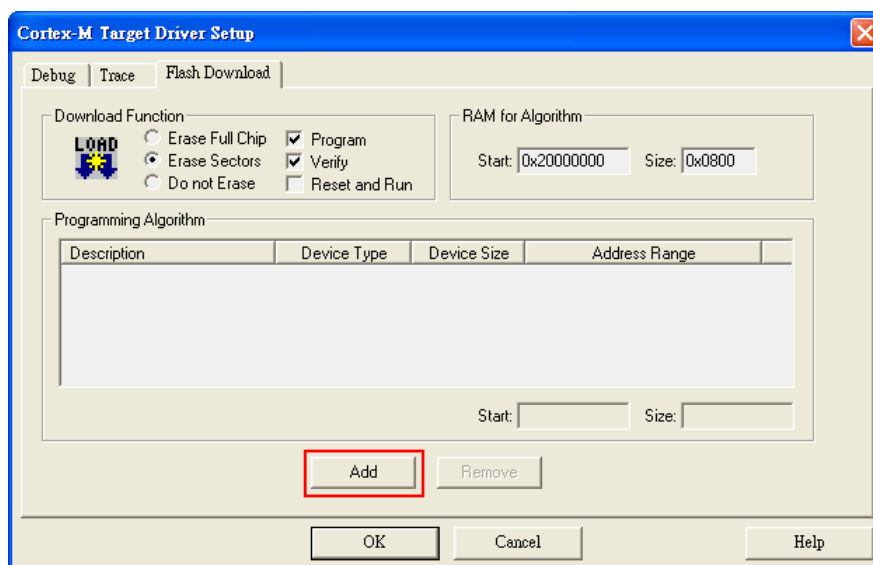


Figure 19. Target Driver Flash Download Tab

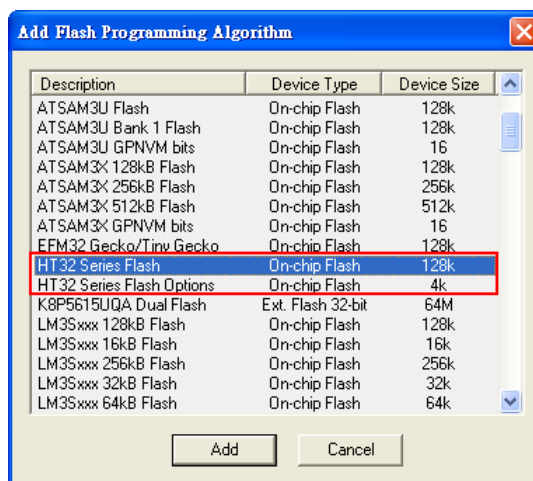


Figure 20. Flash Programming Algorithm Select

5. Once all the setups have finished, the HT32 Series Flash will be listed in the Programming Algorithm dialog and can therefore be programmed using the Keil µVision.

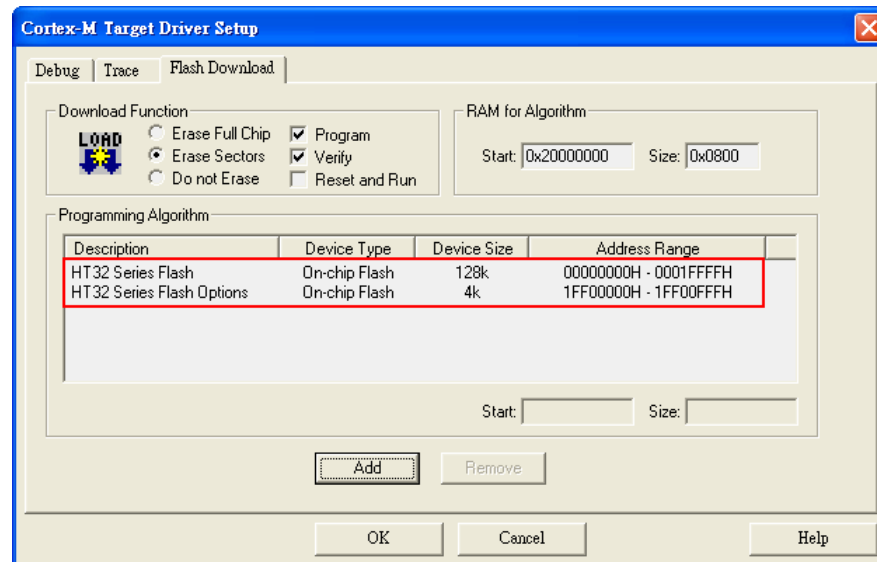


Figure 21. Flash Programming Algorithm

Adding Source Files to the Project

Source code can be added into the project using the following procedure:

1. Click the “New” icon in the toolbar or click “File → New...” to create a new file.

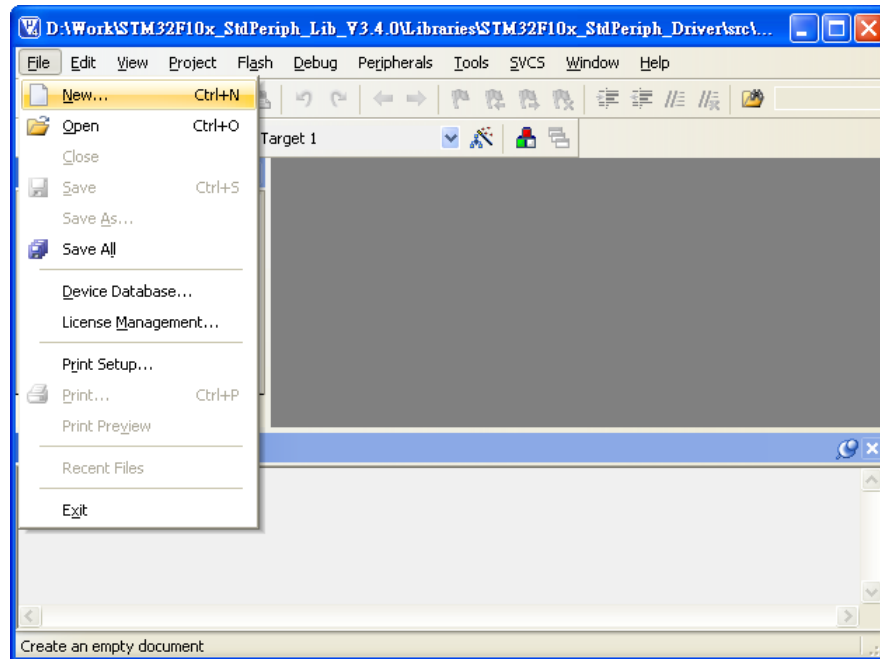


Figure 22. New File Creation

2. Edit the code shown below in the editor window. Click “File → Save” and save as “main.c”

```
int main(void)
{
    int i = 0;
    int j = 0x20001000;
    while(1)
    {
        *(int *)j = i;
        i++;
    }
}
```

