IAR EWARM 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 IAR EWARM 6
2 System Requirements 7
3 Software Installation 8
IAR EWARM Installation 8
Installing the IAR HT32 Support Package (For EWARM v6.20 or Below) 8
4 Installing the USB Debug Adapter 12
5 Connecting to the Target Board 13
6 IAR EWARM Quick Start 14
Create and Save New Project 14
Project Options Setup 17
Open Project Options Setup Page 17
Device Selection 19
Configure the USB Debug Adapter 20
Flash Loader Setup 22
Adding Source Files to the Project 23
7 Compiling the Project 26
8 Download and Debug 27
Enter/Exit Debug Mode 27
Free Running, Break, and Single Step Operation 28
Breakpoints 29
Memory Window 31
9 Mass Erase 32
10 Option Bytes Programming 36
11 Conclusion 40

List of Figures

Figure 1. Software Development Flow 5
Figure 2. IAR EWARM Software Development Environment 6
Figure 3. System Architecture and Requirements 7
Figure 4. Support Package Installation 8
Figure 5. Support Package Installation – Ready to Install 9
Figure 6. Support Package Installation – Version Selection 10
Figure 7. Support Package Installation – Completion 11
Figure 8. PC, USB Debug Adapter and Target Board Connection 13
Figure 9. Create New Project 14
Figure 10. Create New Project 15
Figure 11. Save Project 15
Figure 12. Save Workspace 16
Figure 13. Save Workspace 16
Figure 14. Open Options 17
Figure 15. Open Options 18
Figure 16. General Options Setup 19
Figure 17. Debugger Driver Select 20
Figure 18. J-Link/J-Trace Interface Setup 21
Figure 19. Tick Flash Loader 22
Figure 20. New File Creation 23
Figure 21. Save File 24
Figure 22. Add File to Workspace 25
Figure 23. Successful Addition of File to Workspace 25
Figure 24. Project Building 26
Figure 25. Compiler Messages 26
Figure 26. Download and Debug Button 27
Figure 27. Stop Debugging Icon 27
Figure 28. Debugger Main Window 27
Figure 29. Go Icon 28
Figure 30. Break Icon 28
Figure 31. Step Into Icon 28
Figure 32. Breakpoint Setup 29
Figure 33. Breakpoint Program Stop 30
Figure 34. Open Memory Window 31
Figure 35. Address 0x20001000 Memory Contents Window 31
Figure 36. Memory Content Examination 31
Figure 37. Add Files 32
Figure 38. Macro File Setup 33
Figure 39. Download and Debug Button 34
Figure 40. Memory Window 34
Figure 41. Quick Watch Window 34
Figure 42. Warning Message Box 35
Figure 43. Debug Log Message Window 35

Figure 44. Memory to be Mass Erased 35
Figure 45. Add Files 36
Figure 46. Macro File Setup 37
Figure 47. FlashMarco.mac 37
Figure 48. Download and Debug Button 38
Figure 49. Option Byte Memory Window Display 38
Figure 50. Quick Watch Window Key-in Function 38
Figure 51. Option Byte Programmed Successfully 39
Figure 52. Option Byte Successful Update 39

1 Introduction

The purpose of this quick start guide is to familiarise users with the IAR Embedded Workbench for ARM (EWARM) 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 IAR EWARM as well as a guide for installing the HT32 support package for the IAR. Necessary information is also provided about using the IAR Embedded Workbench IDE (Integrated Development Environment) to compile and run software projects.

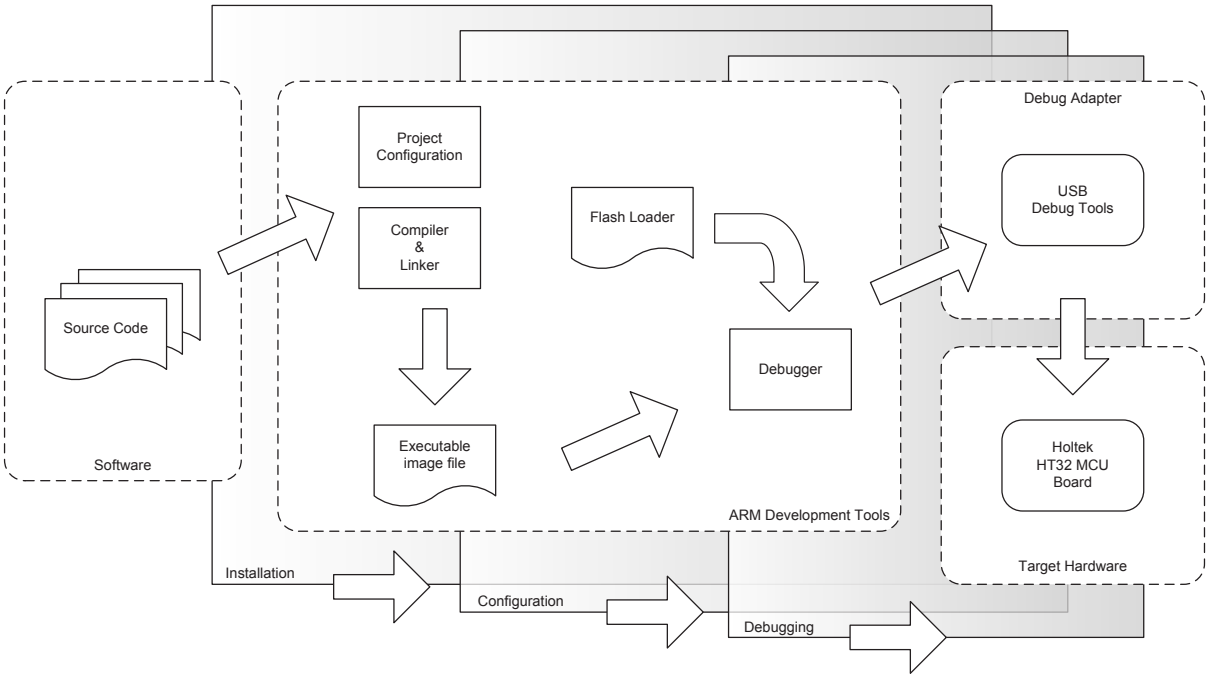
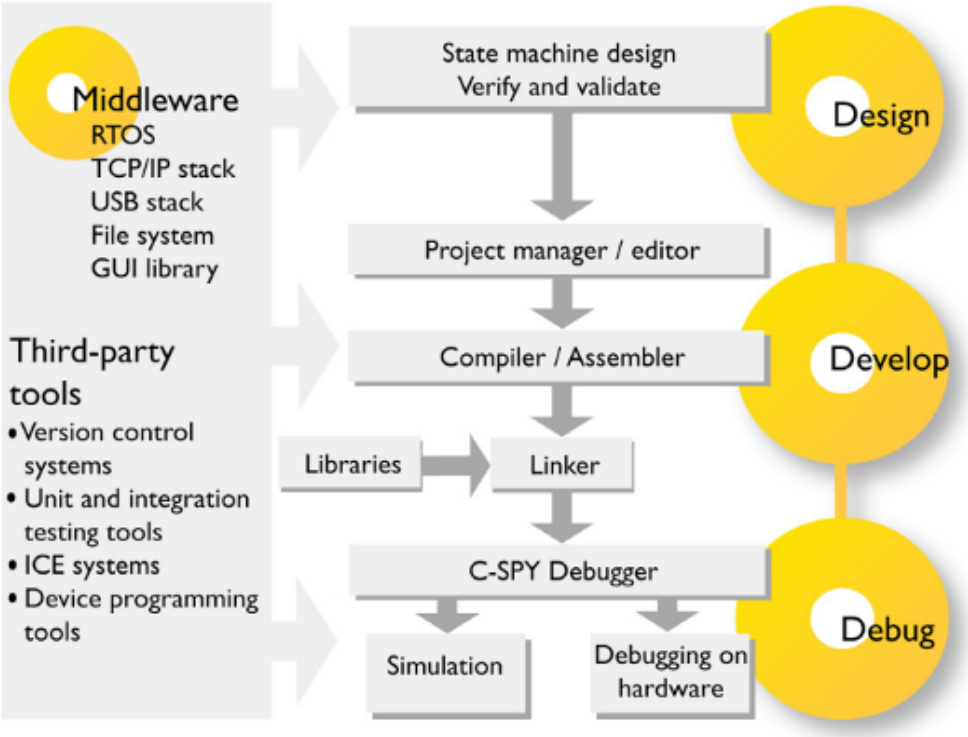


Figure 1. Software Development Flow

About the IAR EWARM

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

- Supports Cortex-M series, Cortex-R4, ARM7 and ARM9 devices
- IAR Embedded Workbench IDE
  - Configuration files for all supported devices
  - Editor, project management and compilation
  - Debugger environment including trace and analysis tools
  - Simulation environment
- IAR C/C++ Compiler, Assembler, Linker and librarian tools
- Run-time libraries
- Support for RTOS-aware debugging on hardware
- Flash Loader for Flash memory programming
- Ready-made code and project examples for supported evaluation boards



NOTE: The above figure was extracted from the document “Getting Started with IAR Embedded Workbench”.

