



# User Manual

## PY32\_IAR\_DFP\_UserManual

---

### Preface

This document describes how to install the IAR Embedded Workbench (EW) IDE and the corresponding Puya IAR pack. Then, it starts from scratch to create a new project to output "Hello World" using the serial port, and to port and debug from the existing Microcontroller Development Kit (MDK) Version 5 - Keil project. Finally, we will add the functions that may be used in the process of using IAR.

PUYA CONFIDENTIAL

## Catalog

<b>1</b>	<b>Installation of IAR EW for Arm 9.20.2.....</b>	<b>3</b>
1.1	Software acquisition, download .....	3
1.2	IAR Software Installation .....	3
1.3	IAR Pack Installation.....	7
<b>2</b>	<b>Software Use .....</b>	<b>9</b>
2.1	New project.....	9
2.2	Transplantation project .....	18
2.3	Common Functions.....	20
<b>3</b>	<b>Version History.....</b>	<b>23</b>

# 1 Installation of IAR EW for Arm 9.20.2

## 1.1 Software acquisition, download

Table 1.1 - 1 Software Download Links

Software	Description	Download Links
IAR	IAR EW for Arm 9.20.2	<a href="https://www.iar.com/products/architectures/arm/iar-embedded-workbench-for-arm/">https://www.iar.com/products/architectures/arm/iar-embedded-workbench-for-arm/</a>
IAR Pack	Puya.IAR.DFP.V01	<a href="#">Tool &amp; Software - Puya Semiconductor/Download - Code Cloud - Open Source China (gitee.com)</a>

## 1.2 IAR Software Installation

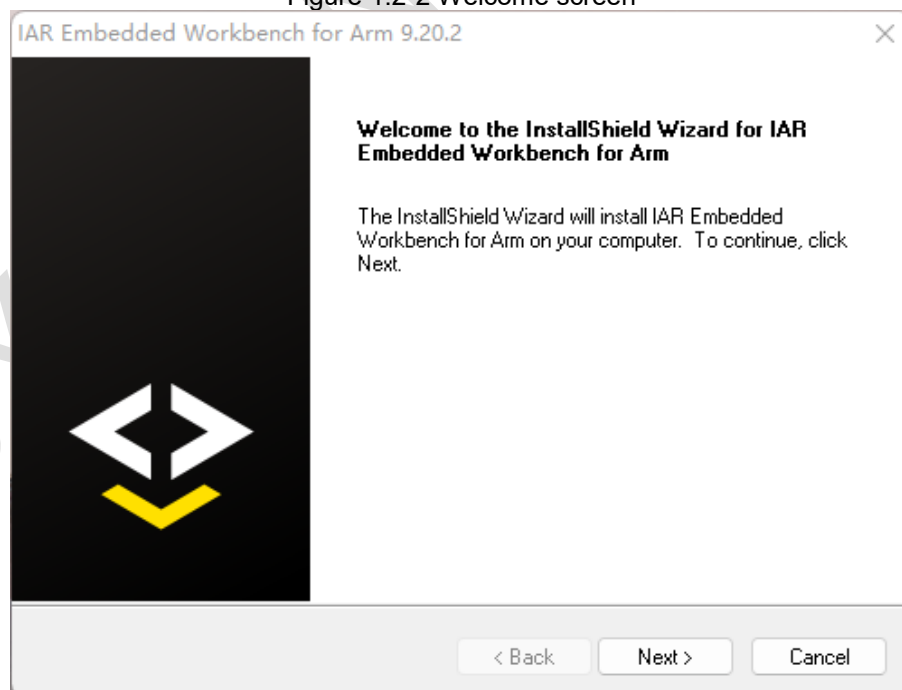
Once the IAR installation package has been downloaded, click on the installer.

Figure 1.2-1 Installation procedure



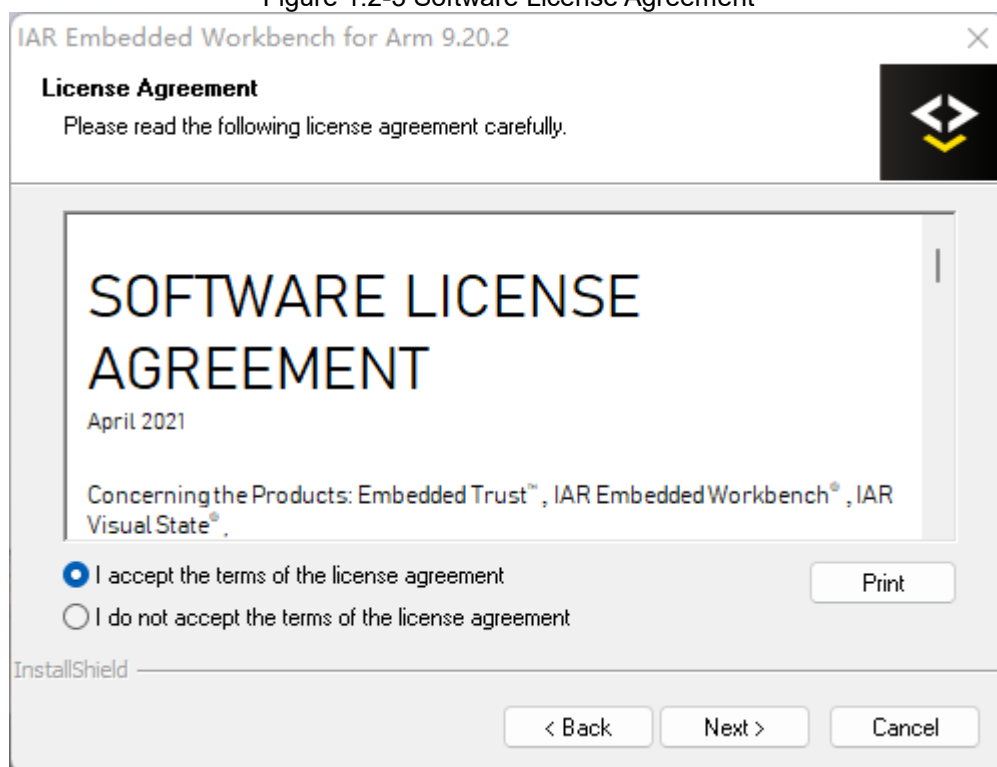
The installation screen in Figure 1.2-2 below will pop up, click the "Next" button to continue the installation.