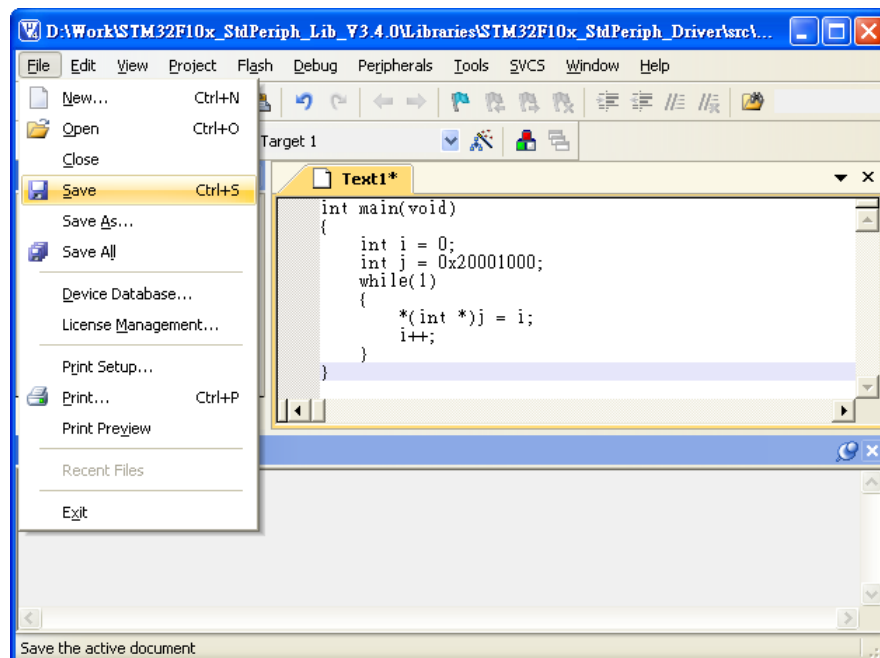


Figure 23. Save File

3. Right click on the “Source Group” and select “Add Files to Group...” to add the “main.c” file into the project.

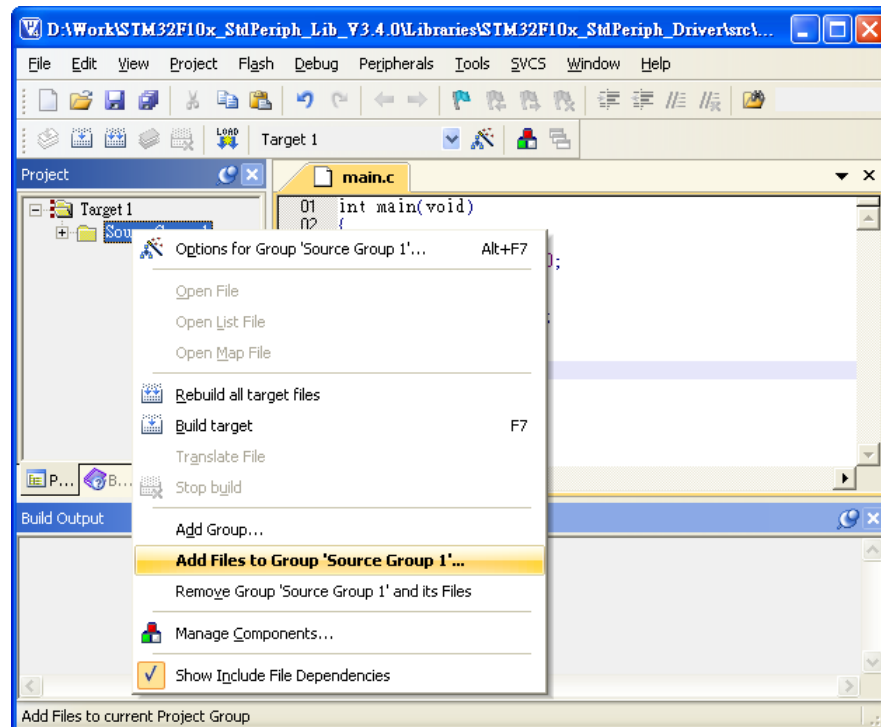


Figure 24. Add File to Group

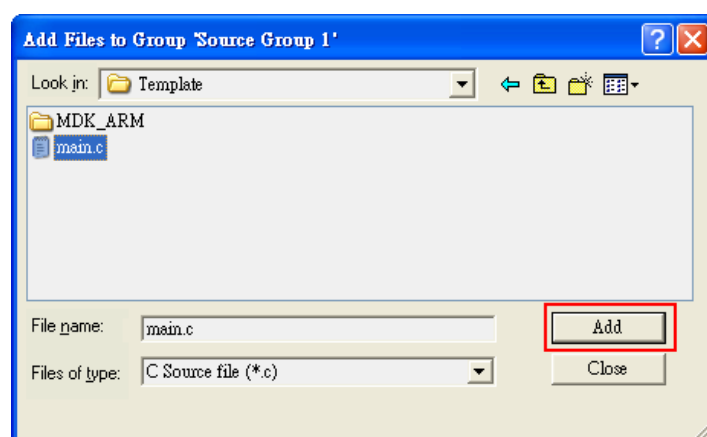


Figure 25. Add File to Group – Specify File Name and Path

4. Select the “Components, Environment, and Books” function to rename, add files, re-order and so on for the project source files.