Figure 2. IAR EWARM 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 J-Link or Holtek's e-Link32
- A host computer running Microsoft® Windows® XP, Vista or Windows® 7
  - A recommended 1 GB RAM and 2 GB of available hard-disk space
  - XGA (1024x768) colour monitor or higher resolution display
  - Mouse or other pointing device
  - A CD-ROM drive (optional)
- IAR EWARM V5.40 or above

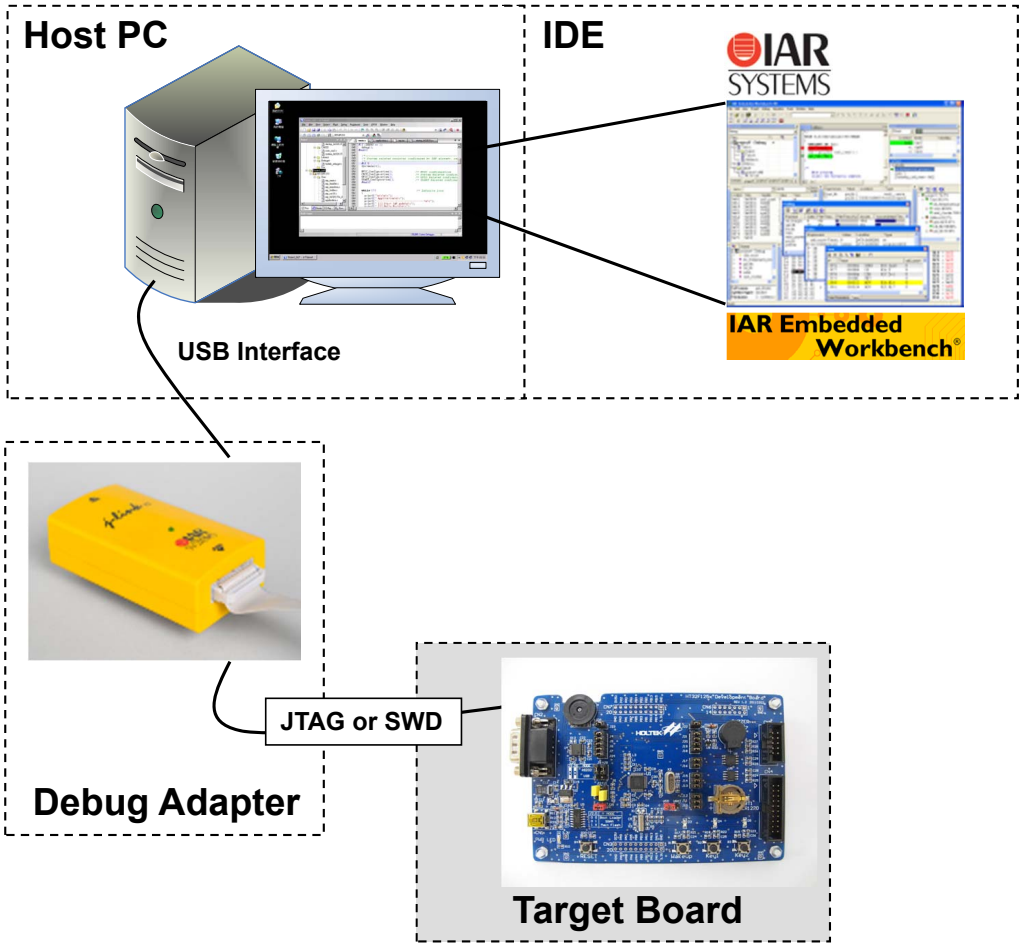


Figure 3. System Architecture and Requirements

3 Software Installation

IAR EWARM Installation

The KickStart version of the IAR EWARM can be downloaded from http://www.iar.com or from a CD-ROM provided by Holtek. The latest version of the EWARM is regularly available at http://www.iar.com/ewarm. Note that the KickStart version has a limitation of 32 KB of image size. For more information about the setup process, refer to the "Installation and Licensing Guide" document from IAR's website (ftp://ftp.iar.se/WWWfiles/guides/InstallationGuide.pdf).

Installing the IAR HT32 Support Package (For EWARM v6.20 or Below)

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

The following steps show how to install the support package into the IAR EWARM.

Step 1: Obtain the latest version of Support Package from the Holtek website or from the CD-ROM provided by Holtek. The filename is "HT32_IAR_Package_vnnn.exe" where "nnn" represents the version number.

Step 2. Execute the support package installation program by double-clicking on "HT32_IAR_Package_vnnn.exe". Press the "Next" button to continue when the screen below appears.

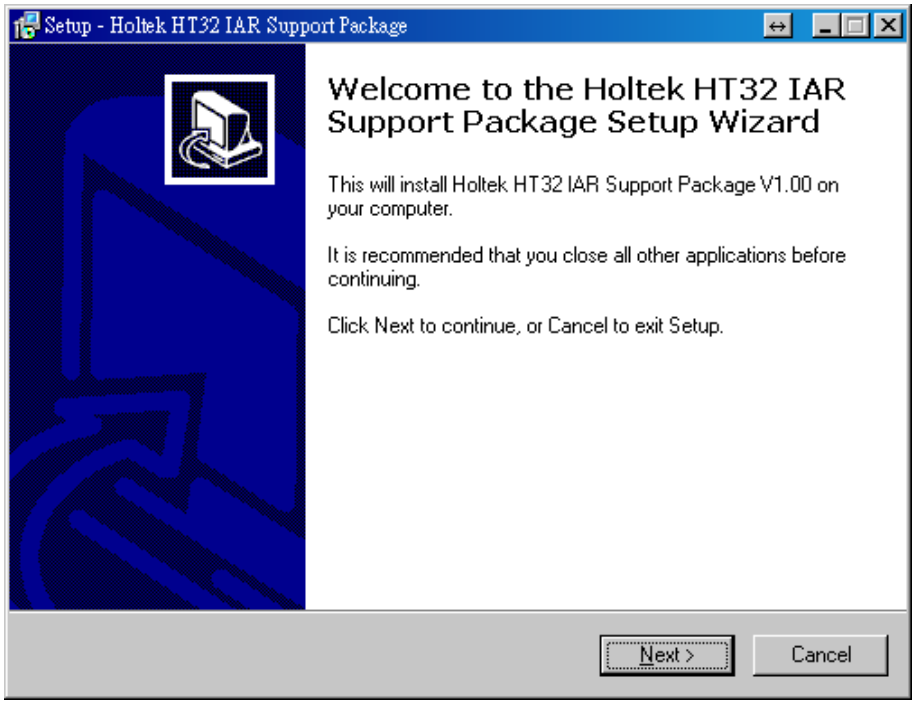


Figure 4. Support Package Installation

Step 3: The support package will detect the last installed path of IAR EWARM automatically. If only a single version of IAR EWARM has been installed on the system, the ready-to-install page will appear directly as shown below. Press "Install" button to continue.

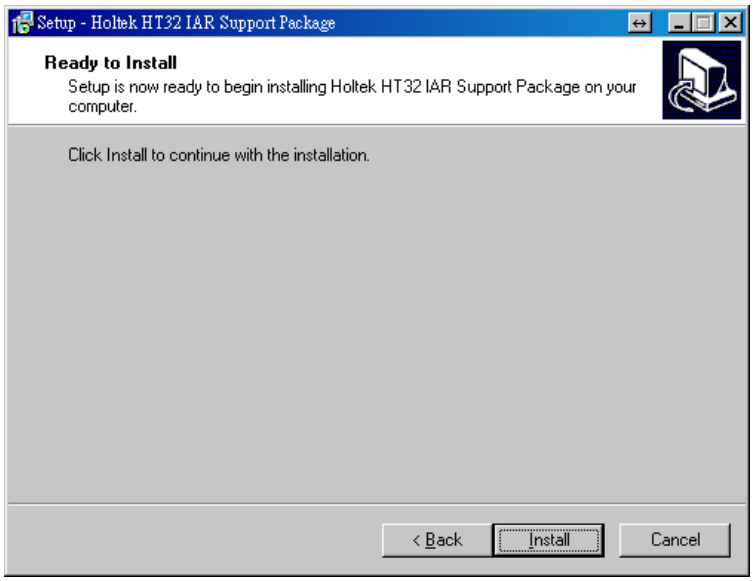


Figure 5. Support Package Installation – Ready to Install

If various versions of IAR EWARM have been installed on the system, a version selection page will appear as shown below. Select a version from the list box to install the HT32 support package and press “Next” to continue. The ready-to-install page will then appear. Press the “Install” button to continue.

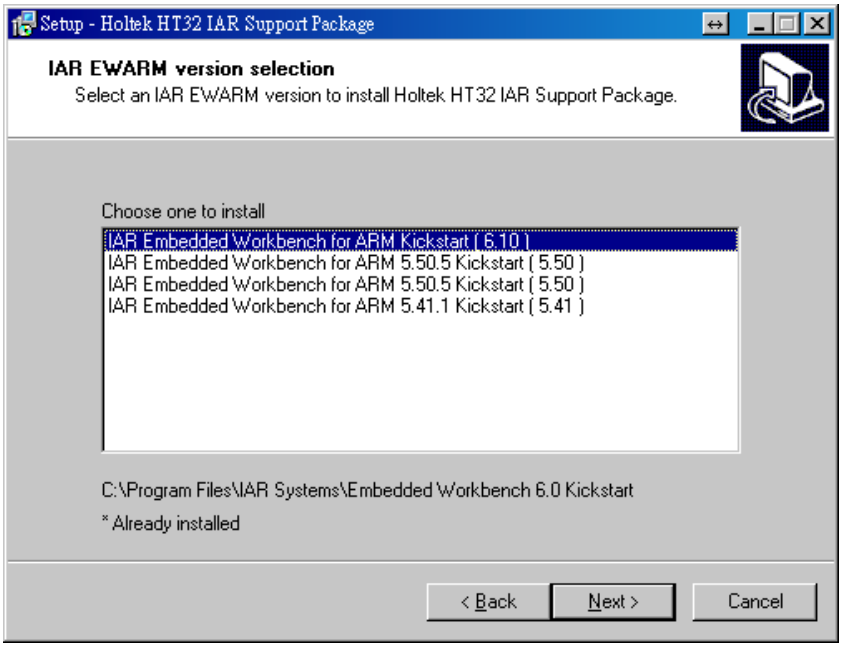


Figure 6. Support Package Installation – Version Selection

Step 4: After the installation has completed, a completing page will appear. Choose whether or not to view the release note. Press “Finish” button to exit the installation program.

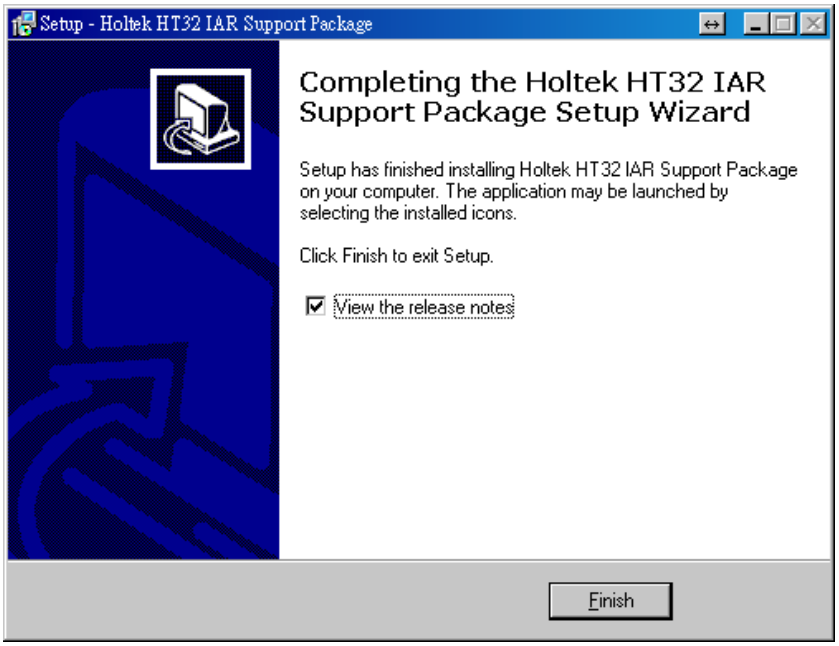


Figure 7. Support Package Installation – Completion

4 Installing the USB Debug Adapter

The J-Link drivers will be installed together with the IAR EWARM. 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 J-Link or e-Link32 USB drivers.

