GR55xx IAR User Manual

Version: 1.1

Release Date: 2021-06-26

Shenzhen Goodix Technology Co., Ltd.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd. All rights reserved.

Any excerption, backup, modification, translation, transmission or commercial use of this document or any portion of this document, in any form or by any means, without the prior written consent of Shenzhen Goodix Technology Co., Ltd is prohibited.

Trademarks and Permissions

GOODIX and other Goodix trademarks are trademarks of Shenzhen Goodix Technology Co., Ltd. All other trademarks and trade names mentioned in this document are the property of their respective holders.

Disclaimer

Information contained in this document is intended for your convenience only and is subject to change without prior notice. It is your responsibility to ensure its application complies with technical specifications.

Shenzhen Goodix Technology Co., Ltd. (hereafter referred to as "Goodix") makes no representation or guarantee for this information, express or implied, oral or written, statutory or otherwise, including but not limited to representation or guarantee for its application, quality, performance, merchantability or fitness for a particular purpose. Goodix shall assume no responsibility for this information and relevant consequences arising out of the use of such information.

Without written consent of Goodix, it is prohibited to use Goodix products as critical components in any life support system. Under the protection of Goodix intellectual property rights, no license may be transferred implicitly or by any other means.

Shenzhen Goodix Technology Co., Ltd.

Headquarters: 2F. & 13F., Tower B, Tengfei Industrial Building, Futian Free Trade Zone, Shenzhen, China

TEL: +86-755-33338828 FAX: +86-755-33338099

Website: www.goodix.com

Preface

Purpose

This document introduces methods to develop and to debug GR55xx Bluetooth Low Energy (Bluetooth LE) applications based on IAR Embedded Workbench IDE, to help users quickly get started with secondary development of GR55xx SDK applications.

Audience

This document is intended for:

- GR55xx user
- GR55xx developer
- GR55xx tester
- Hobbyist developer
- Technical writer

Release Notes

This document is the second release of *GR55xx IAR User Manual*, corresponding to GR55xx SoC series.

Revision History

Version	Date	Description
1.0	2021-03-08	Initial release
1.1	2021-06-26	• Updated the descriptions in “Configuring an Application Project”. • Updated GR551x to GR55xx. • Optimized “Installation Instructions”.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

Contents

Preface ..... **I**

1 Introduction **1**

2 Installation Instructions **2**

2.1 Installation Requirements 2

2.2 Software Installation 2

3 Configurations for SDK Applications **4**

3.1 Building an Application Project 4

3.2 Configuring an Application Project 5

3.3 Compiling and Downloading a Project 11

3.4 Project Debugging 13

4 FAQ **14**

4.1 Why does an “Undefined Variable” Error Occur During Project Compilation? 14

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

1 Introduction

IAR Embedded Workbench IDE for Arm (IAR EWARM, mentioned as IAR below) is an integrated development environment (IDE) built by IAR Systems, supporting 8-bit, 16-bit, and 32-bit microprocessors and microcontrollers, and providing more user-friendly experience among ARM IDEs. IAR features:

- Highly optimized C/C++ Compiler
- IAR XLINK Linker supporting common applications
- Powerful support for project management
- C-SPY Debugger, a fully integrated debugger with performance analysis, power visualization, and RTOS plugins

This document serves as a roadmap for users building GR55xx application development environment with IAR on Windows PCs, so as to create, compile, download, and debug Bluetooth LE applications, with an example provided as reference.

Before getting started, you can refer to the following documents.

Table 1-1 Reference documents

Name	Description
GR55xx Developer Guide	Introduces the software/hardware and quick start guide of GR55xx SoCs.
J-Link/J-Trace User Guide	Provides J-Link operational instructions. Available at http://www.segger.com/downloads/jlink/UM08001_JLink.pdf.
Bluetooth Core Spec v5.1	Offers official Bluetooth standards and core specification (v5.1) from Bluetooth SIG. Available at https://www.bluetooth.com/specifications/bluetooth-core-specification/.
Bluetooth GATT Spec	Provides details about Bluetooth profiles and services. Available at https://www.bluetooth.com/specifications/gatt.
IAR Embedded Workbench IDE User Guides	Provides various information about IAR IDE. Available at:https://www.iar.com/support/user-guides/user-guide-iar-embedded-workbench-for-arm/.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

2 Installation Instructions

This chapter describes the requirements and process for installing IAR.

2.1 Installation Requirements

- **Supported Development Platform**

Table 2-1 Supported Development Platform

Hardware Platform	Development Board Model
GR551x development kit	GR5515-SK-BASIC

- **Hardware and Software preparation**

Table 2-2 Hardware and Software preparation

Name	Description
Windows	Windows 7/Windows 10
IAR Embedded Workbench IDE for Arm (IAR EWARM)	IAR EWARM 8.2.22 or later versions
J-Link driver	A J-Link driver. Available at www.segger.com/downloads/jlink/.
Connection cable	Micro USB 2.0 cable

2.2 Software Installation

Download the IAR installation file (IAR for ARM 8.2.22 or later versions) on IAR Systems official website.

Double-click the .exe installation file. Choose **Install IAR Embedded Workbench ® for Arm**, and follow the prompts to finish installation.