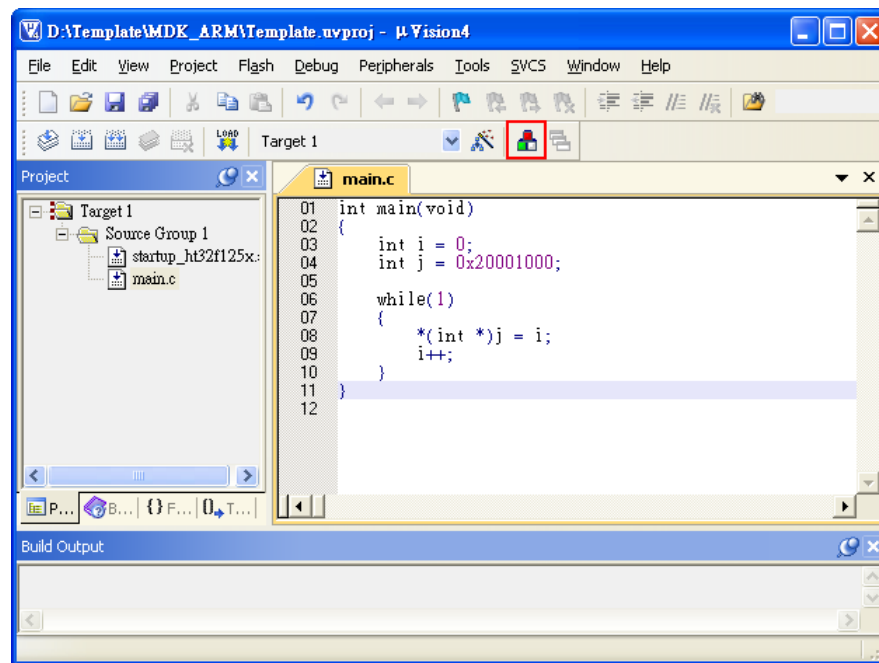


Figure 26. Components, Environment and Books Function

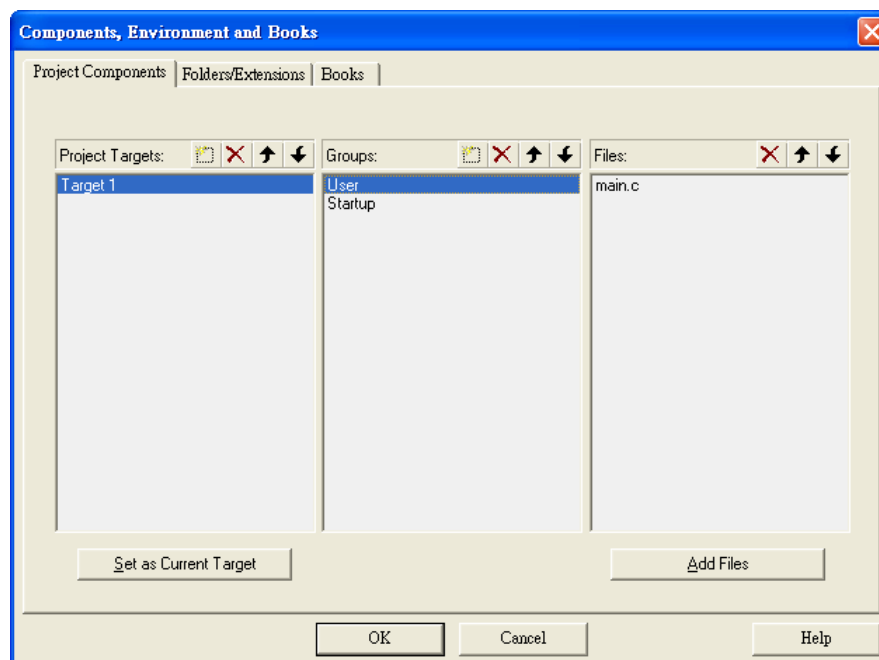


Figure 27. Modify Project Components

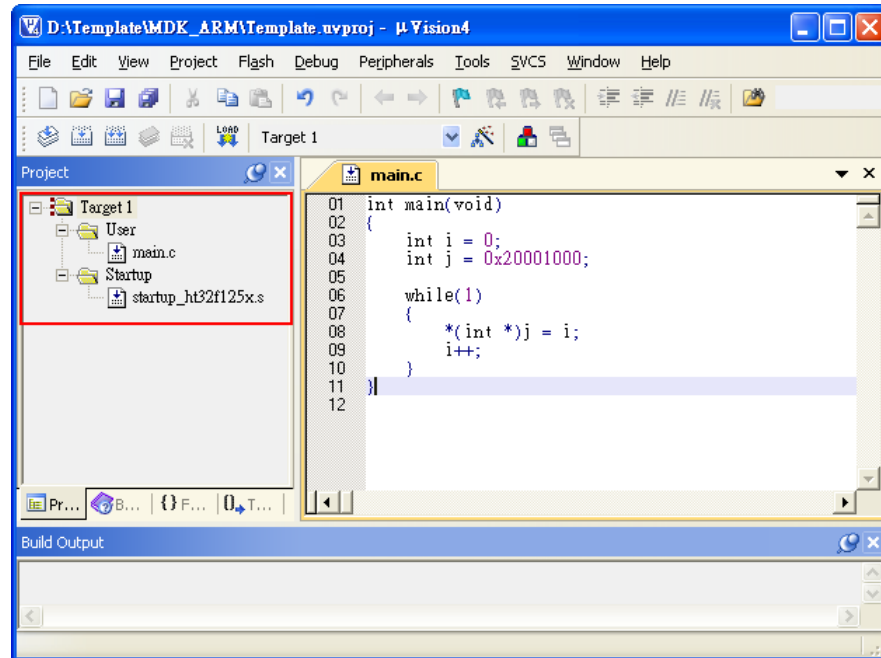


Figure 28. Project Component List

7 Compiling the Project

Use the following procedure to compile the project:

1. Choose “Project → Rebuild all target files” to recompile all the files in the project.

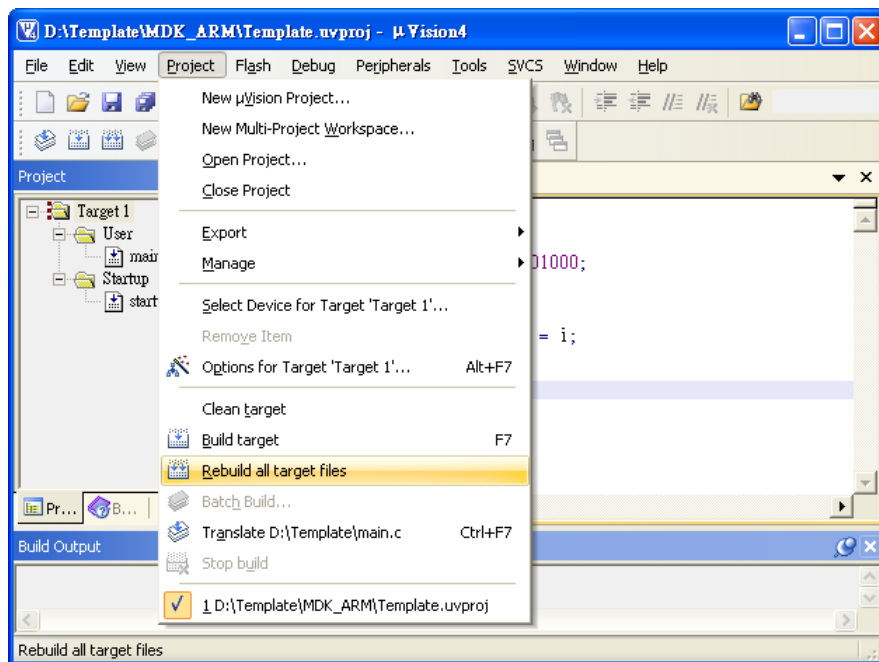


Figure 29. Build All target Files

2. Check the “Build Output” message to confirm if the project has been successfully built and linked.

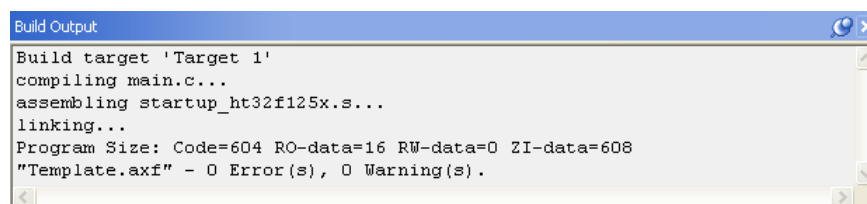


Figure 30. Compiler Messages

8 Download and Debug

The following section shows how to download the application and use the debug features such as free running, break, single step and breakpoint.

Enter/Exit Debug Mode

1. Click “Debug → Start/Stop Debug Session (Ctrl + F5)” or the “Debug” icon to enter the debug mode. The image will be downloaded into Flash memory automatically when the “Update Target before Debugging” option is enabled.