Step 1: Connect the J-Link or e-Link32 to the host PC through 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 drivers\" - for e-Link32
"{IAR_PATH}\arm\drivers\Jlink\x86" - for J-Link 32-bit platform
"{IAR_PATH}\arm\drivers\Jlink\x64" - for J-Link 64-bit platform
```

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, J-Link 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.

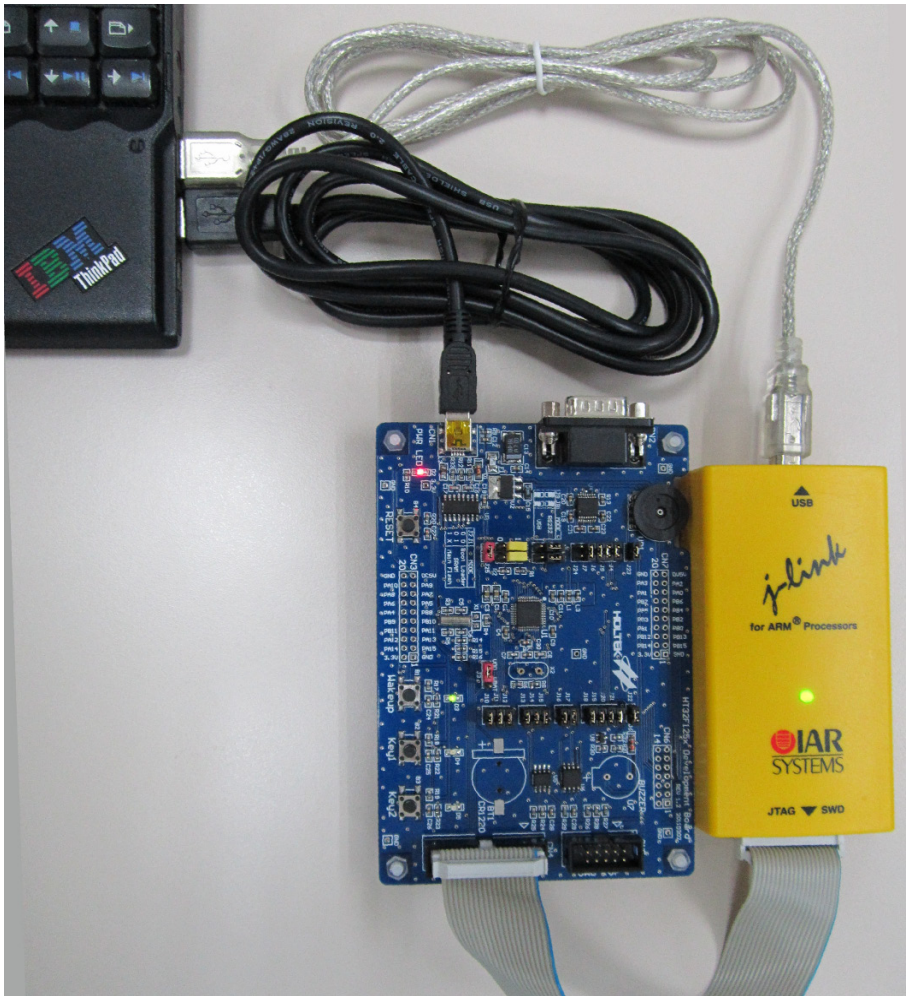


Figure 8. PC, USB Debug Adapter and Target Board Connection

6 IAR EWARM Quick Start

The IAR system provides a complete development tool "IAR Embedded Workbench" for project creation. The tool can edit both C and assembly code, setup the development tools, view the assembler code, connect and perform tests. Visit the IAR Systems' website http://www.iar.com for more information.

Create and Save New Project

To create a new project, follow the steps below:

1. Make sure that the IAR Holtek HT32 Support Package has been installed.
2. Double click on the "IAR Embedded Workbench" shortcut or click "Start → All Programs → IAR Systems → IAR Embedded Workbench" to run the IAR Embedded Workbench.
3. Choose "Project → Create New Project…" to create a new IAR project in the menu.

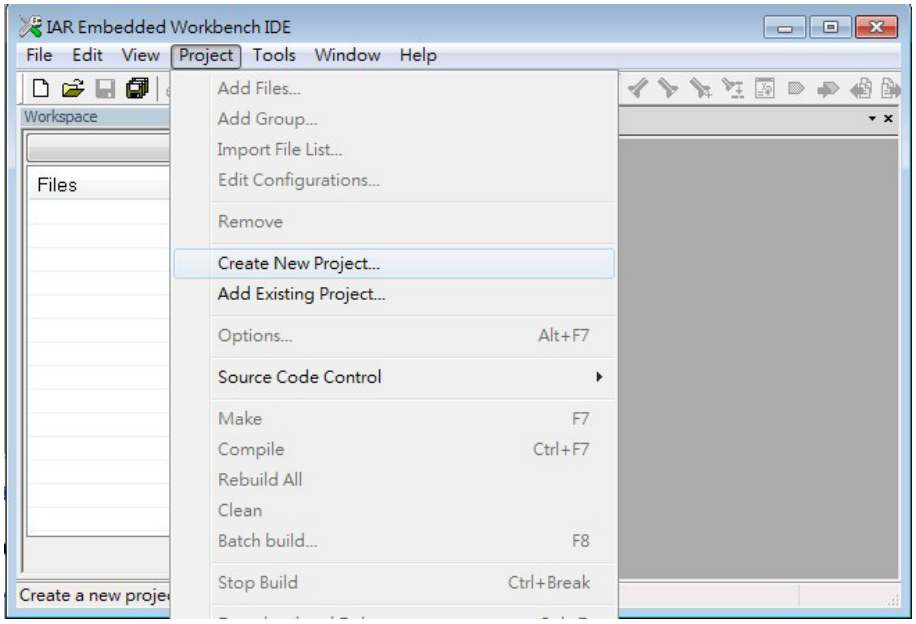


Figure 9. Create New Project

4. A "Create New Project" dialog will pop up. Choose "ARM" from the "Tool chain" drop-down list.