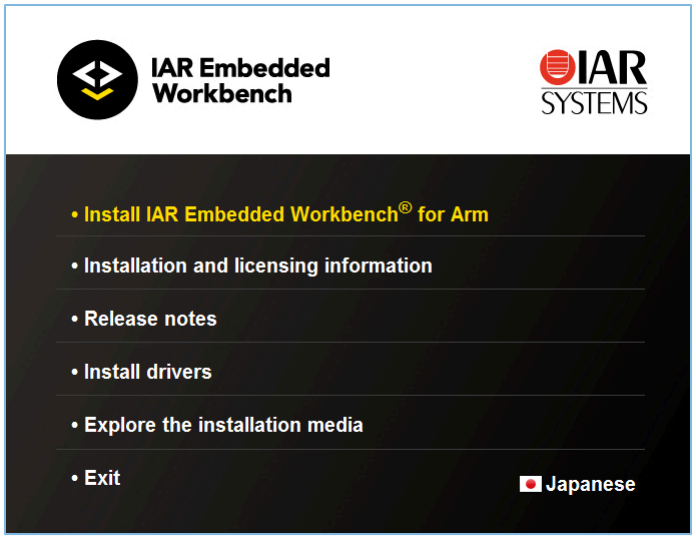

Figure 2-1 IAR installation page

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

IAR will be launched automatically after successful installation. The main user interface is shown as below:

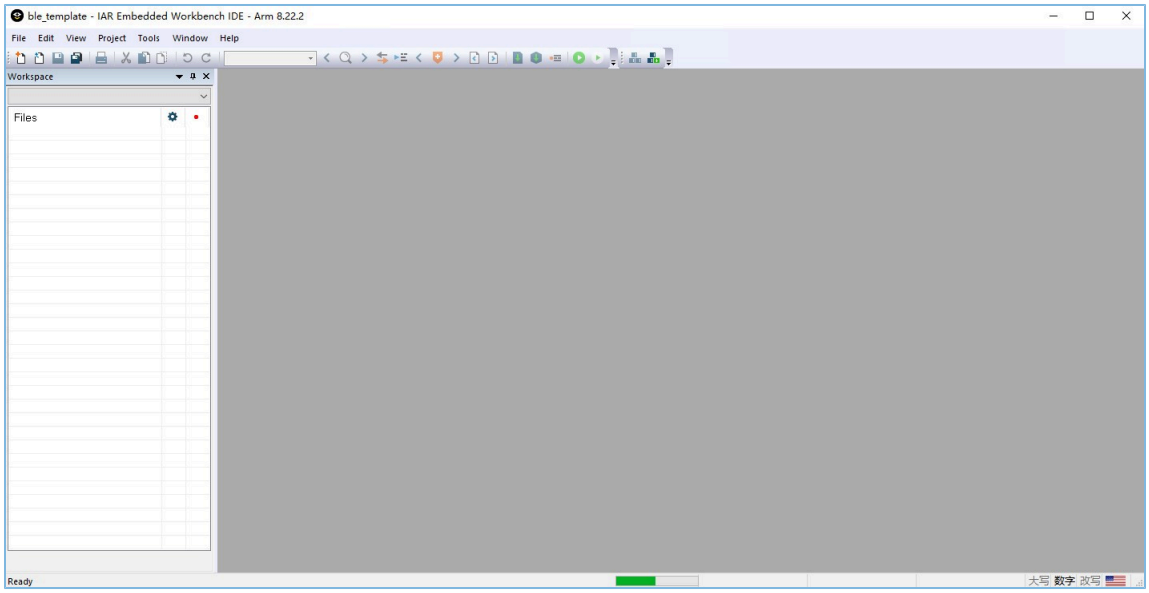

Figure 2-2 IAR main user interface

Users can choose to install the necessary software as prompted after installing IAR.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

3 Configurations for SDK Applications

This chapter elaborates on how to configure an IAR project, to help users quickly get familiar with and use IAR projects with GR55xx SDK.

Note:

- SDK_Folder is the root directory of the GR55xx SDK in use.
- IAR_Folder is the root directory of IAR installation.

3.1 Building an Application Project

This section introduces how to generate an IAR application project as needed based on an existing project (ble_app_template as an example, available in `SDK_Folder\projects\ble\ble_peripheral\ble_app_template\IAR`).

1. Open `SDK_Folder\projects\ble\ble_peripheral\`, and copy ble_app_template to the directory. Rename the project to ble_app_example. In `ble_app_example\IAR`, change the file names of the *.ewp* file, the *.ewt* file, and the *.eww* file to ble_app_example.
2. Open *ble_app_example.eww* in text format, and replace ble_app_template (as shown in the figure below) with ble_app_example.