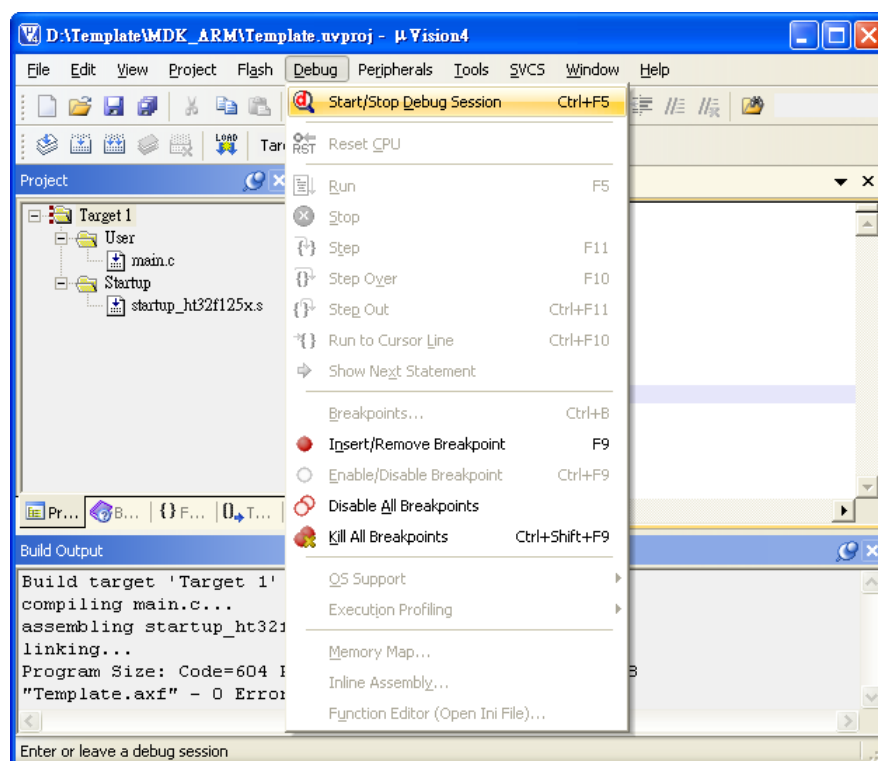


Figure 31. Start Debugging Session

2. After the debugging is finished, click “Debug → Start/Stop Debug Session (Ctrl + F5)” or the “Debug” icon again to exit the debug mode.
3. The debug window as shown below will appear which will display information regarding “Registers”, “Disassembly”, “Source Code Window”, “Memory Window”, “Command” and so on.

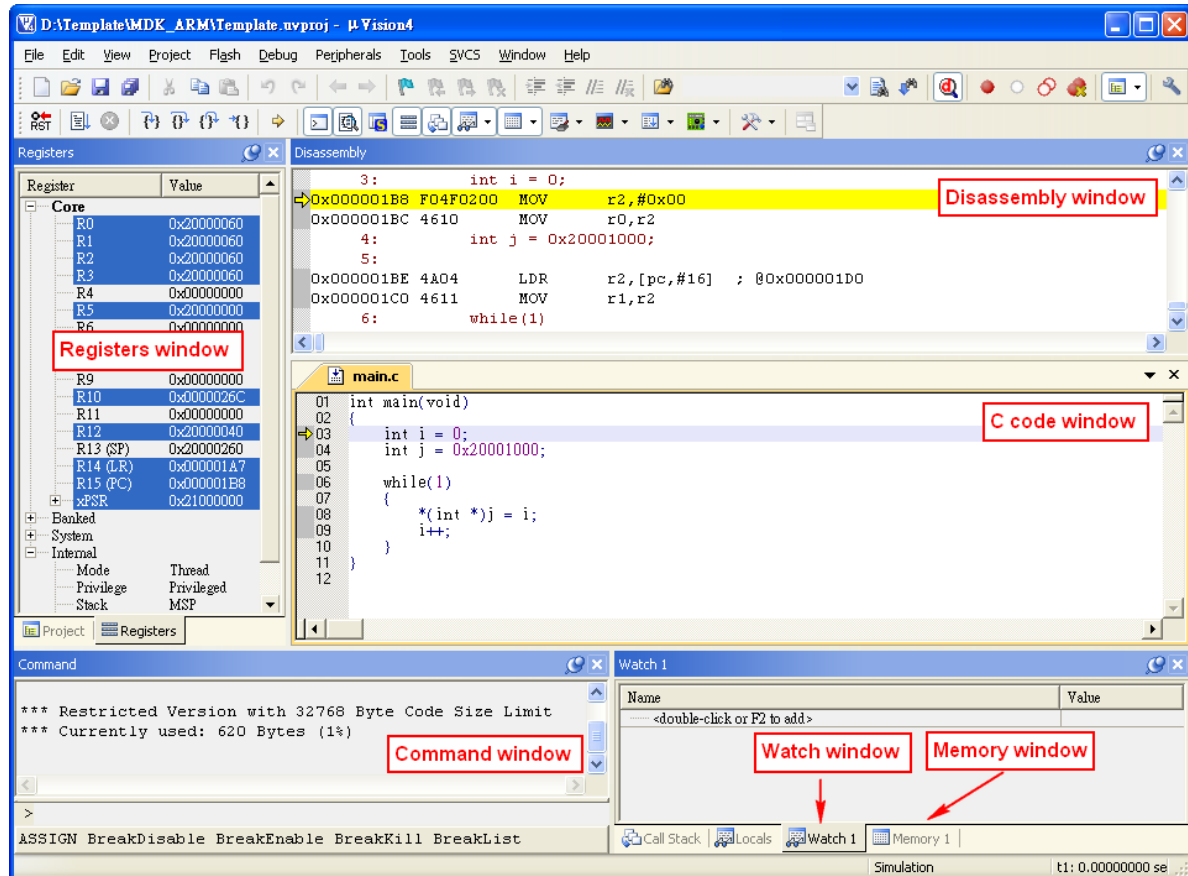


Figure 32. Debugging Session

Free Running, Break, and Single Step Operation

The debugger provides Free Running, Break and Single Step functions to assist with application debug.

1. Click the “Run” icon to allow the program to free run.

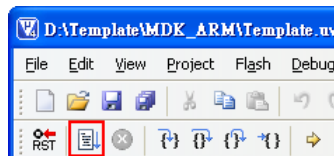


Figure 33. Run Icon

2. Click the “Stop” icon to stop free running.

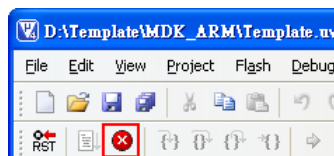


Figure 34. Stop Icon

3. Click the “Step” icon or “F11” to single step the program. Use this feature to debug the program step by step.

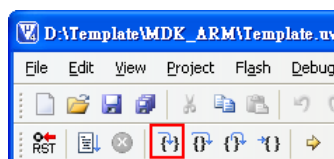


Figure 35. Step Icon

Breakpoints

Breakpoints can be setup to halt the program at user specified program locations to allow program status examination at these user defined locations. The following procedure shows how this is implemented:

1. Before the program starts running double click on the desired breakpoint program line to set the breakpoint. A red point mark will then be displayed next to the breakpoint line. Multiple breakpoints can be setup using this method.