Figure 1.2-2 Welcome screen



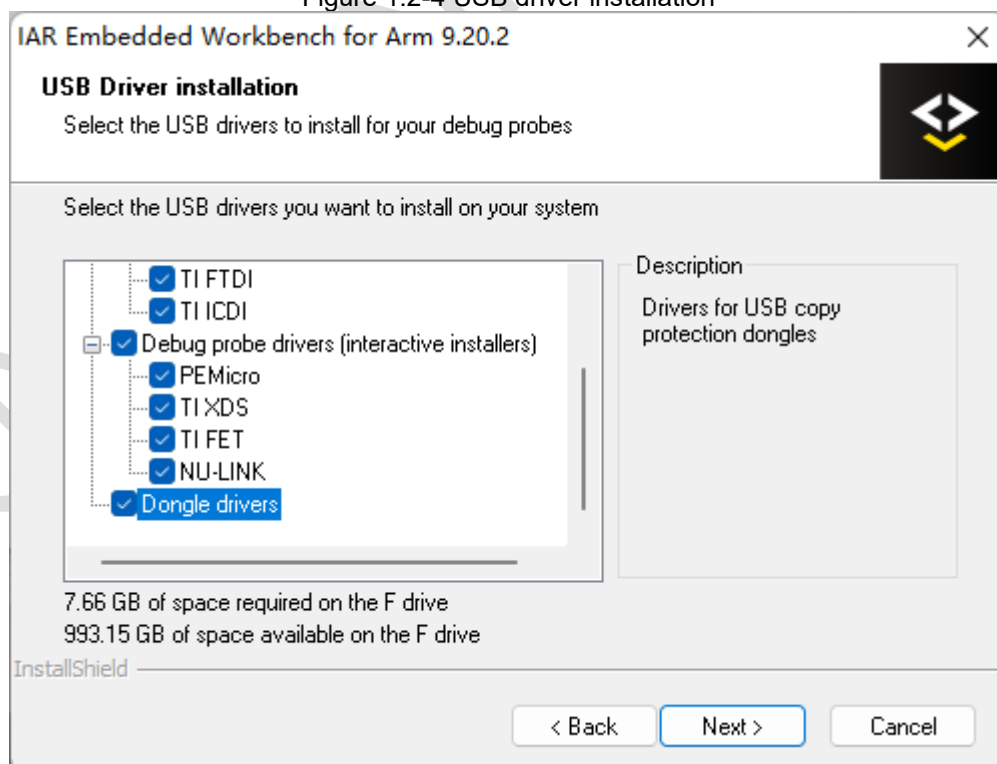
You need to check the agree agreement before proceeding to the next step of installation.

Figure 1.2-3 Software License Agreement



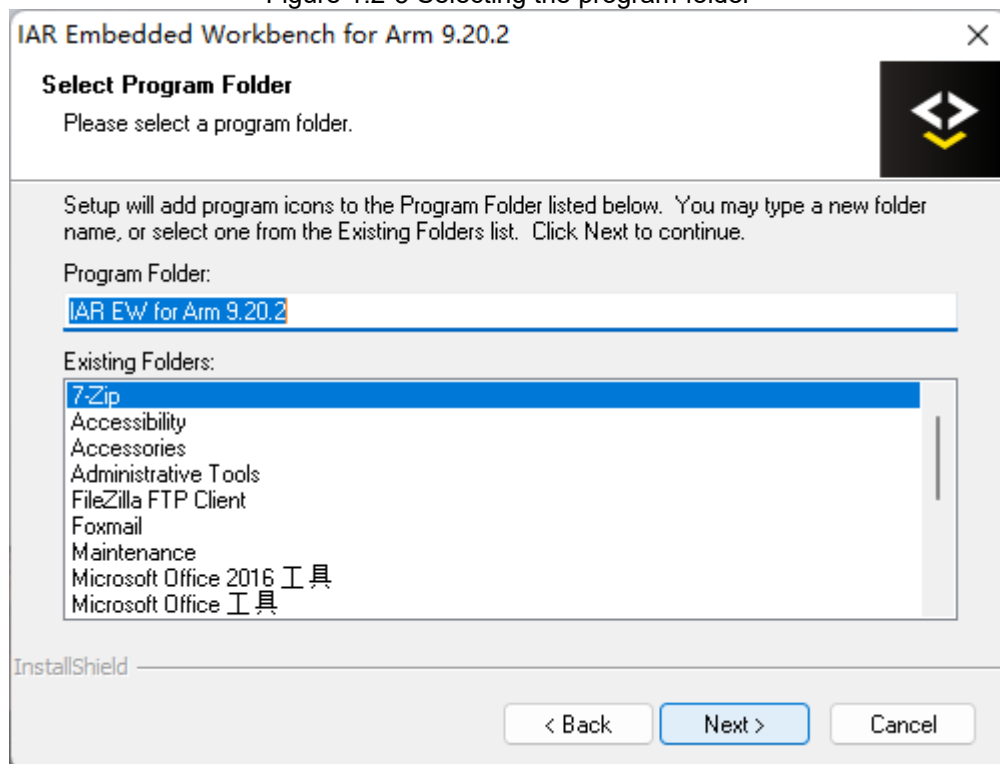
Select the appropriate USB driver to install as required.