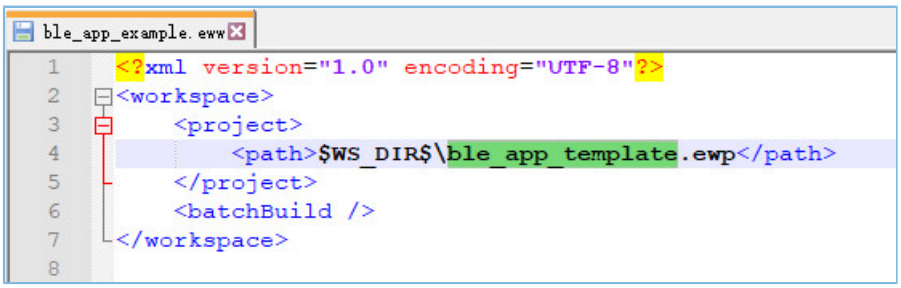

Figure 3-1 Identifying and replacing ble_app_template

3. After the modification completes, double-click *ble_app_example.eww* to enter the IAR main user interface.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

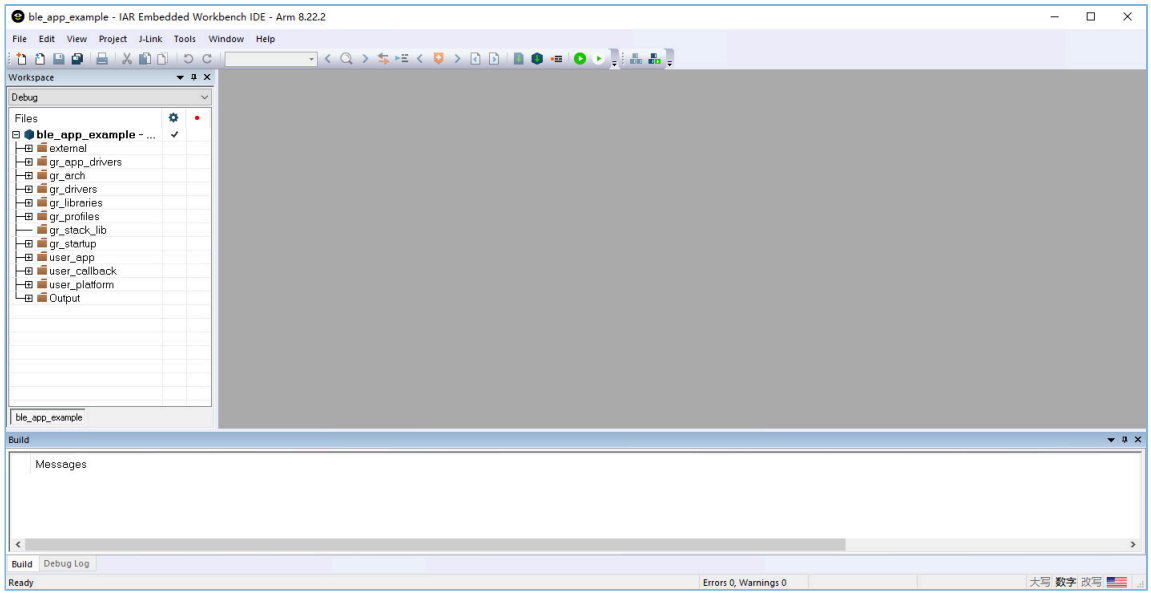

Figure 3-2 Main user interface when an example project opened in IAR

3.2 Configuring an Application Project

Copy the IAR download algorithm files to an appropriate directory before configuring an application project.

1. Find the IAR download algorithm files (*GR55xx_IAR_8M.board*, *GR55xx_IAR_8M.flash*, and *GR55xx_IAR_flashloader_8M.out*) in `SDK_Folder\build\binaries\xflash_flm_tools\IAR`.

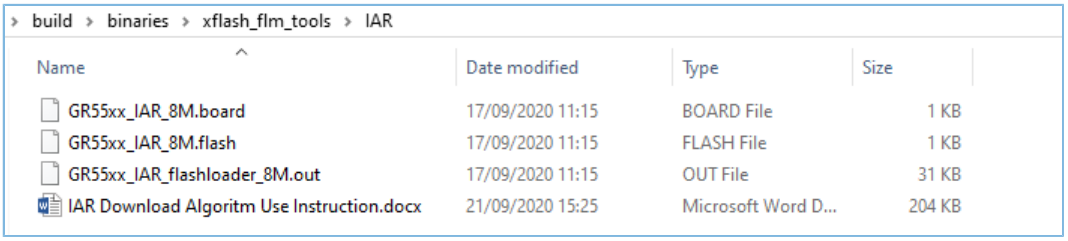

Figure 3-3 IAR download algorithm files

2. Create a new folder (such as Goodix) in `IAR_Folder\arm\config\flashloader`, and copy the three algorithm files mentioned in Step 1 to the new directory, as shown in the figure below.

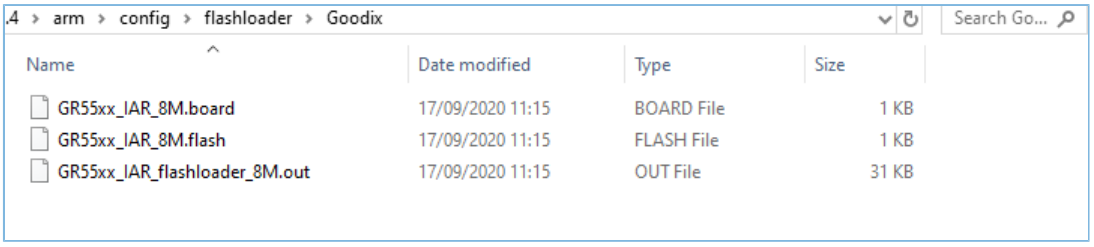

Figure 3-4 Copying the algorithm files

After copying the algorithm files to the target directory, users can start to configure the application project.