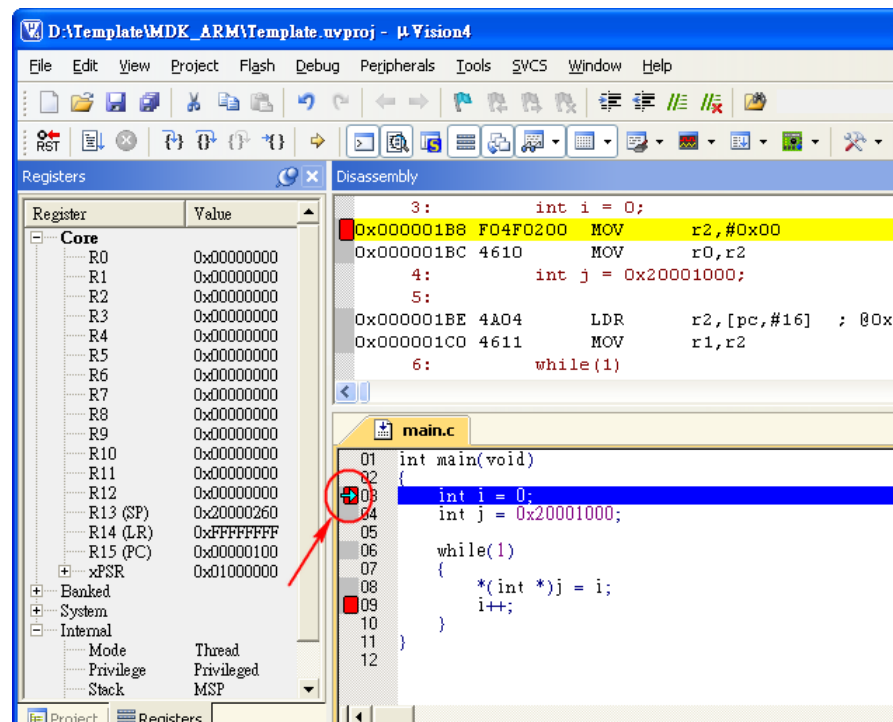


Figure 36. Breakpoint Setup

- Click the “Run” icon to start the program. The program will run normally until it encounters the first breakpoint where it will halt execution. A yellow arrow will indicate the present program execution location.

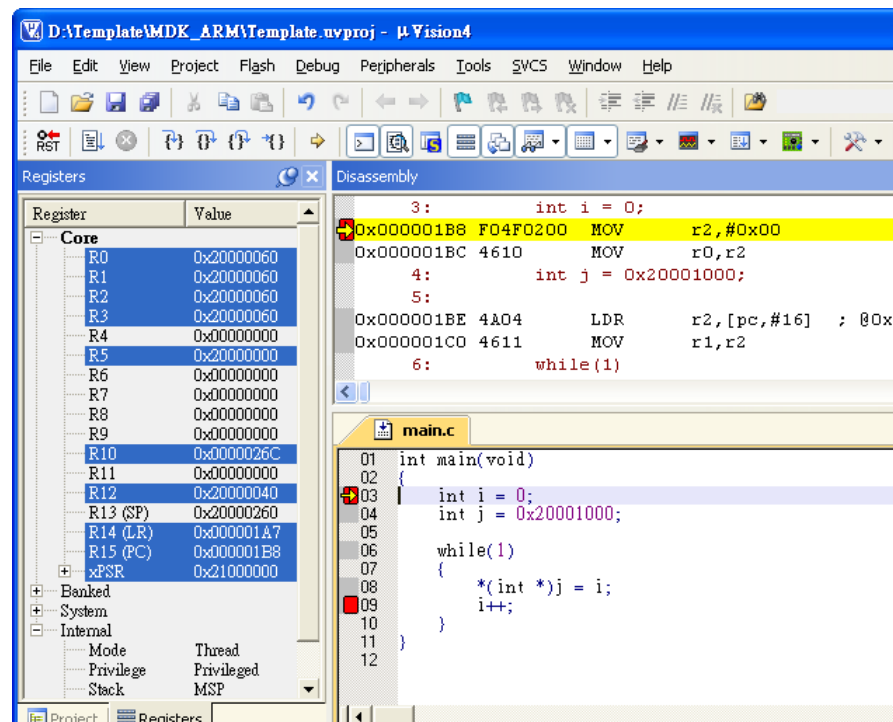


Figure 37. Breakpoint Program Stop

Memory Window

A function is included to examine user selected memory contents.

1. Click “View → Memory Windows → Memory 1” to open the memory window.

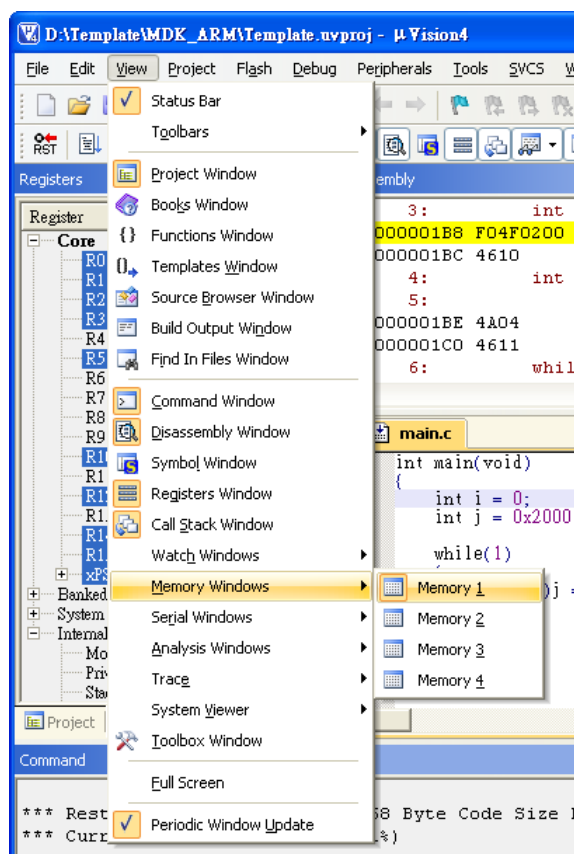


Figure 38. Open Memory Window

2. Type “0x20001000” into the “Address” input box to view the memory contents.

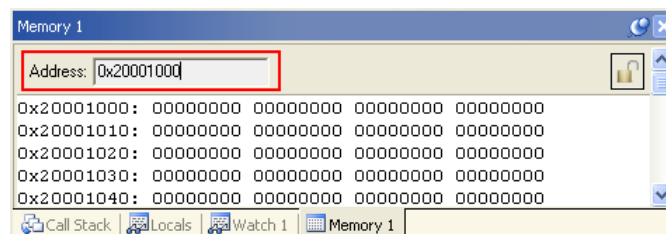


Figure 39. Address 0x20001000 Memory Contents Window

3. Click the “Step” icon or “F11” to single step the program. Use this to examine consecutive memory contents.

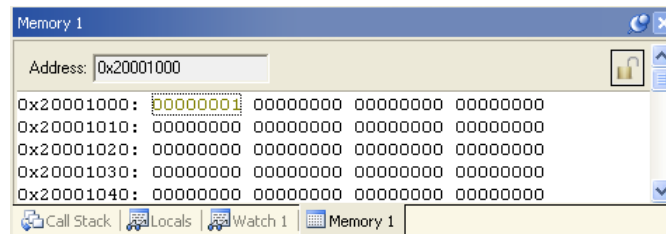


Figure 40. Memory Content Examination

9 Mass Erase

Mass erase is an operation that erases the whole flash memory including the main flash memory and the Option Bytes. It can be used to clear all the data in the flash memory or to disable the security settings.