Figure 1.2-4 USB driver installation



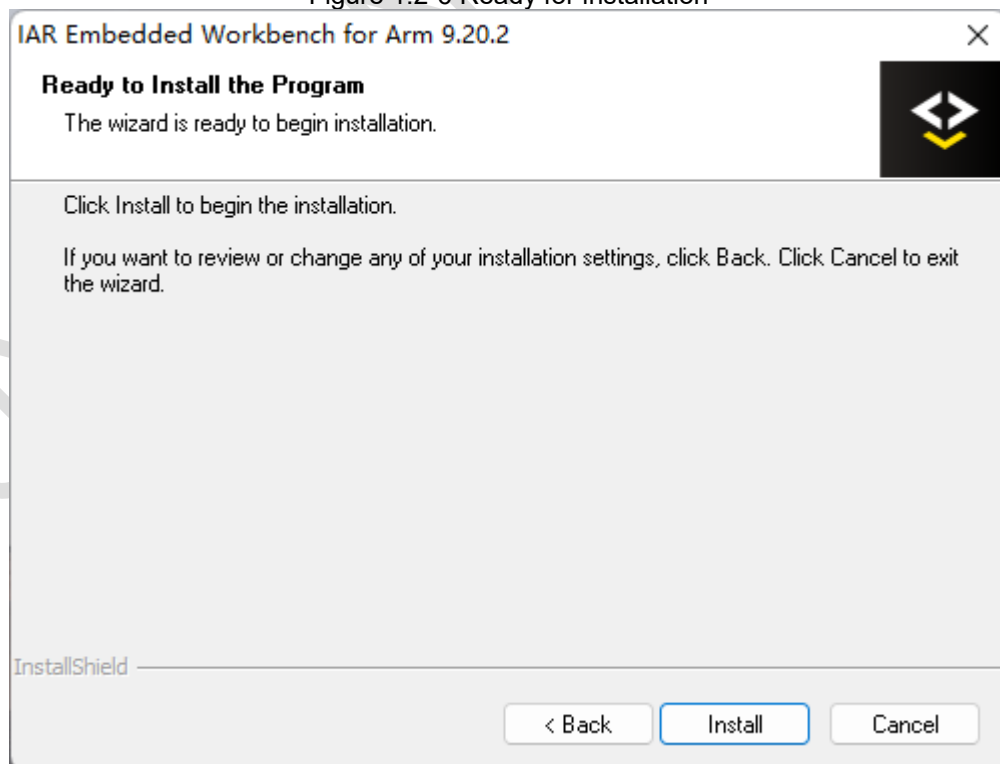
Create a folder in the start menu and click "Next" to proceed to the next step of the installation.

Figure 1.2-5 Selecting the program folder



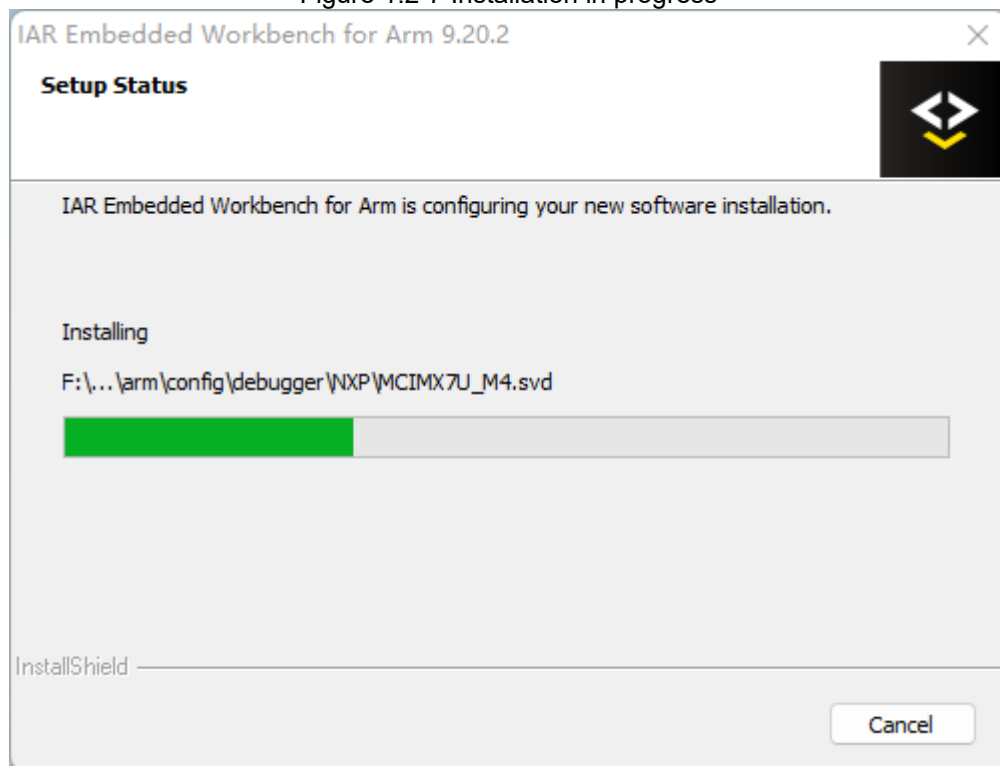
The installer is ready to install the program, click "Next" to proceed to the next step of the installation.