1. Right-click the project name in the **Workspace** pane and choose **Options** to start configuration.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

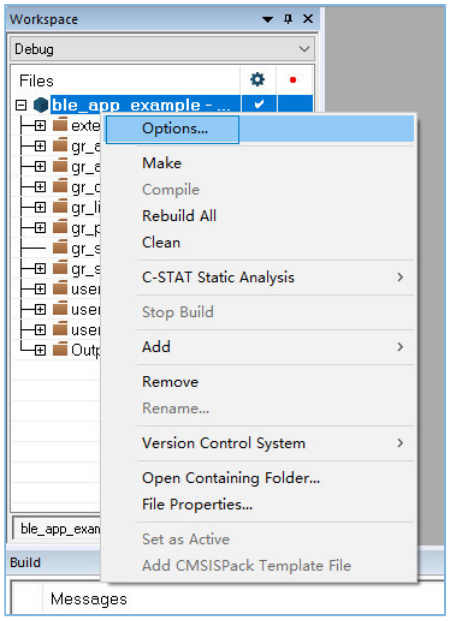

Figure 3-5 To configure the project

(1). Configure project compilation items.

In the **C/C++ Compiler** > **Preprocessor** pane, configure **Additional include directories** and **Defined symbols** in the pop-up window, as shown in the figure below.

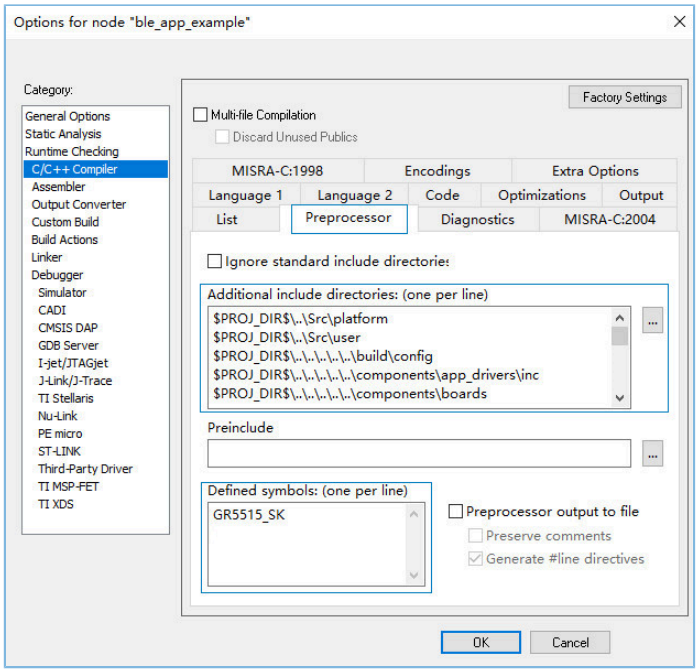

Figure 3-6 Project compilation

Note:

$PROJ_DIR$ in the **Additional include directories** pane is the environment variable for IAR project file path.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

(2). Configure **Linker configuration file**.

Configure **Linker configuration file** in the **Linker** > **Config** pane, as shown in the figure below. Choose the IAR linker configuration file provided in the SDK, available in `SDK_Folder\toolchain\gr551x\source\iar\gr5515.icf`. Click **OK** after the file is selected.

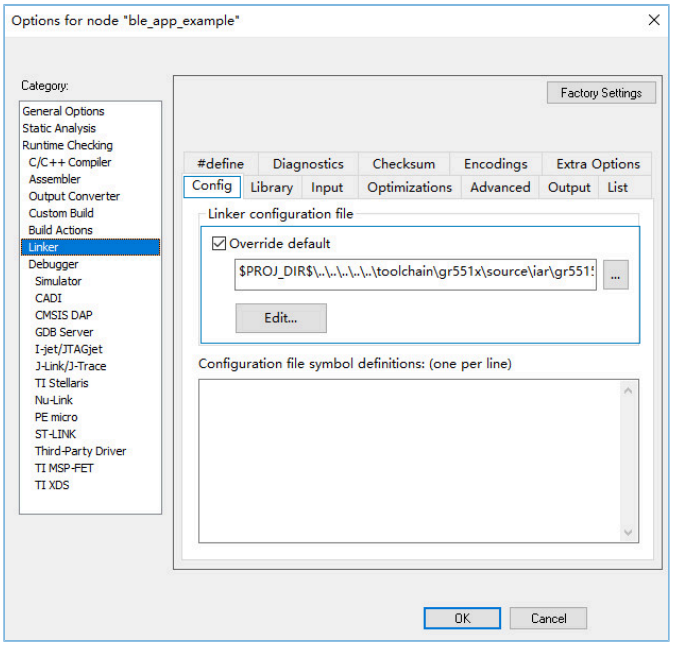

Figure 3-7 Choosing the linker configuration file

In the **Linker** > **Output** pane, modify the file names of the .out file and the .map file (generated when building the project) to ble_app_example. Click **OK** after modification completes.