1. In the μ Vision window, Click “Flash → Erase”

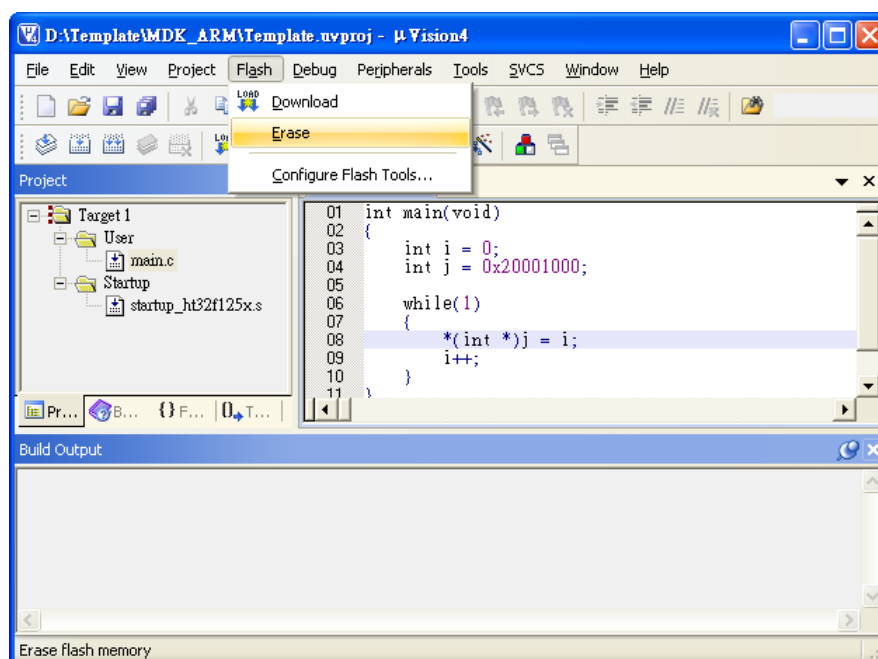


Figure 41. Mass Erase Function

2. After a mass erase operation has completed, a “Build Output” message window will display a message to notify that the flash memory has been successfully erased.

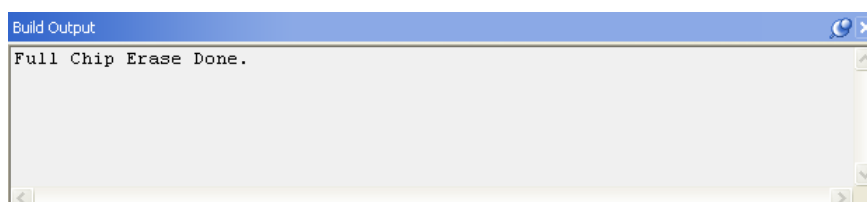


Figure 42. Build Output Message Window

10 Option Bytes Programming

Option Byte Programming provides protection for flash page erase/programming and for enhanced device security.

1. Copy “ht32f125x_op.s” to the project folder. The original “ht32f125x_op.s” file is located in the Keil installation folder: “{KEIL_PATH}\ARM\Startup\Holtek”.