Figure 1.2-6 Ready for installation



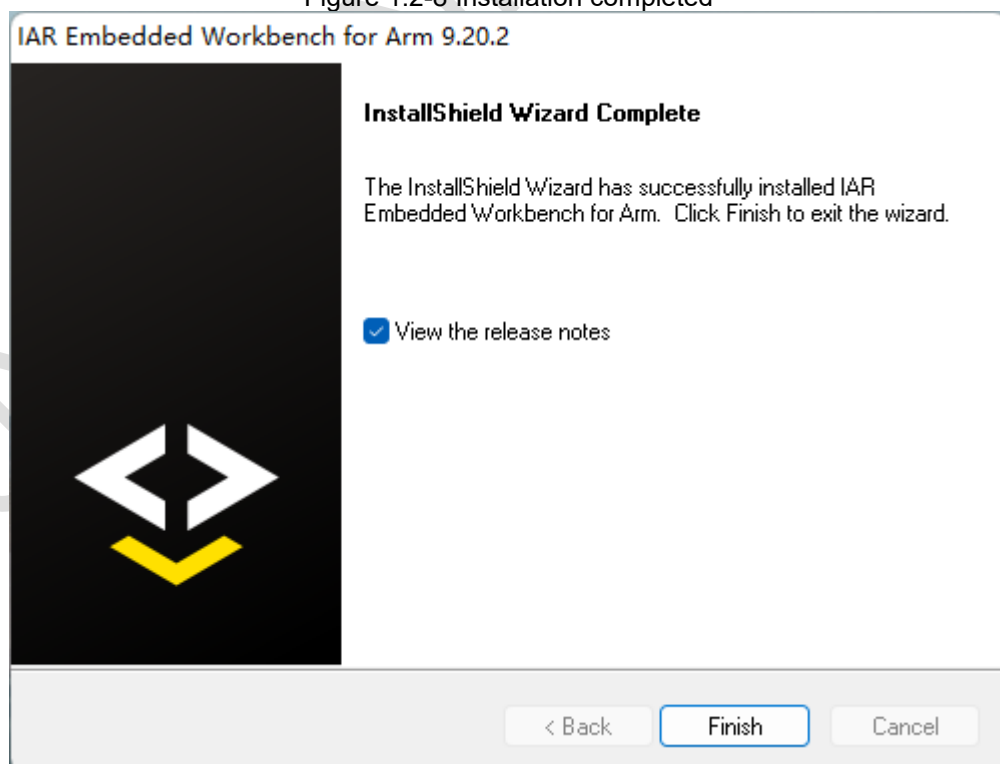
Wait for the installer to install.

Figure 1.2-7 Installation in progress



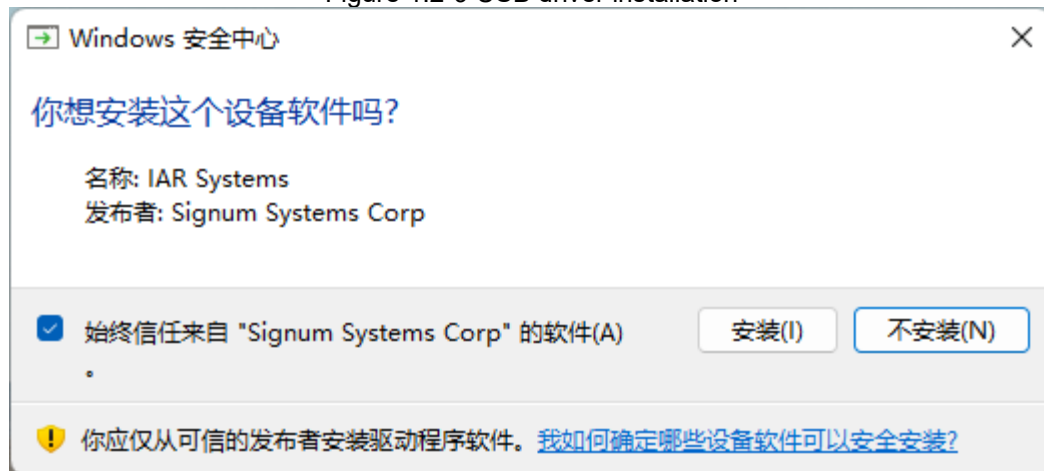
Installation completed

Figure 1.2-8 Installation completed



After the installation is complete, the USB driver will be installed, a new window will pop up, click the "Install" button  
Perform the installation.