5. Select "Empty project" from the project template.
6. Click "OK" to create an empty project.

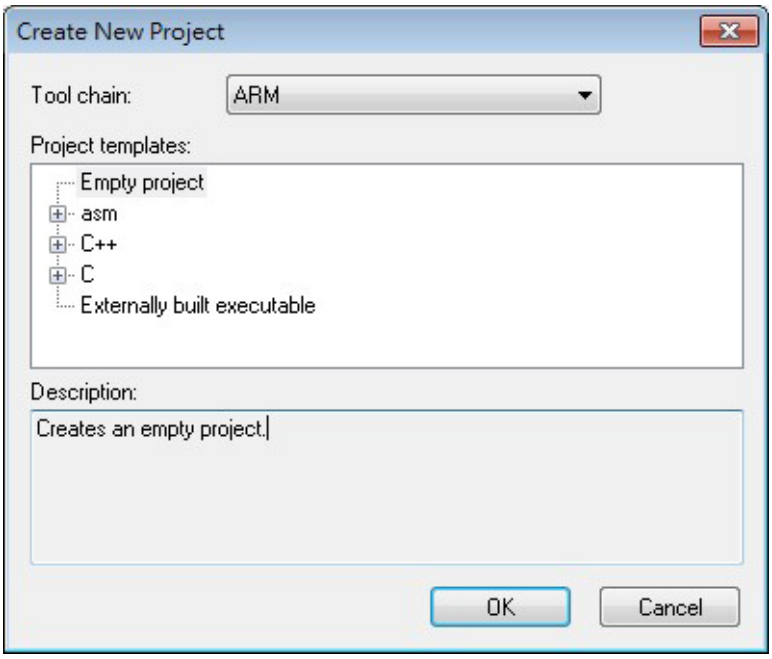


Figure 10. Create New Project

7. Specify the name and path of the project and press the "Save" button. The related information and files can then be found in the "Workspace" window.

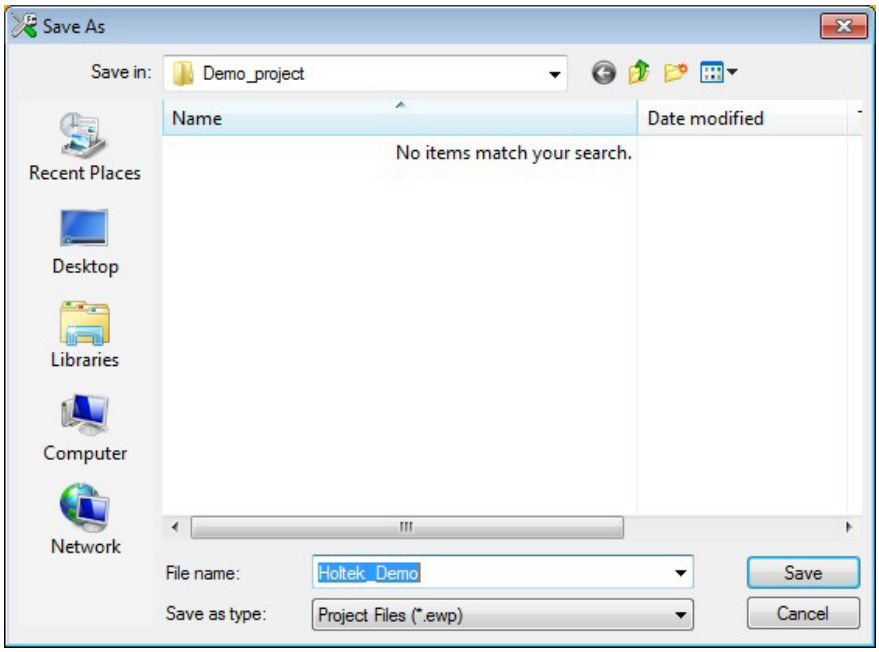


Figure 11. Save Project

8. Click “File → Save Workspace” to save the workspace.

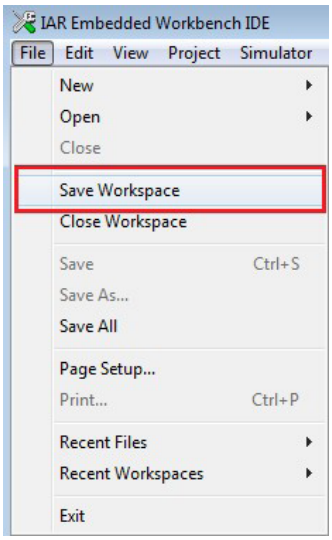


Figure 12. Save Workspace

9. Input the “File name” and click “Save” in the save dialog box.

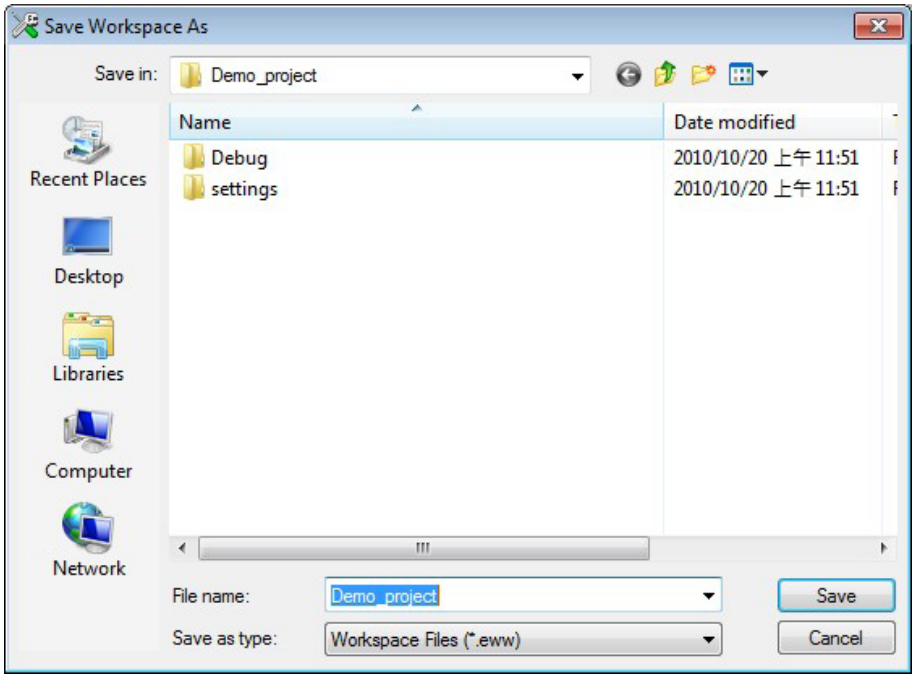


Figure 13. Save Workspace

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…" pages:

a. Right-click on the target name in the "Workspace" window to display the Context Menu and choose "Options…" to open the option dialog.

b. Click on the target name in the "Workspace" window and then select "Project → Options…" in the Project menu.

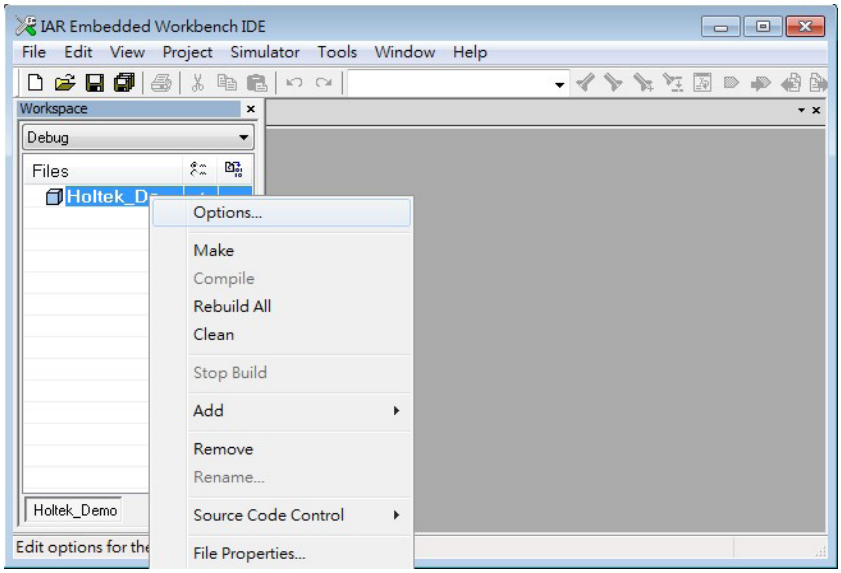


Figure 14. Open Options

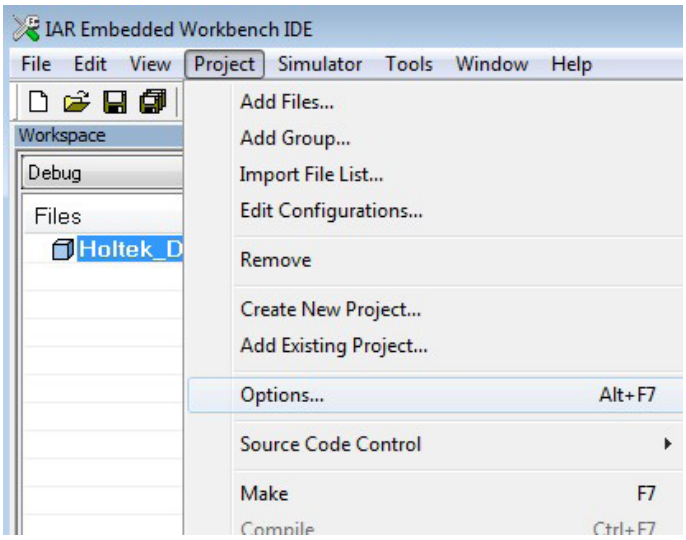


Figure 15. Open Options

Device Selection

1. Open the "Options…" dialog.
2. In the "General Options → Target" tab, click on the "Device" option and press the icon on the right side to select a chip name. For example, the "HT32F1253".