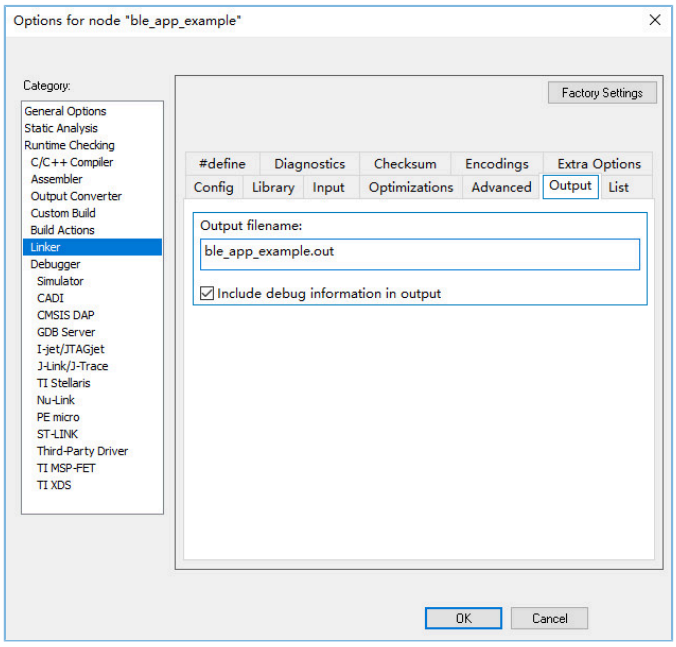

Figure 3-8 Modifying the name of a file generated when building the project

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

(3). Configure the output file.

Configure the format and the name of the file (generated when building the project) in the **Output Converter** > **Output** pane.

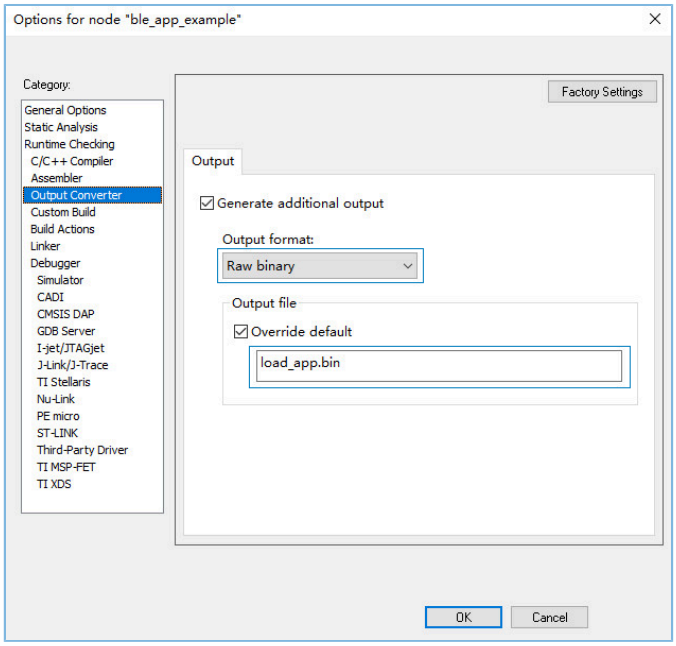

Figure 3-9 Modifying format and name of the output file

(4). Configure **Post-build command line**.

Users shall also configure the post-build script (in the **Build Actions** > **Post-build command line** box) for IAR projects provided in GR55xx SDK (available in `SDK_Folder\build\scripts\after_build_for_IAR.bat`), to generate a *.hex* file.

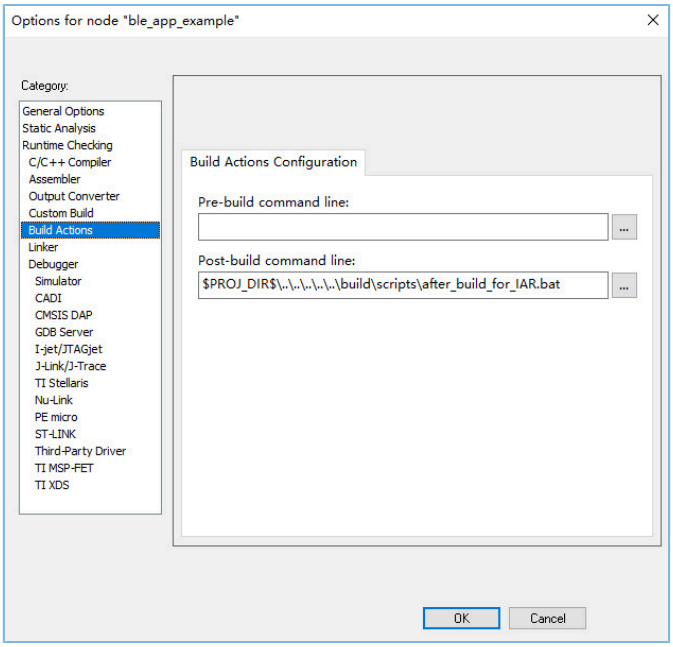

Figure 3-10 Configuring the post-build script

(5). Configure the download algorithm.

Enter the **Debugger** > **Download** pane, and choose the directory of IAR download algorithms mentioned in Step 2 (`IAR_Folder\arm\config\flashloader\Goodix\GR55xx_IAR_8M.board`), to choose the SoC download algorithm.