Figure 1.2-9 USB driver installation



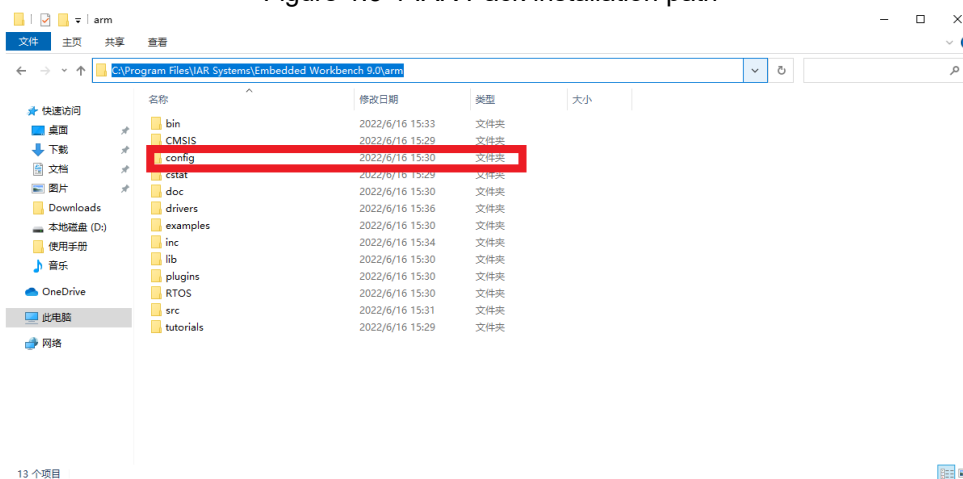
### 1.3 IAR Pack Installation

Extract the IAR Pack zip file.

Unzip the following files, and copy the "config" folder as shown in Figure below to the arm folder in the IAR installation directory. For example: C:\Program Files\IAR Systems\Embedded Workbench 9.0\arm.

Replace the "config" folder in the red box in Figure below.

Figure 1.3-1 IAR Pack installation path



Select "Replace files in target".

Figure 1.3-2 Replace or skip file options



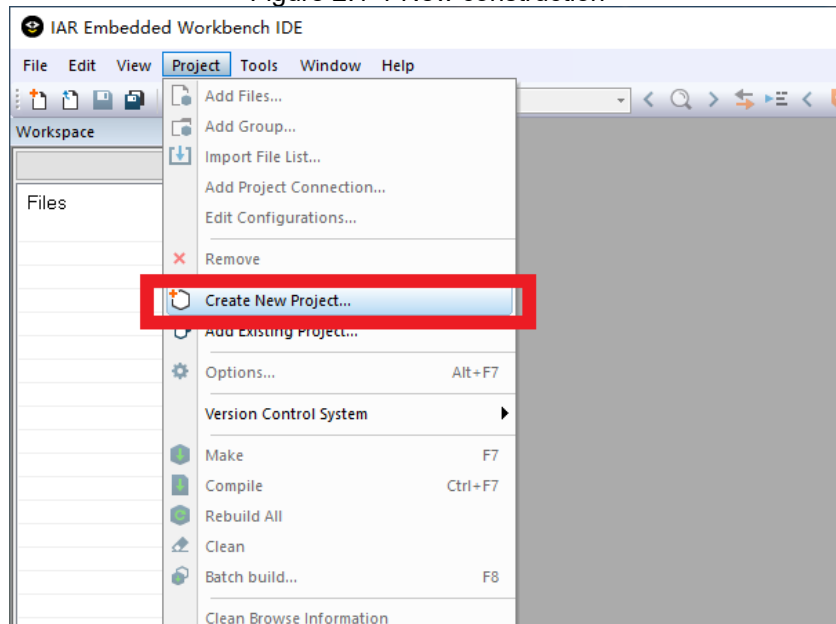


## 2 Software Use

### 2.1 New project

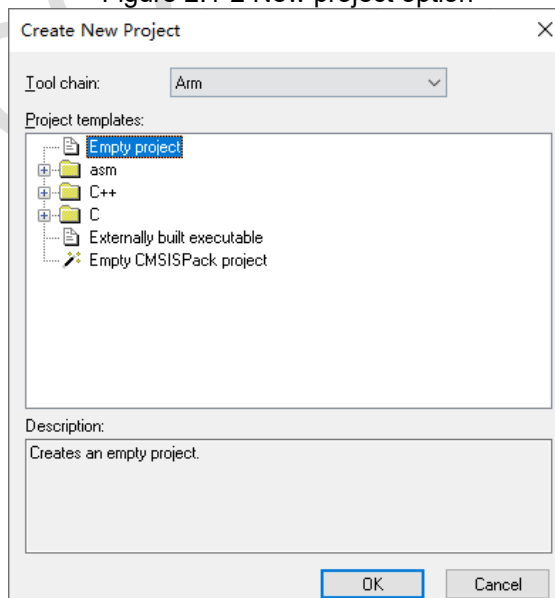
Open the IAR EW for Arm software and click on the menu bar "Project" -> "Create New Project".