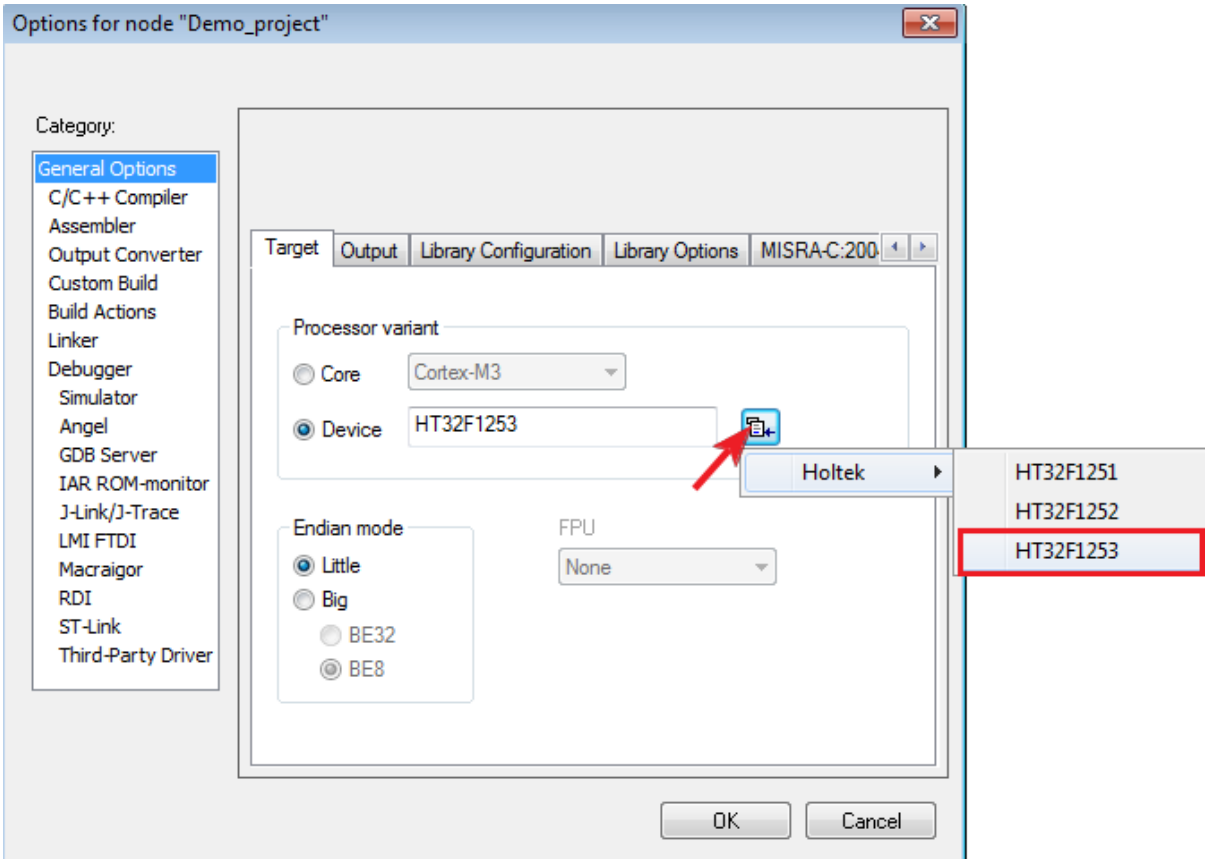


Figure 16. General Options Setup

Configure the USB Debug Adapter

The following shows the method of configuring the IAR EWARM USB debug adapter. This example will take the J-Link as an example.

1. Connect the J-link to the PC.
2. Open the "Options…" dialog.
3. Select "Debugger" in the "Category" list to display the debugger option page. In the "Setup" tab, choose "J-Link/J-Trace" from the drop-down list named "Driver".

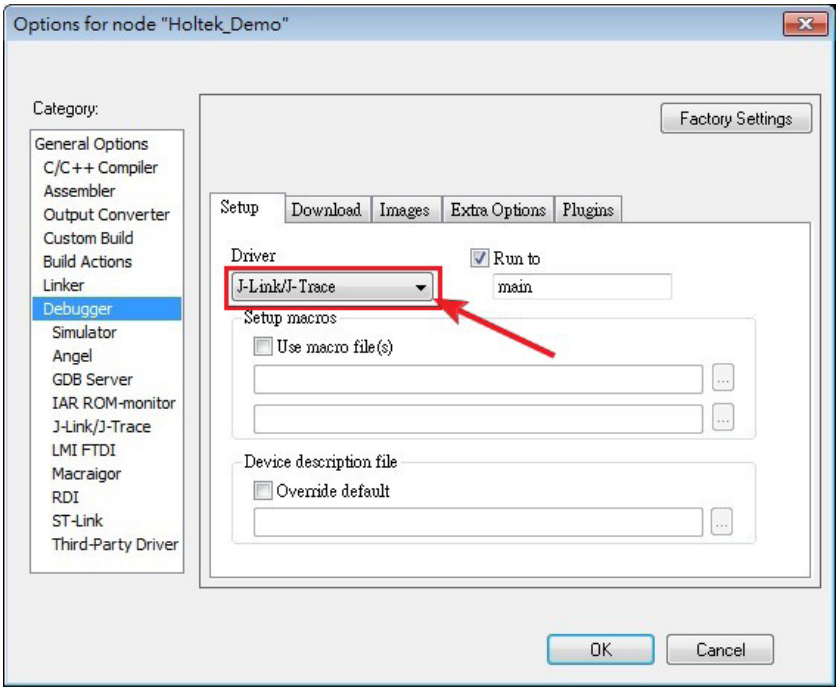


Figure 17. Debugger Driver Select

4. Select “J-Link/J-Trace” in the “Category” list and choose the “Connection” page. Select “Interface” as “SWD” or “JTAG”. Note that some HT32 series devices support only the “SWD” interface.

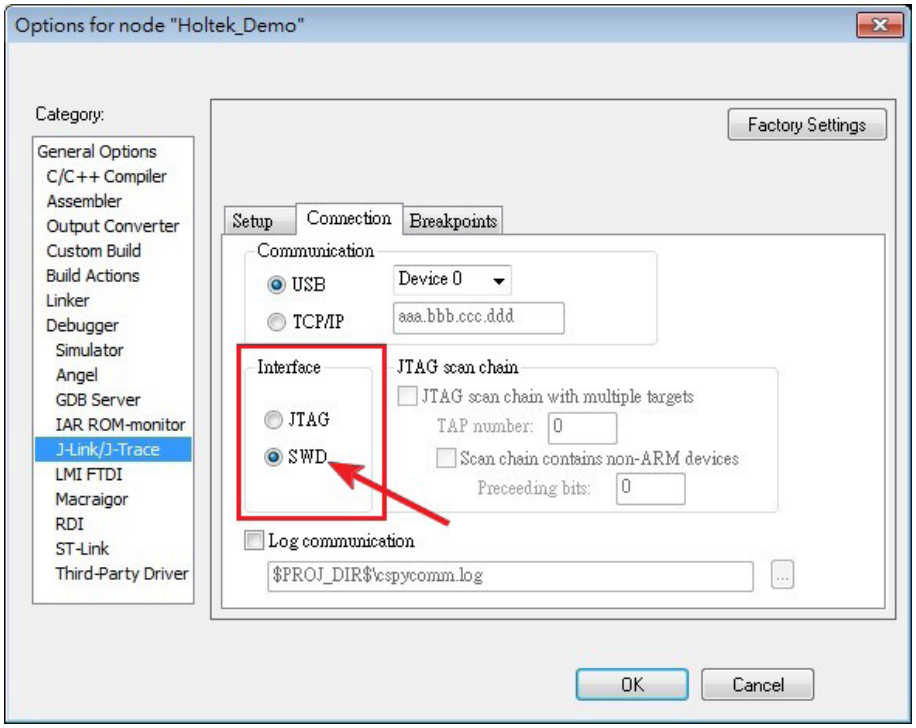


Figure 18. J-Link/J-Trace Interface Setup

Flash Loader Setup

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

1. Open the "Options…" dialog.
2. Select "Debugger" in the "Category" list and choose the "Download" page. Tick "Use flash loader(s)".

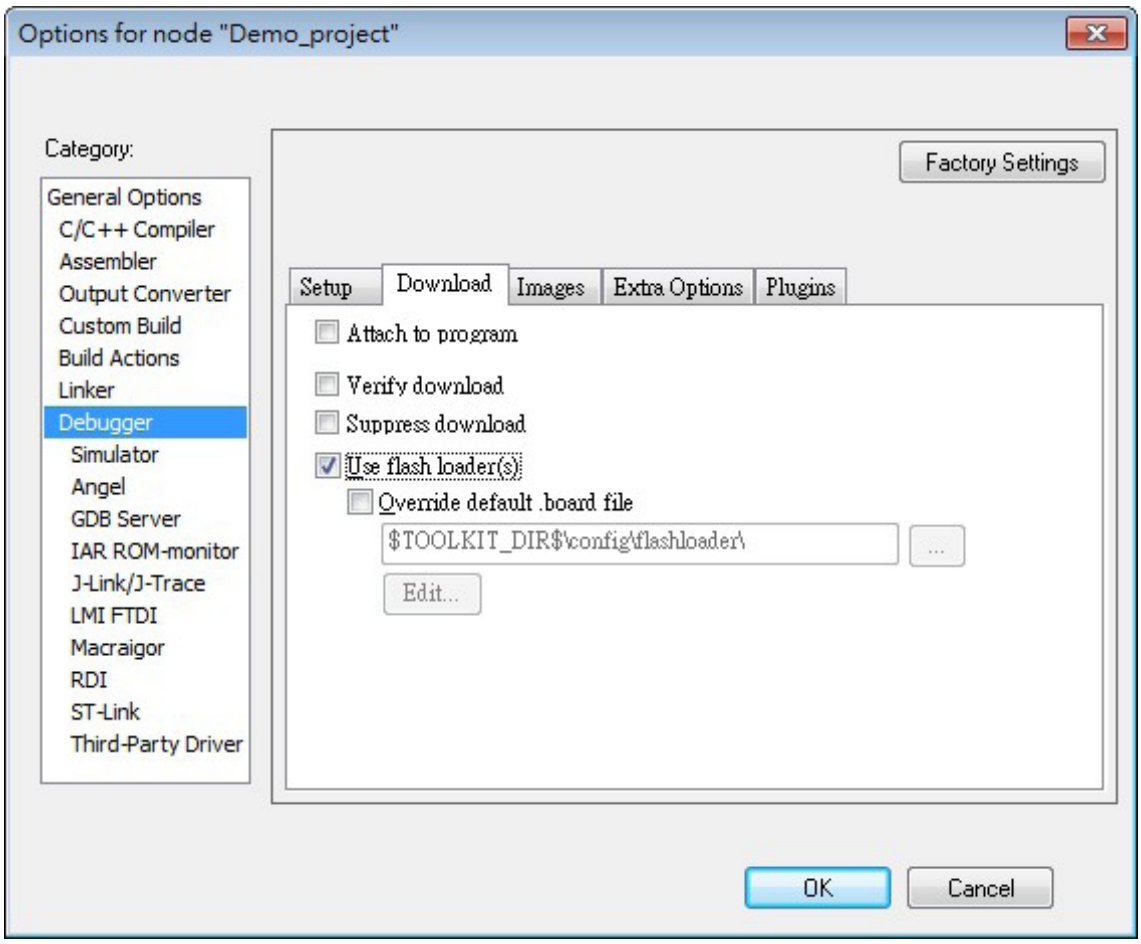


Figure 19. Tick Flash Loader

Adding Source Files to the Project

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

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

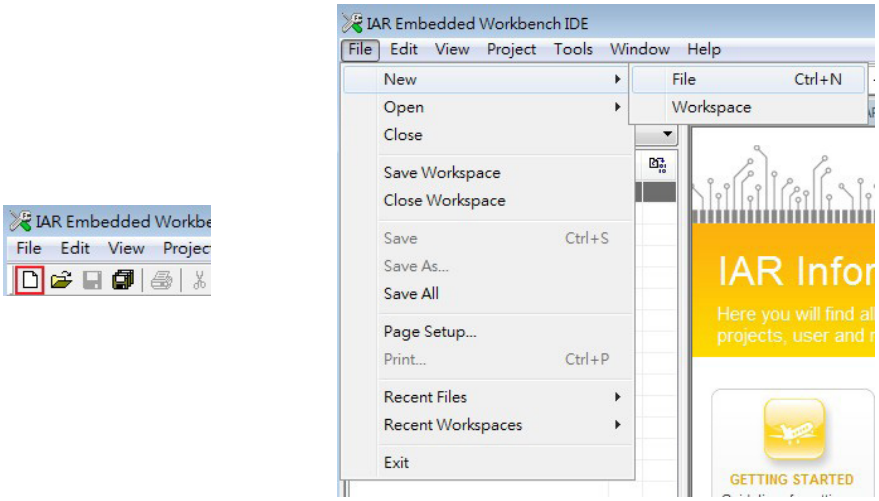


Figure 20. New File Creation

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