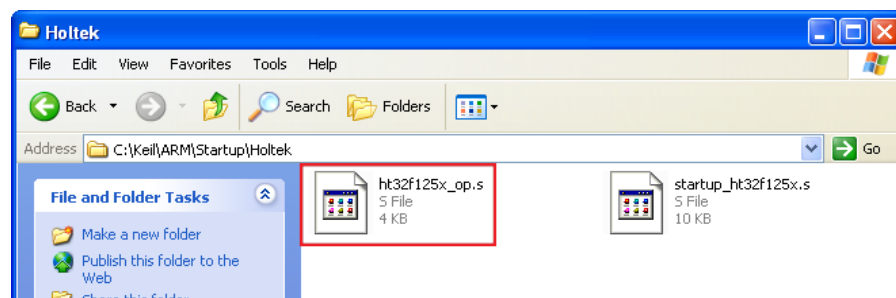


Figure 43. Option Bytes Setup File

2. Right click on the item in the Project window, then select “Add Files to Group” to add the Option Byte setup file.

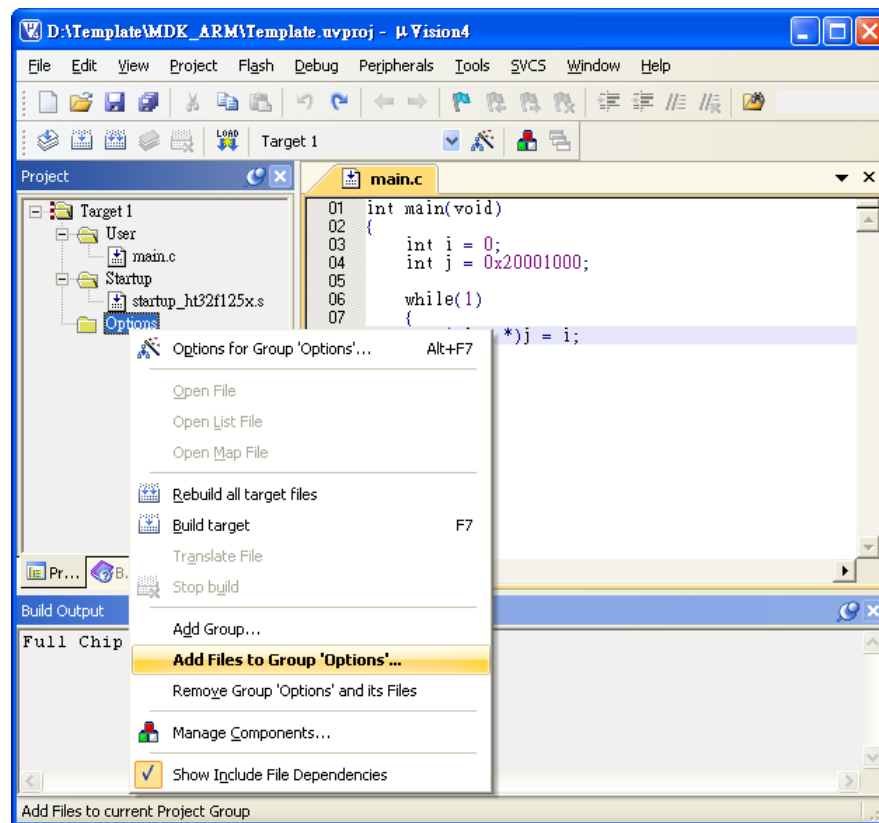


Figure 44. Add Option Bytes File

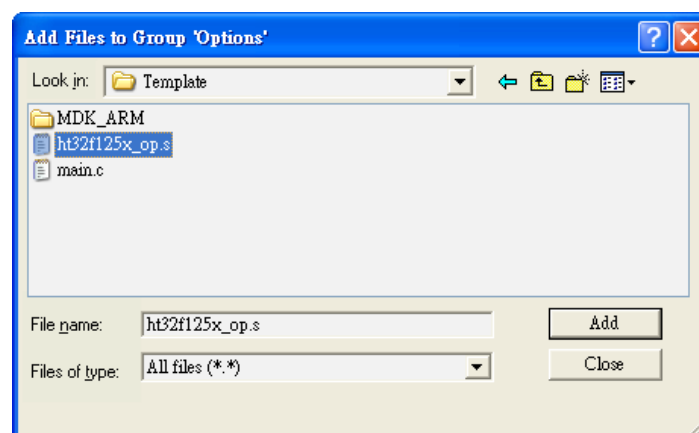


Figure 45. Add Files to Group Dialog

3. Select the Option Byte file and change the values using the “Configuration Wizard” according to actual requirements.

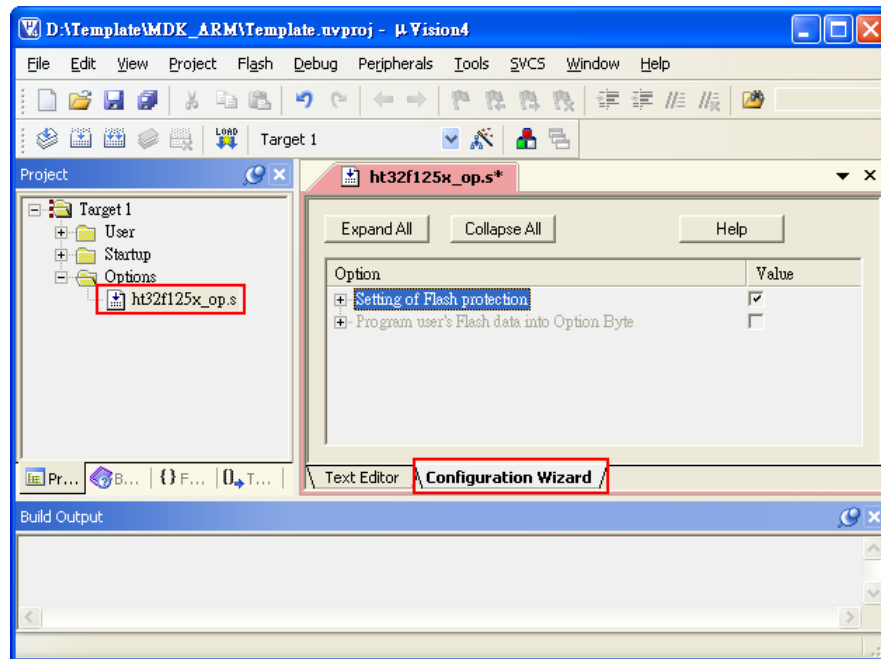


Figure 46. Option Bytes Configuration Wizard

4. Rebuild all the files once the Option Bytes setup is complete. These Option Bytes will be programmed into the HT32 series MCU automatically when the download procedure starts.
NOTE: To program the Option Bytes, confirm that the “HT32 Series Flash Options” are listed in the Programming Algorithm dialog. Refer to the Flash Loader Setup section for more information.

11

Using the CodeSourcery GNU Toolchain with Keil µVision

CodeSourcery have developed a GNU Toolchain for ARM processors and provide regular, validated releases of the GNU Toolchain. The Sourcery G++ Lite Edition supports ARM, Thumb, and Thumb-2 compilation for all architectures, including the ARMv7.

This section describes how to configure the Keil™ µVision and install the Sourcery G++ Lite Edition to use the CodeSourcery GNU ARM tool chain.

Installing the Sourcery G++ Lite Edition

The Sourcery G++ Lite Edition contains free and command-line only versions of the core development tools. The latest version can be downloaded from the following website:

http://www.codesourcery.com/gnu_toolchains/arm/download.html

For general cases, the EABI version for the non-OS applications of Cortex-M3 can be used. For more information about the setup process, refer to the “Getting Started Guide” document from the CodeSourcery website.

(<http://www.codesourcery.com/sgpp/lite/arm/portal/doc9876/getting-started.pdf>).

Configuring Keil µVision for Sourcery G++

The following contents assume that a project has been created. For more information regarding creating a project, refer to Chapter 6 Keil MDK-ARM Quick Start. Note that the following steps will reset all the options to their default value. It is important to first backup the project.

- ## Using the CodeSourcery GNU Toolchain with Keil μ Vision



- Click the “Folders/Extensions” tab, change the setting as shown in Figure 48. If “Use GNU Compiler” is checked, a warning message will be displayed to notify that all the options in the project will be reset to their default values. Reconfirm that the project has already been backed up and press “Yes” to continue. Select or modify the installed paths of μ Vision or Sourcery G++ if required.

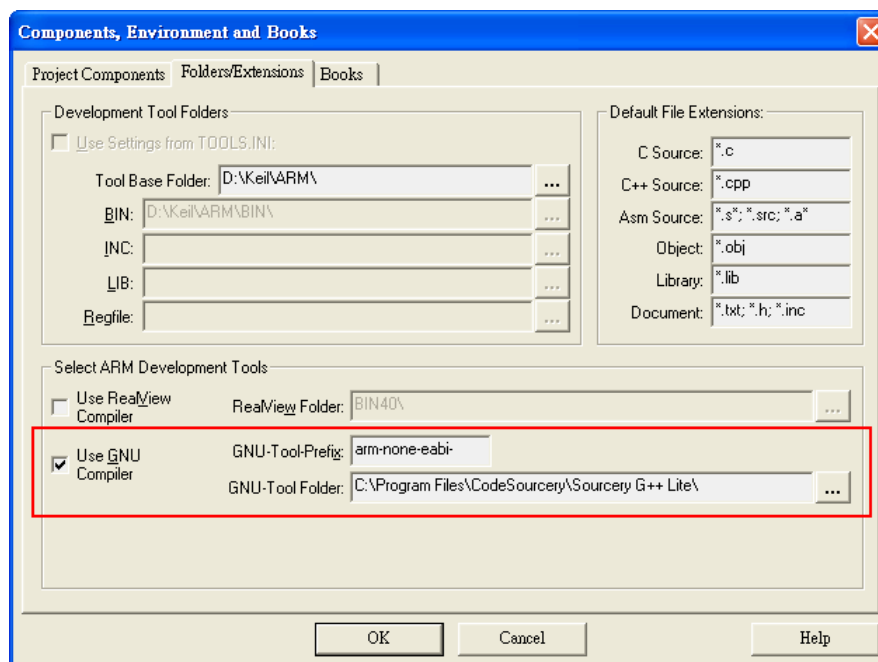


Figure 48. Folders/Extensions Setup Change

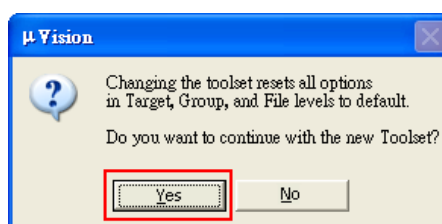


Figure 49. “Use GNU Compiler” Warning Message

3. Refer to Section Project Options Setup to setup the project options such as USB adapter or Flash loader. Additionally, modify also the Linker settings as shown in the accompanying diagram. The linker script file can be found in the example code of the HT32 Keil quick start guide, obtainable from the Holtek website. Path: \\Quick_Start_Example_Keil\\SourceryG++Lite\\