Figure 2.1-1 New construction



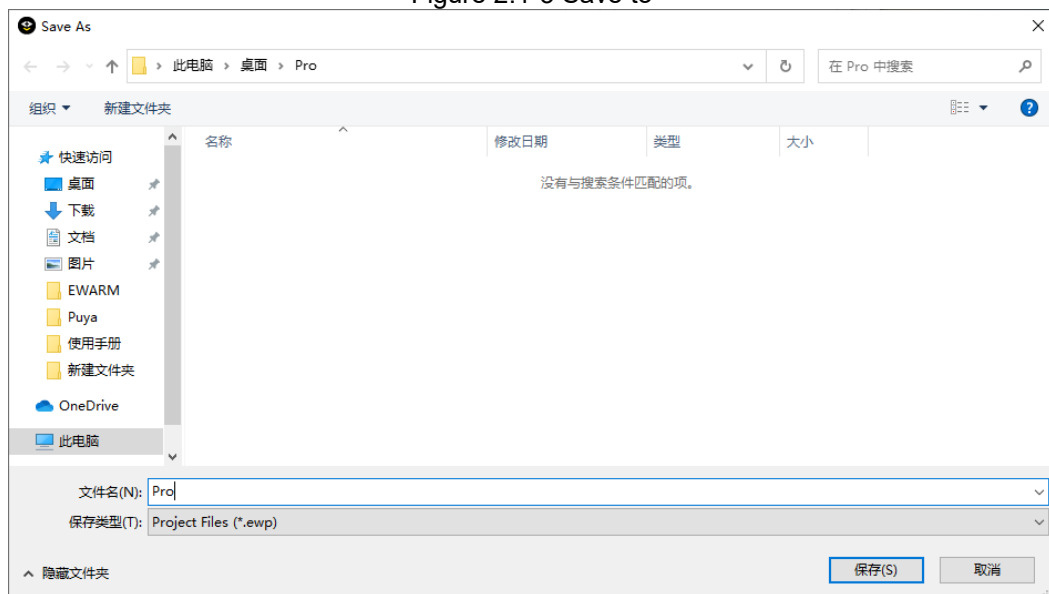
In the new window, select "Empty project" (if you select C/C++, it will have main.c in the new project) and click "OK".

Figure 2.1-2 New project option



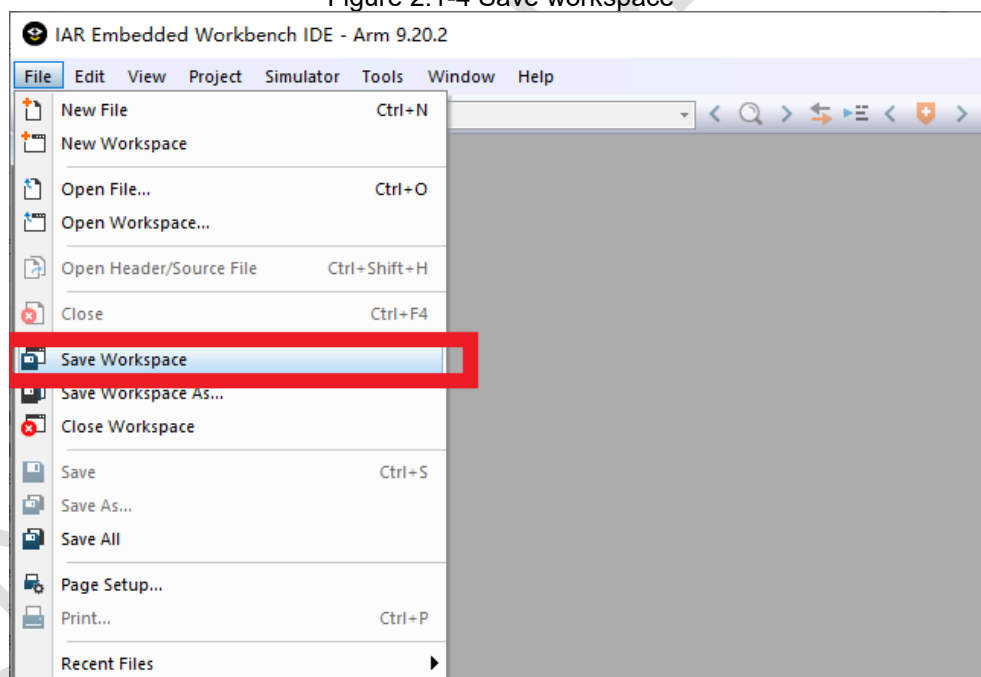
Next, the Save window pops up and select the appropriate path to save.

Figure 2.1-3 Save to



Then click "File"->"Save Workspace" to save the workspace.

Figure 2.1-4 Save workspace



After saving in the workspace, there will be a file with the project name .eww in the project folder, which can be clicked directly to open the EWARM project.