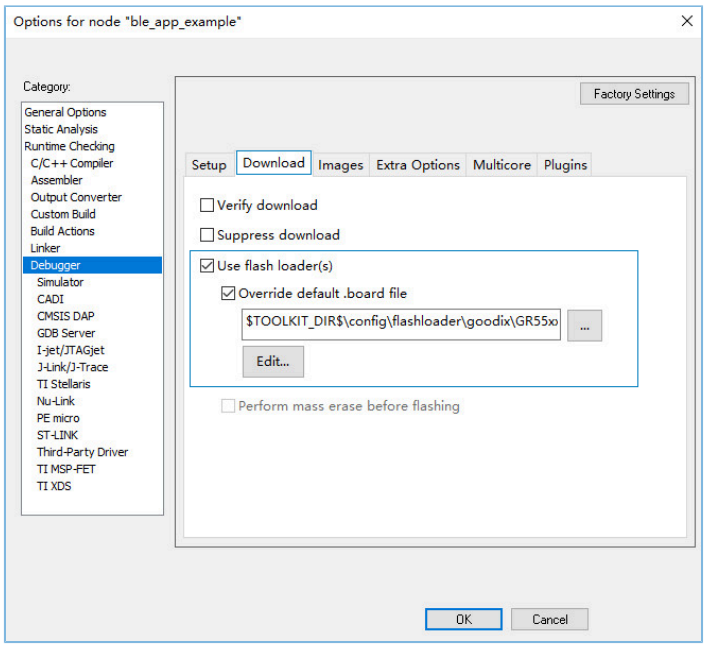

Figure 3-11 Configuring IAR download algorithm

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

Note:

$TOOLKIT_DIR$ in the **Use flash loader(s)** pane is the environment variable for IAR installation path.

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.

3.3 Compiling and Downloading a Project

1. Click **Project** on the menu bar, and choose **Make** or **Rebuild All** to compile firmware. Wait until the compilation completes.

2. Choose **Project** > **Download** > **Download file**, as shown in Figure 3-12. Choose `IAR\build\load_app.hex`, as shown in Figure 3-13.

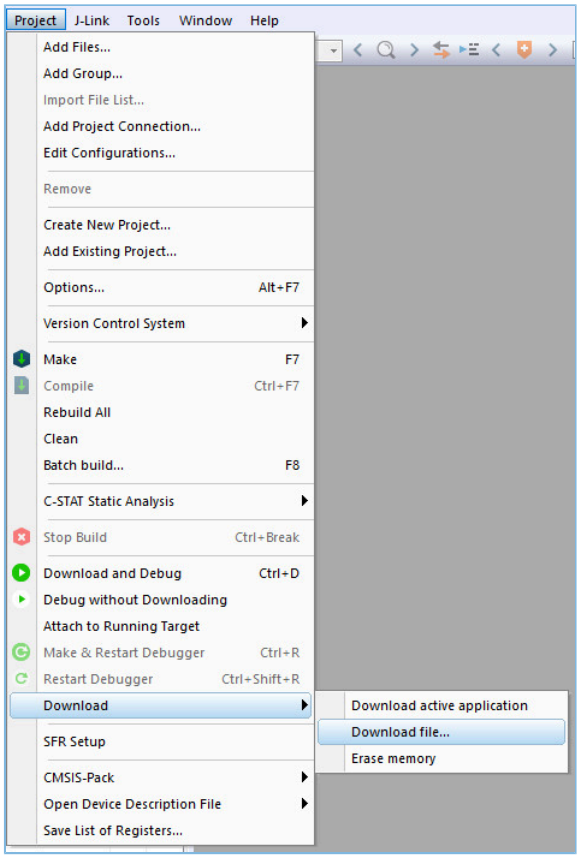

Figure 3-12 Choosing Download file

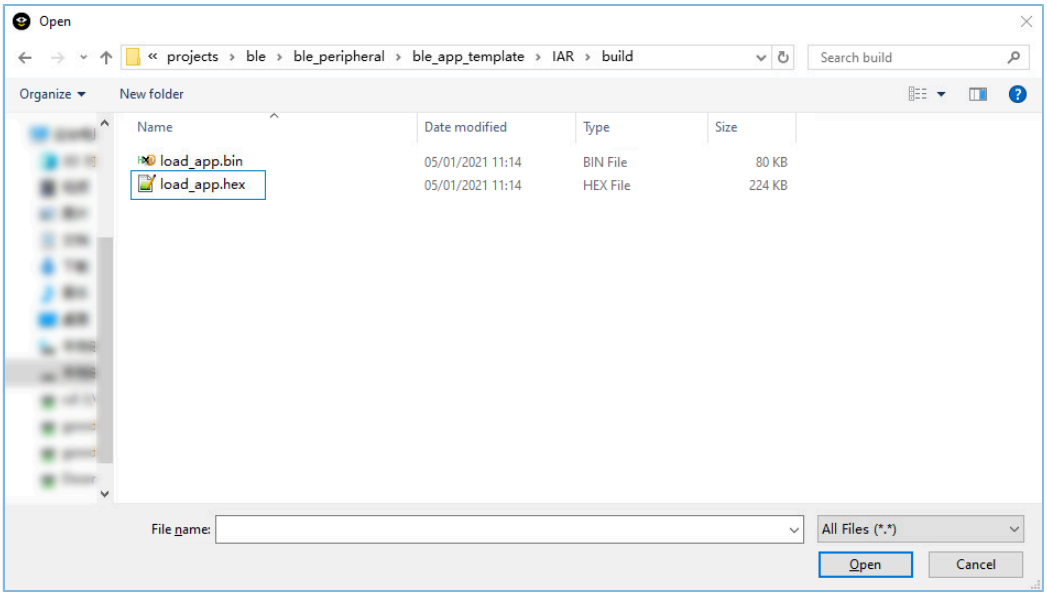

Figure 3-13 Selecting load_app.hex

3. Choose the device to be connected to for the first download operation of firmware on the current project. When the dialog box shown below pops up, click **OK** to choose the device.