```
#include <stdio.h>

int main()
{
        int memory_address;
        int counter = 0;
        memory_address = 0x20001000;
        while(1)
        {

        *((int *) memory_address) = counter++;
        }
}
```

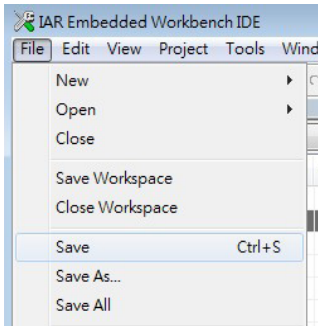


Figure 21. Save File

3. Right click on the project folder icon and select "Add → Add Files" to add "main.c" file into the project.

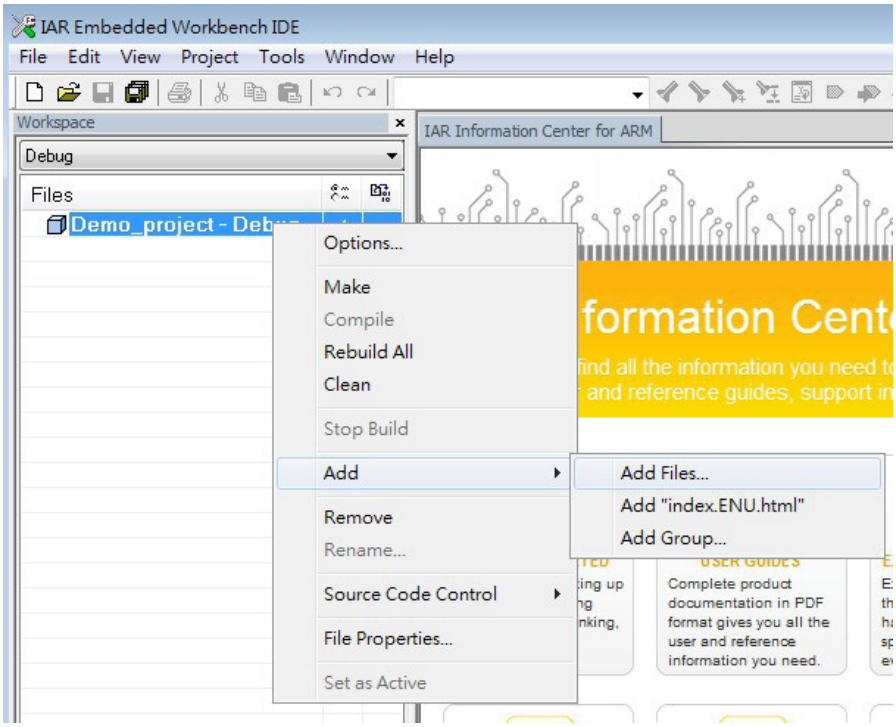


Figure 22. Add File to Workspace

4. The file will then be listed in the Workspace window.

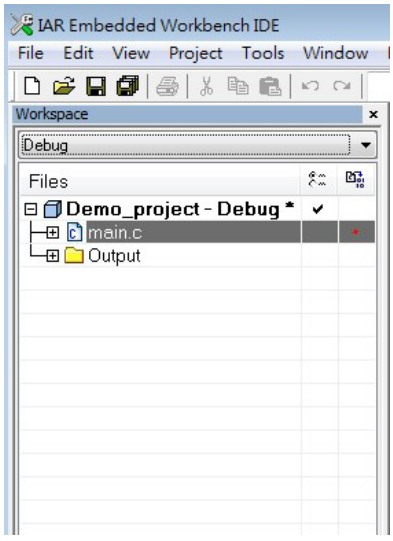


Figure 23. Successful Addition of File to Workspace

7 Compiling the Project

Use the following procedure to compile the project:

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

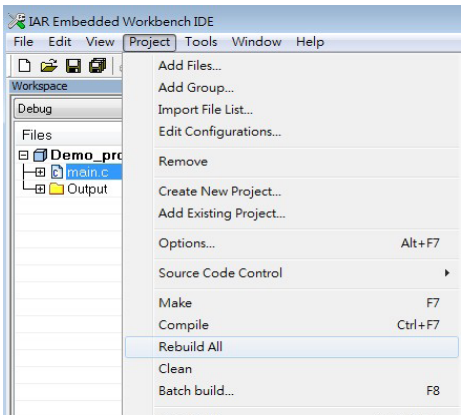


Figure 24. Project Building

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

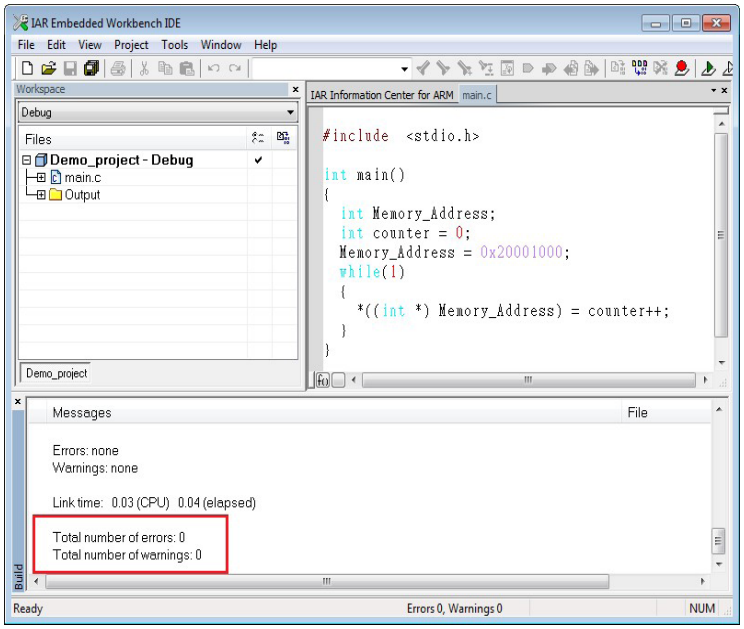


Figure 25. Compiler Messages

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 the "Download and Debug" icon in the toolbar to enter the debug mode. The image will be downloaded into Flash memory automatically.

Figure 26. Download and Debug Button

2. After the debugging is finished, click the "Stop Debugging" icon to exit the debug mode.

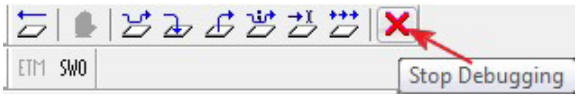


Figure 27. Stop Debugging Icon

3. The debug window is shown below. Included is information regarding "Registers", "Disassembly", "Source Code Window", "Memory Window", "Command" and so on.

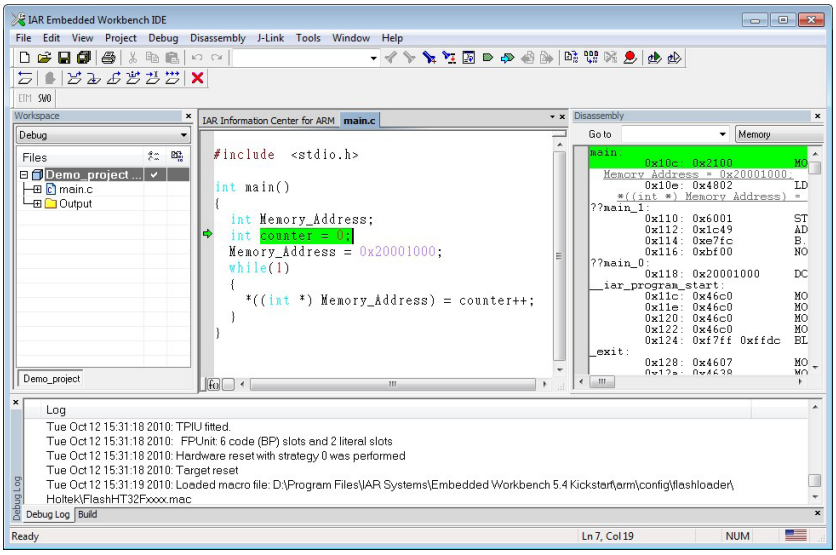


Figure 28. Debugger Main Window

Free Running, Break, and Single Step Operation

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

1. Click the "Go" icon to allow the program to free run.

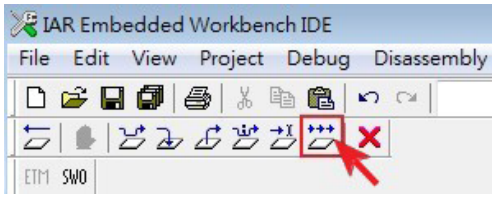


Figure 29. Go Icon

2. Click the "Break" icon to stop the program free running.

Figure 30. Break Icon

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

Figure 31. Step Into 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.

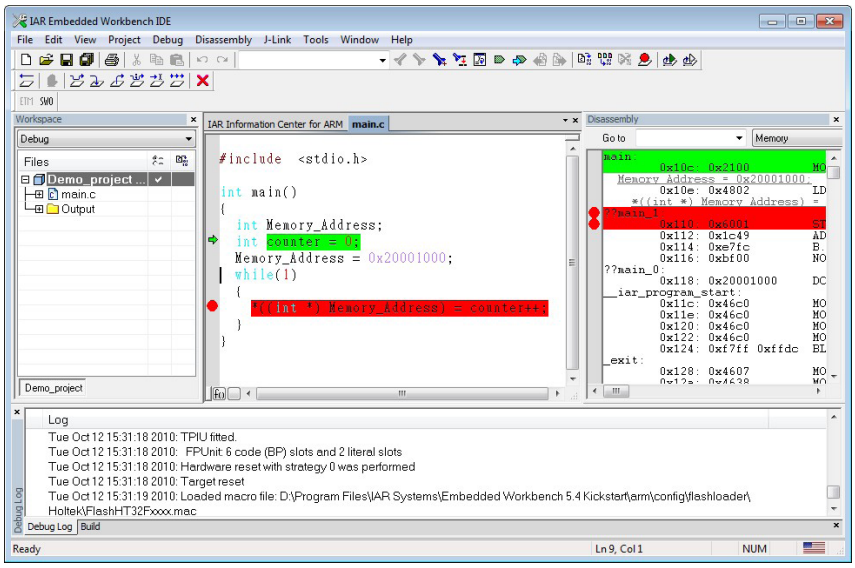


Figure 32. Breakpoint Setup

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

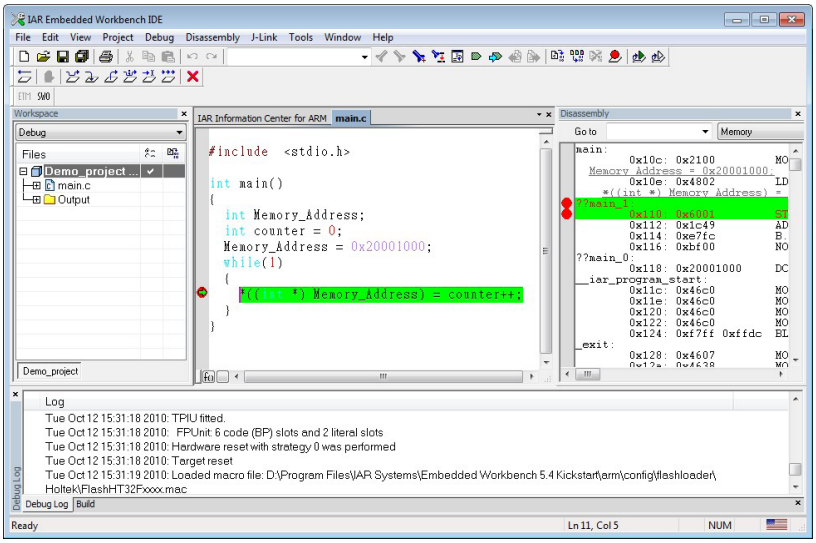


Figure 33. Breakpoint Program Stop

Memory Window

A function is included to examine user selected memory contents.