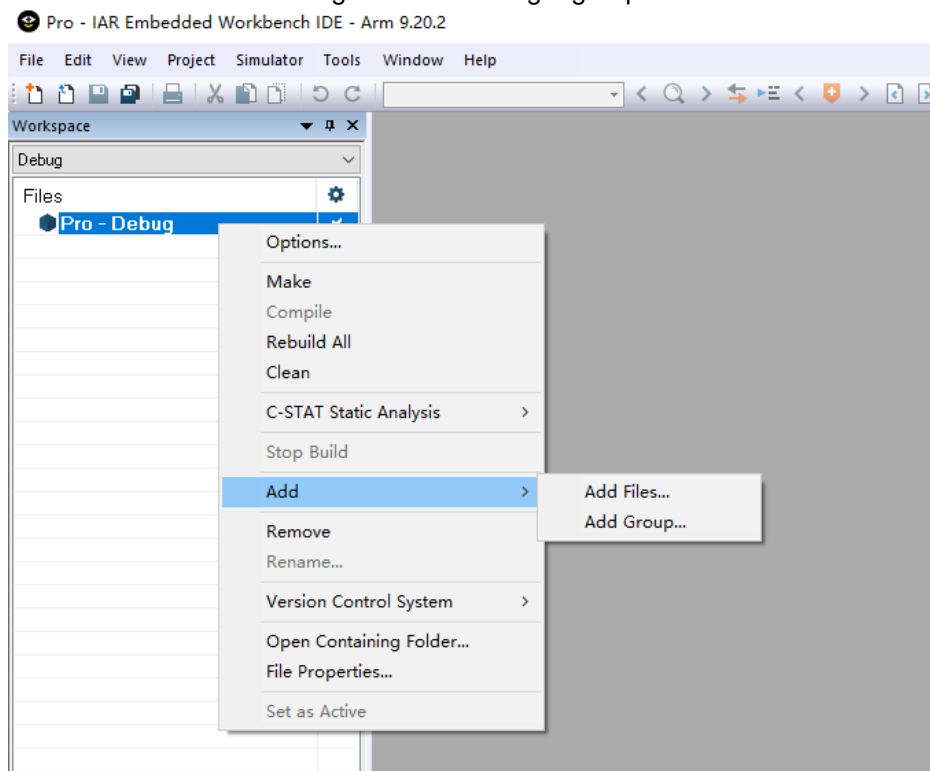
Figure 2.1-5 Project Icon



After the above steps are completed, there will be a project file in the software, right-click on the project file and select "Add" to add the file. (Add File for adding files, Add Group for creating

folders)

Figure 2.1-6 Adding a group



Click Add Group and enter the name of the folder in the new pop-up window to create the folder in the following figure. The Output folder does not need to be created, it will be generated automatically when compiling.

Figure 2.1-7 Enter group name

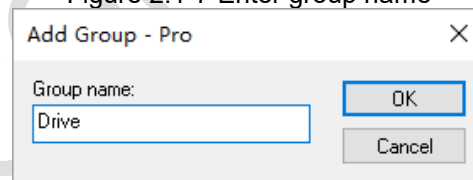
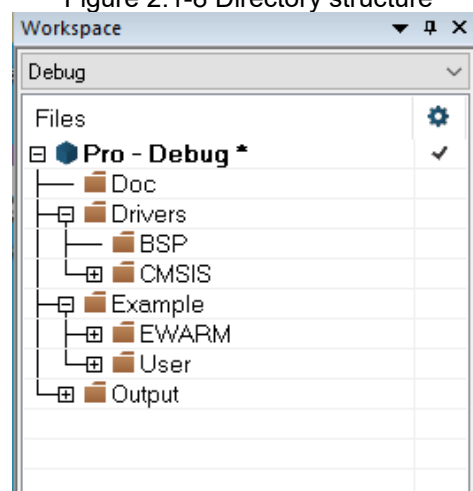
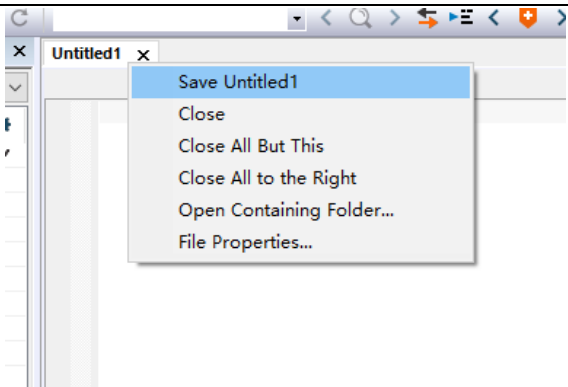