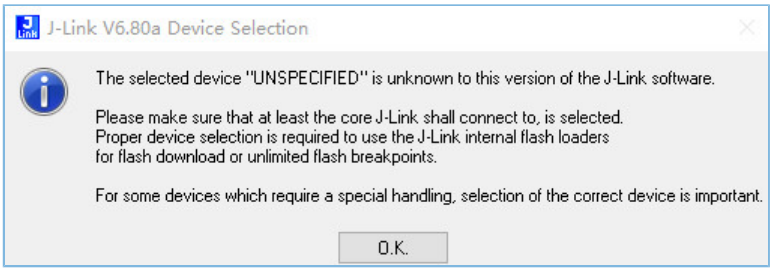

Figure 3-14 J-Link device selection window

4. Choose the device **Cortex-M4**, and click **OK** to start download.

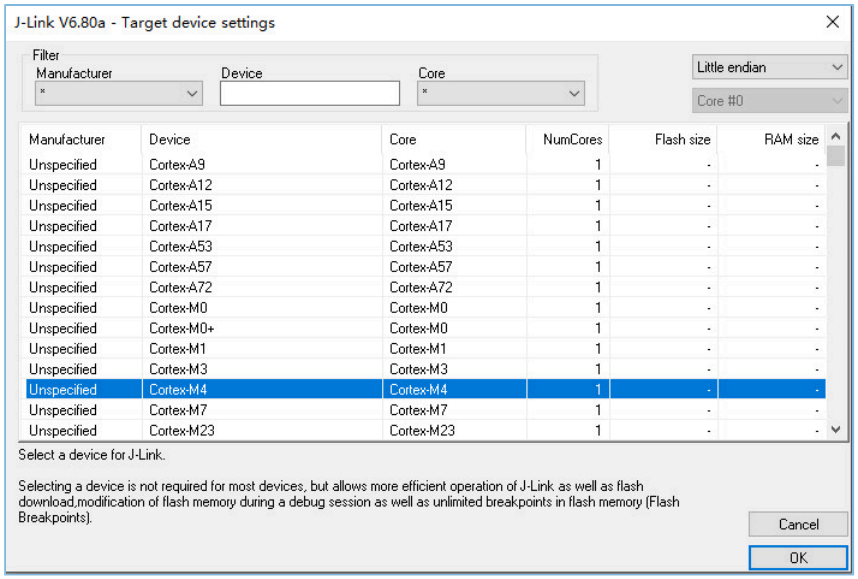

Figure 3-15 Selecting “Cortex-M4”

5. Press the **RESET** button on the GR55xx board after download completes, to start to run the firmware on the SoC.

3.4 Project Debugging

Before project debugging, make sure the project firmware is programmed to the SoC by following the instructions above.

After the firmware is programmed to the SoC, click **Debug** on the menu bar to enter the debug mode. The program will run to the main() function line.

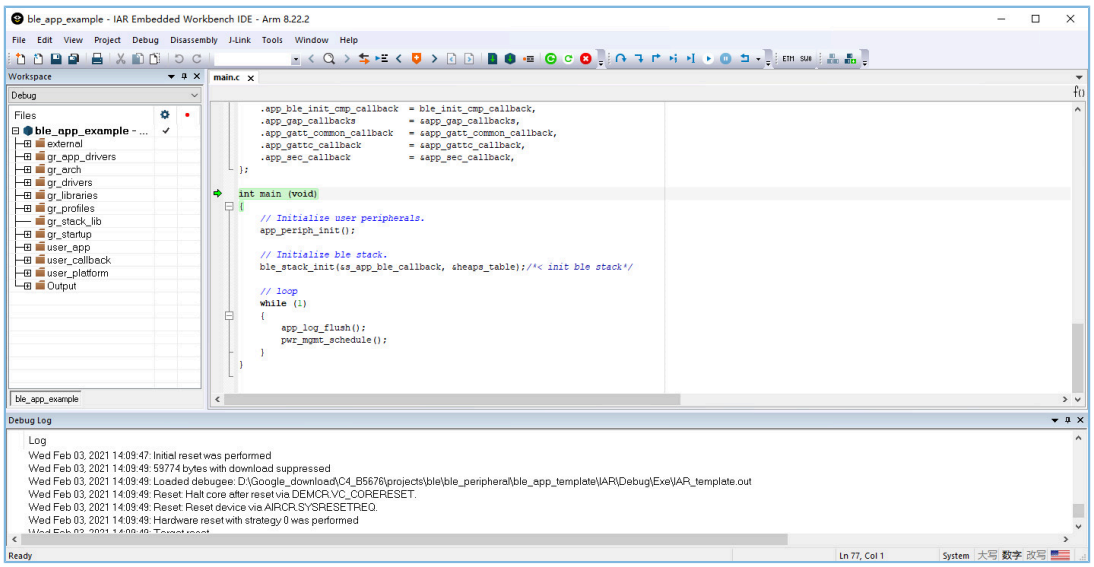

Figure 3-16 Entering the debug mode

In the debug mode, users can debug the project by breakpoints and single stepping.