1. Click "View → Memory" to open the memory window.

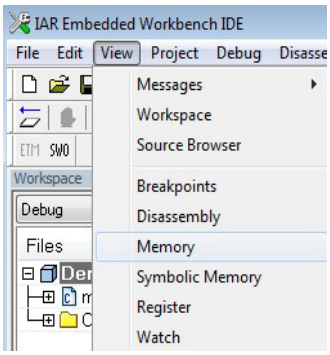


Figure 34. Open Memory Window

2. Type "0x20001000" in the "Go To" input box to view the value in the memory.

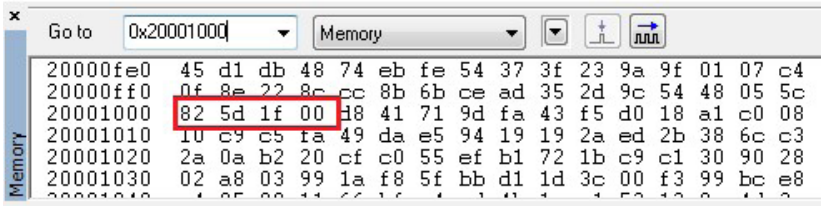


Figure 35. Address 0x20001000 Memory Contents Window

3. Click the "Step Into" icon or "F11" to single step the program. Use this feature to examine consecutive memory contents.

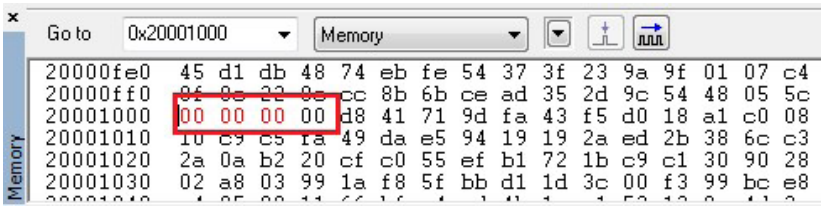


Figure 36. Memory Content Examination

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. Copy “FlashMacro.mac” to the project folder. The original “FlashMacro.mac” file is located in the IAR Systems installation folder: “arm\config\flashloader\Holtek”.

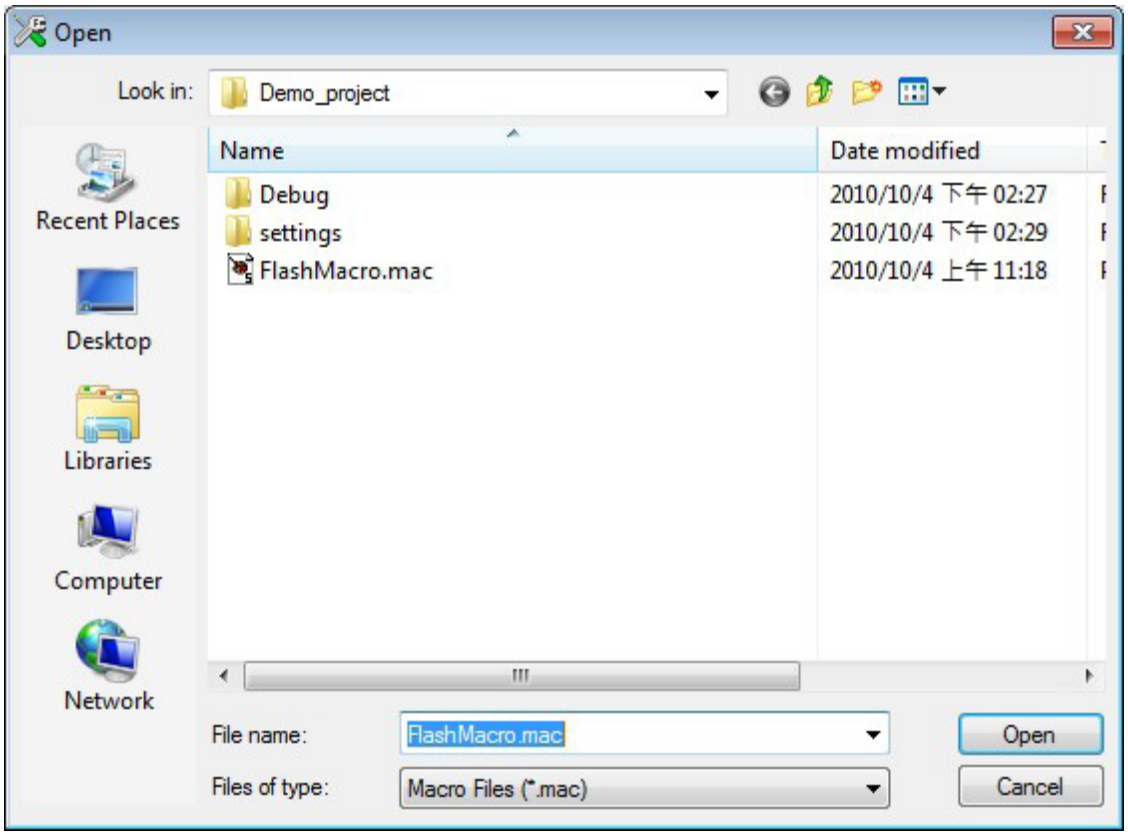


Figure 37. Add Files

2. To load the file automatically through the IAR Embedded Workbench:

a. Select "Options…" and then select the "Debugger → Setup" page.

b. Tick "Use macro file(s)" and add the path correctly to the window as shown below.

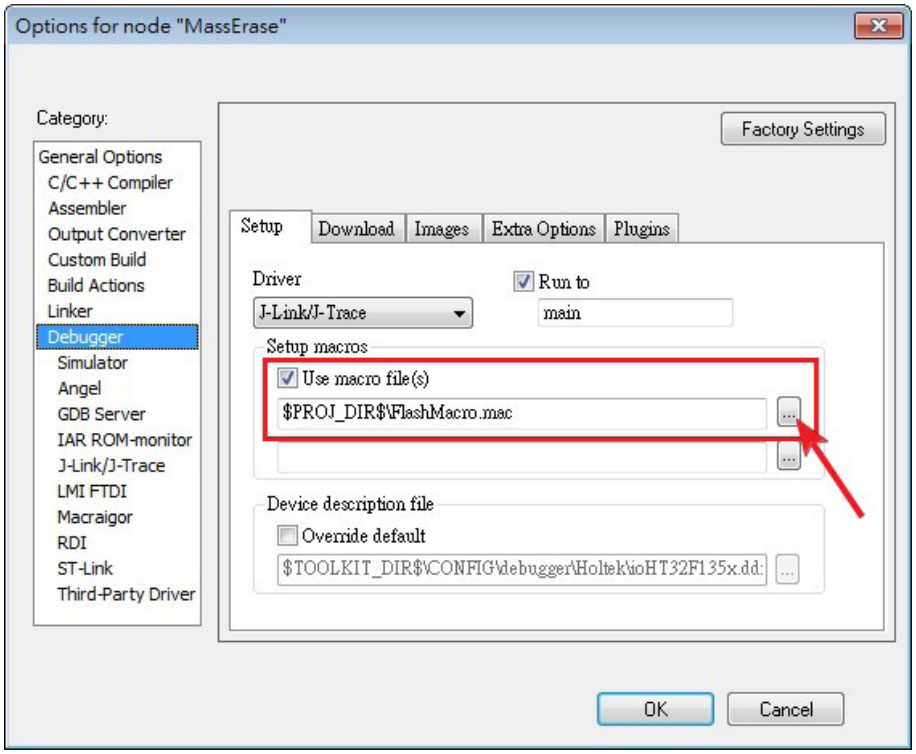


Figure 38. Macro File Setup

3. Press "Ctrl+D" or the "Download and Debug" button to start the debug session. The "FlashMarcro.mac" will then be loaded by IAR EWARM automatically.

Figure 39. Download and Debug Button

4. Click "View → Memory" and type "0x0" in the "Go To" input box to view the Flash data that has been previously downloaded by the Flash loader. The data is shown in the memory window as follows.

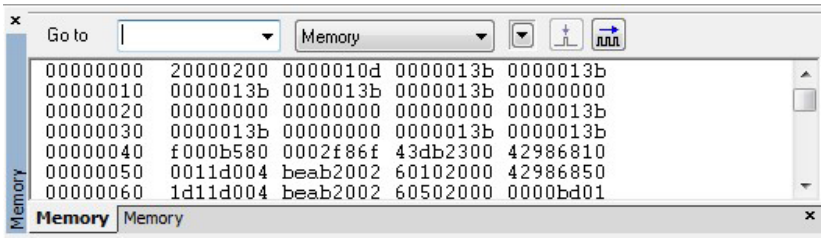


Figure 40. Memory Window

5. Click "View → Quick Watch" to open the quick watch window. Type "ht32f_masserase()" in the window and press "Enter".

Figure 41. Quick Watch Window

6. A warning message box will be shown since the default stack point in the main Flash has already been reset to 0xFFFFFFFF by a mass erase. This message can be ignored.

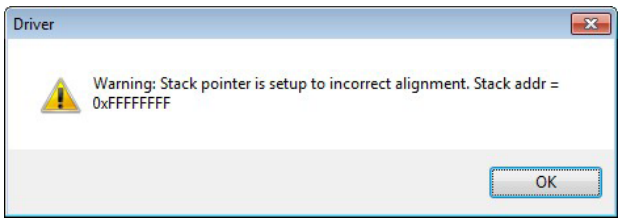


Figure 42. Warning Message Box

7. Press OK and a message like "HT32: Mass erase…" will be shown in the Debug Log window.

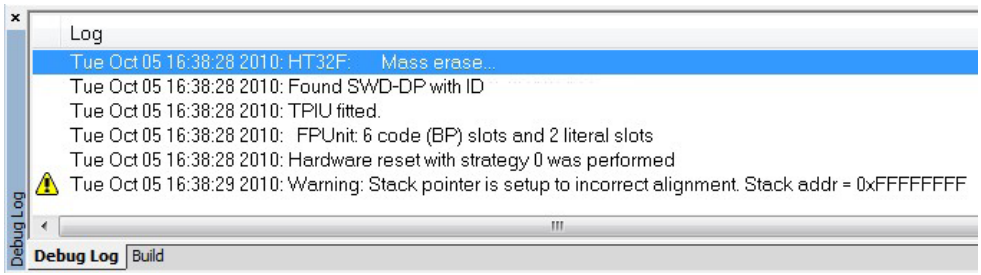


Figure 43. Debug Log Message Window

8. Check the memory window again. The whole flash memory should have been reset to 0xFFFFFFFF.

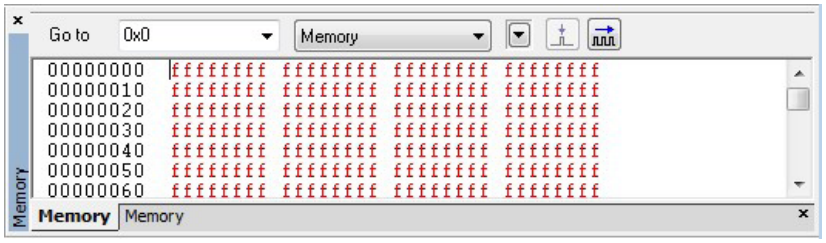


Figure 44. Memory to be Mass Erased

10 Option Bytes Programming