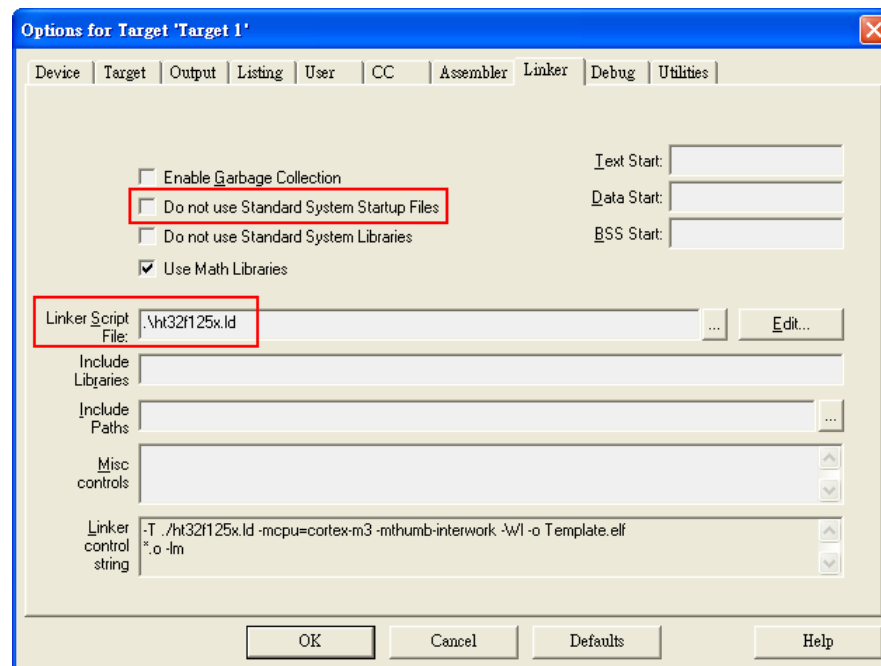


Figure 50. Linker Options

4. In the “CC” tab, tick “Compile Thumb Code” for the Cortex-M3.

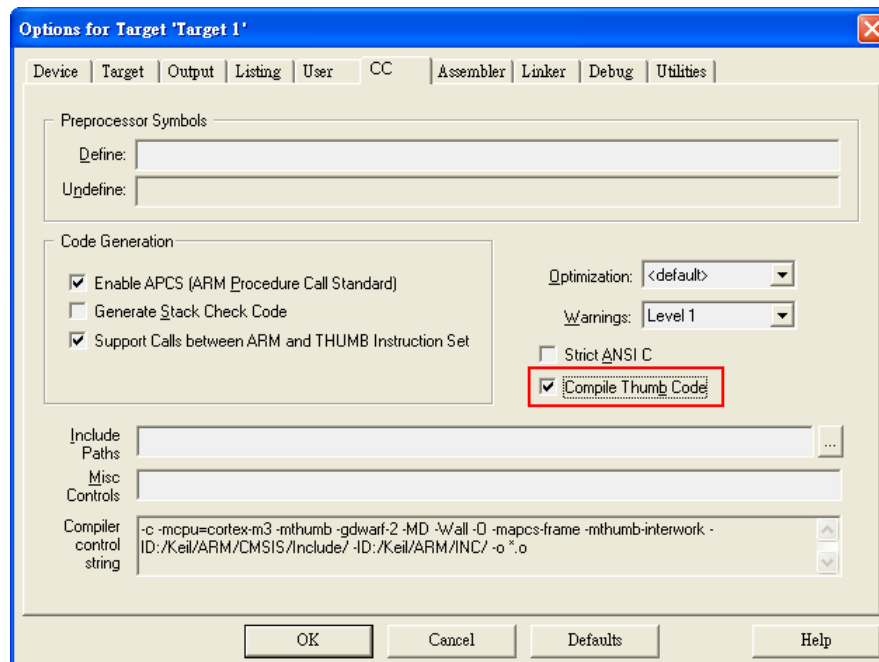


Figure 51. C Compiler Options

5. The startup file should be changed for the GNU Toolchain. For example, remove the original “startup_ht32f125x.s” and add the new one to the project. The startup file for Sourcery G++ can be found in the example code of the HT32 Keil quick start guide, obtainable from the Holtek website.
Path: \\Quick_Start_Example_Keil\\SourceryG++Lite\\

6. Choose “Project → Rebuild all target files” to recompile all the files in the project. Check the “Build Output” message to confirm if the project has been successfully built and linked.

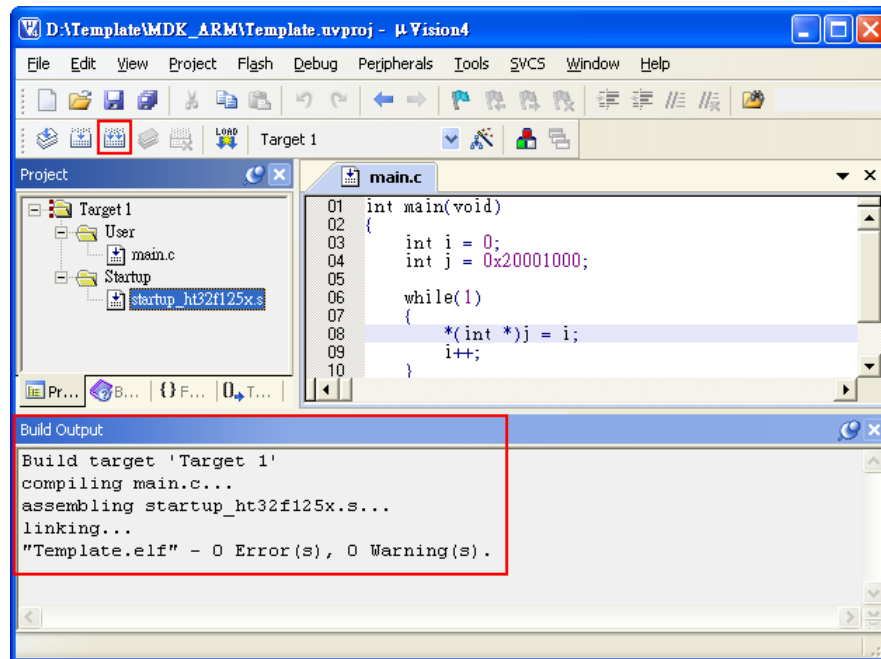


Figure 52. GNU Toolchain Compiler Messages

12 Conclusion

All the necessary items for developing embedded software such as Keil MDK-ARM, USB debug adapter, software project, debugging environment have been provided in this document. A brief introduction to the creation, management, building and debugging of software has also been provided to help get started with software development using the Keil MDK-ARM for Holtek 's HT32 series microcontrollers.

To help users start creating their own HT32 series applications, Holtek provides a related firmware library, example code, documents and other services to reduce the user development cycle time. Additional further technical support such as application notes etc. can be obtained from the Holtek website.

Holtek Semiconductor Inc. (Headquarters)

No.3, Creation Rd. II, Science Park, Hsinchu, Taiwan
Tel: 886-3-563-1999
Fax: 886-3-563-1189
<http://www.holtek.com.tw>

Holtek Semiconductor Inc. (Taipei Sales Office)

4F-2, No. 3-2, YuanQu St., Nankang Software Park, Taipei 115, Taiwan
Tel: 886-2-2655-7070
Fax: 886-2-2655-7373
Fax: 886-2-2655-7383 (International sales hotline)

Holtek Semiconductor Inc. (Shenzhen Sales Office)

5F, Unit A, Productivity Building, No.5 Gaoxin M 2nd Road, Nanshan District, Shenzhen, China 518057
Tel: 86-755-8616-9908, 86-755-8616-9308
Fax: 86-755-8616-9722

Holtek Semiconductor (USA), Inc. (North America Sales Office)

46729 Fremont Blvd., Fremont, CA 94538, USA
Tel: 1-510-252-9880
Fax: 1-510-252-9885
<http://www.holtek.com>

Copyright© 2011 by HOLTEK SEMICONDUCTOR INC.

The information appearing in this document is believed to be accurate at the time of publication. However, Holtek assumes no responsibility arising from the use of the specifications described. The applications mentioned herein are used solely for the purpose of illustration and Holtek makes no warranty or representation that such applications will be suitable without further modification, nor recommends the use of its products for application that may present a risk to human life due to malfunction or otherwise. Holtek's products are not authorized for use as critical components in life support devices or systems. Holtek reserves the right to alter its products without prior notification. For the most up-to-date information, please visit our web site at <http://www.holtek.com.tw>.