4 FAQ

This chapter describes possible problems, reasons, and solutions when using an IAR example.

4.1 Why does an “Undefined Variable” Error Occur During Project Compilation?

- Description

  When I try to compile the an IAR project, an “undefined variable” error occurs.

- Analysis

  This may be because some project macros (such as GR5515_SK and APP_DRIVER_USE_ENABLE) have not been defined prior to project compilation, and thus the corresponding contents in the source file are not compiled. Users can check whether there is any macro left undefined by comparing the project macros in Keil.

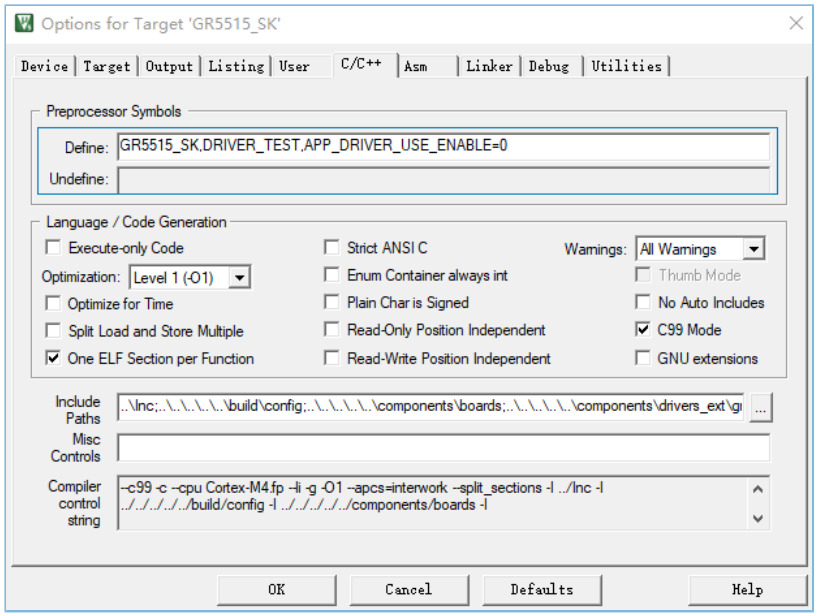

Figure 4-1 Project macro configuration in Keil

- Solution

  Add the undefined macro in the **C/C++ Compiler** > **Preprocessor** pane in IAR, as shown in the figure below.

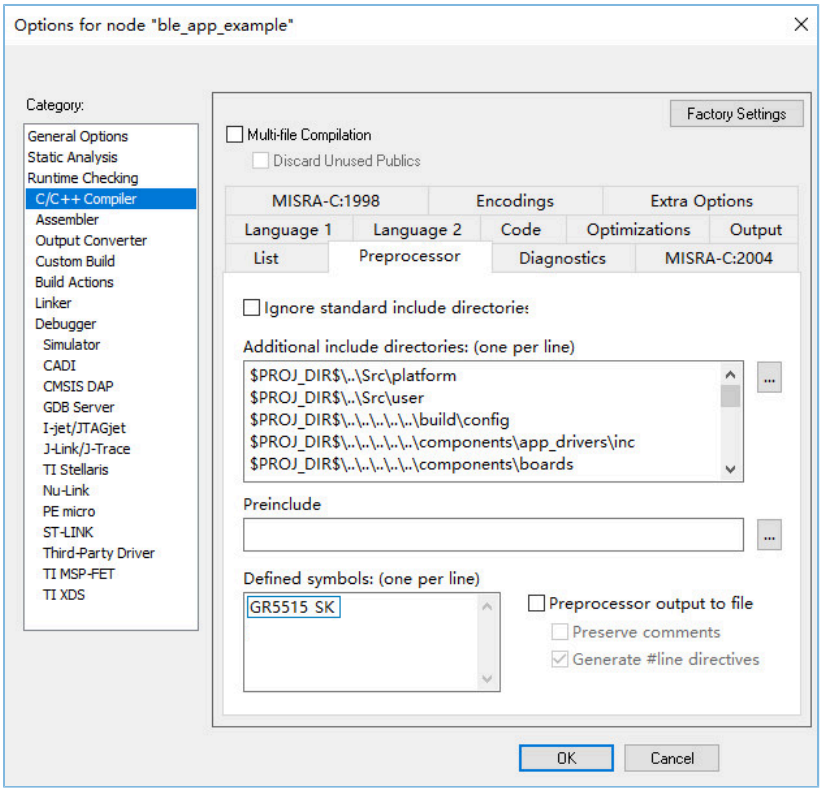

Figure 4-2 Adding an undefined macro

Copyright © 2021 Shenzhen Goodix Technology Co., Ltd.