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

1. Copy “FlashMacro.mac” to the project folder. The original “FlashMacro.mac” file is located in the IAR Systems installation folder: “{IAR_PATH}\arm\config\flashloader\Holtek”.

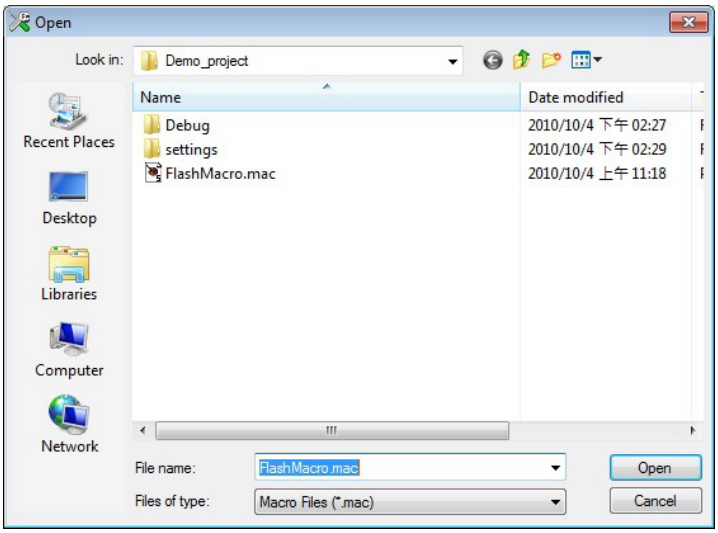


Figure 45. Add Files

2. To load the file automatically through the IAR Embedded Workbench.

a. Select “Options…” and then select the “Debugger → Setup” page.

b. Tick “Use macro file(s)” and add the path correctly to the window as shown below.

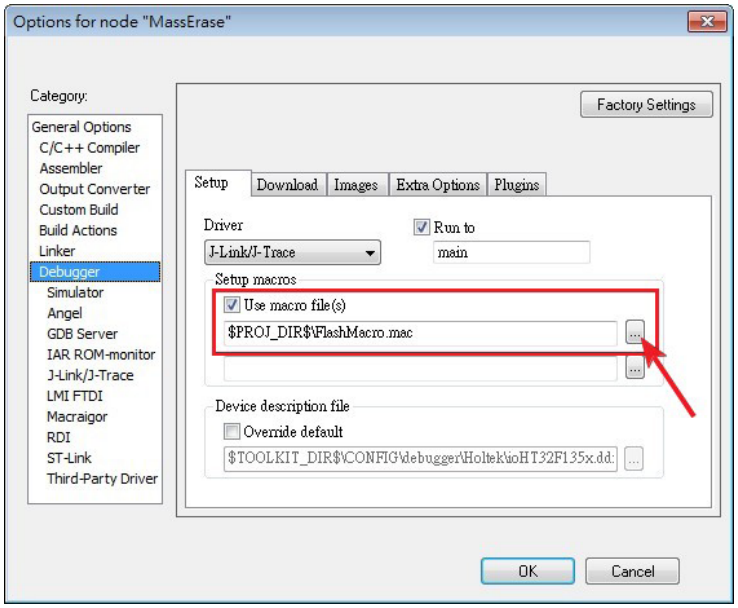


Figure 46. Macro File Setup

3. Open FlashMarco.mac to modify the Option Byte values inside the “InitOBValue()” function. Uncomment “PROGRAM_OB=1”; and it will automatically update the Option Byte when the debugging session starts.

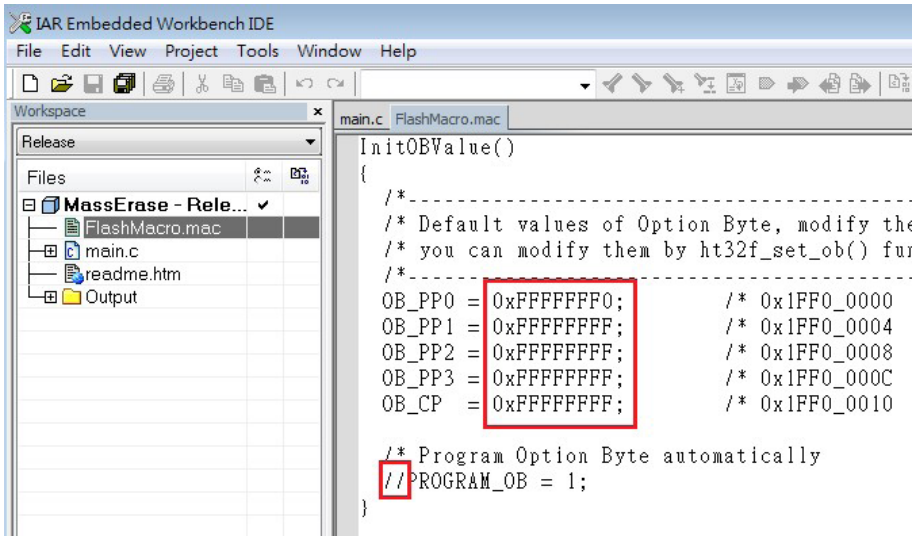


Figure 47. FlashMarco.mac

4. Press “Ctrl+D” or “Download and Debug” button to start debugging session. The FlashMarcro.mac will then be loaded by IAR EWARM automatically.

Figure 48. Download and Debug Button

5. View the Option Byte information from the memory window. The Option Byte address is at “0x1FF00000”. For this example, “PROGRAM_OB=1” is commented and so the Option Byte will not be automatically updated.

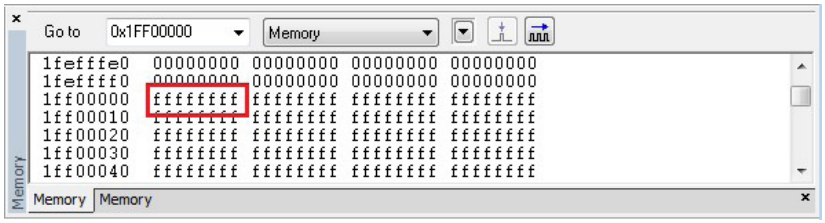


Figure 49. Option Byte Memory Window Display

6. Click “View → Quick Watch” to open the Quick Watch window. Type “ht32f_update_ob()” in the window and press “Enter”.

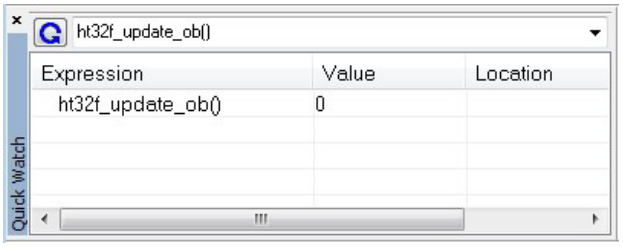


Figure 50. Quick Watch Window Key-in Function

7. Once the "ht32f_update_ob()" function has been executed, a message will appear as shown in the Debug Log window below.

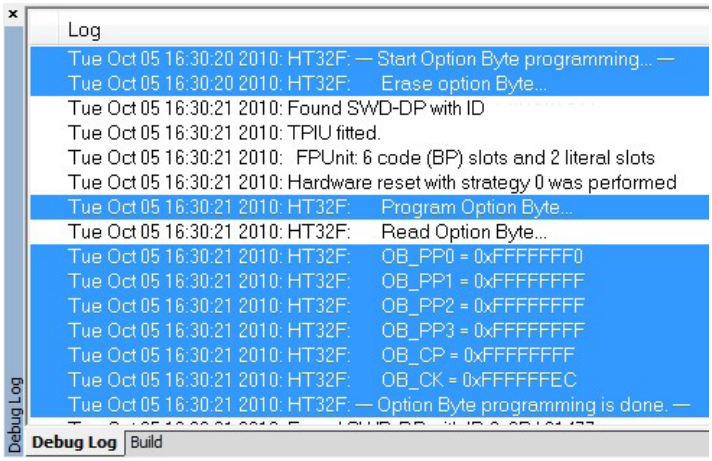


Figure 51. Option Byte Programmed Successfully

8. Check the memory window to confirm the Option Byte value.

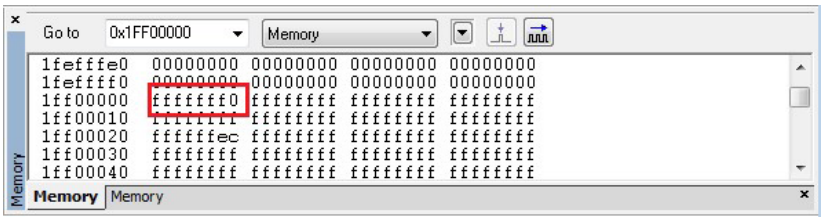


Figure 52. Option Byte Successful Update

11 Conclusion

All the necessary items for developing embedded software such as IAR EWARM, USB debug adapter, software project, debugging environment have been included in this document. A brief introduction to the creation, management, building and debugging of software using the IAR EWARM has also been provided to help get started with software development using the IAR EWARM 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.