Figure 2.1-8 Directory structure



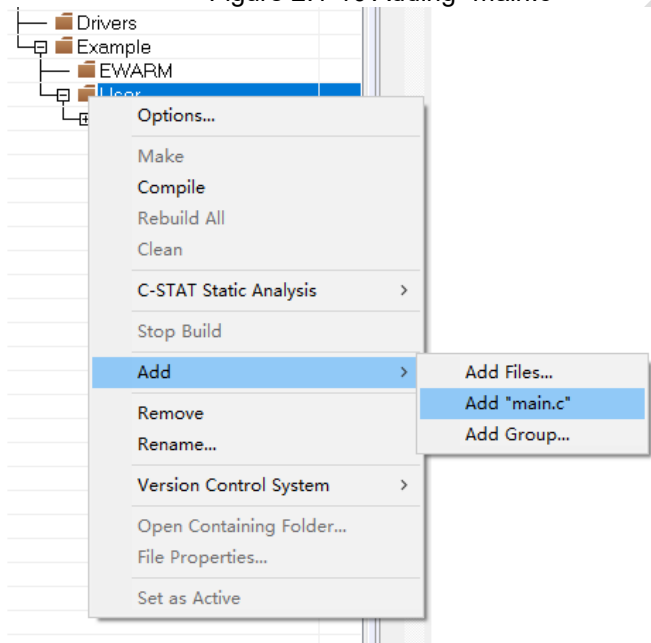
Click the menu bar File->New File, right-click the title of the new file, name it main.c and save it .

Figure 2.1-9 Create a new file and save it



After the save is complete, right-click the User folder, Add->Add "main.c" to add the file you just saved, if you already have main.c you can click Add->Add Files to add it directly.

Figure 2.1-10 Adding "main.c"



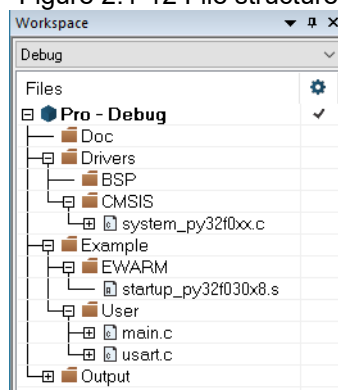
When you create a new project, you need the corresponding files, which should be copied to the project folder.

Figure 2.1-11 Required Documents

main.c	2022/6/29 14:31	C Source	1 KB
Pro.dep	2022/6/29 14:34	DEP 文件	5 KB
Pro.ewd	2022/6/29 14:01	EWD 文件	108 KB
Pro.ewp	2022/6/29 14:33	EWP 文件	74 KB
Pro.ewt	2022/6/29 14:33	EWT 文件	208 KB
	2022/6/24 18:15	IAR IDE Worksp...	1 KB
py32f0xx.h	2022/2/9 14:48	C/C++ Header	7 KB
py32f030x8.h	2022/3/16 11:10	C/C++ Header	366 KB
startup_py32f030x8.s	2022/3/7 14:42	Assembler Source	11 KB
system_py32f0xx.c	2022/2/9 14:50	C Source	5 KB
system_py32f0xx.h	2022/2/9 14:49	C/C++ Header	4 KB
usart.c	2022/6/29 14:19	C Source	3 KB
usart.h	2022/6/29 11:27	C/C++ Header	2 KB

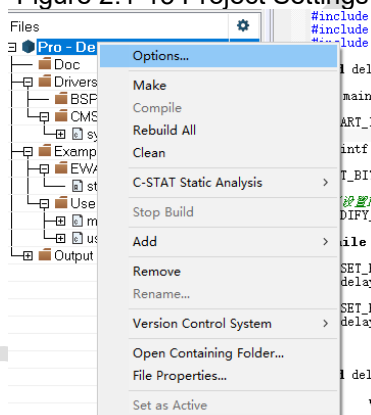
The "startup\_py32f030x8.s" file should be added to the Example->EWARM folder (Note: the startup file of EWARM is different from that of MDK, which is explained in the next section of the porting project), and the "system\_py32f0xx.c" file should be added to Drivers->CMSIS folder, and the file structure is built as below.

Figure 2.1-12 File structure



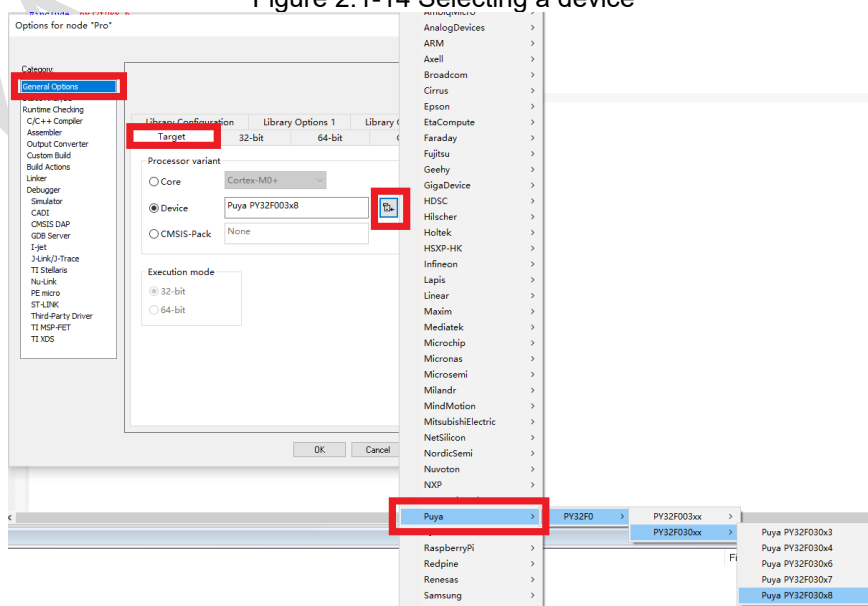
Right-click the project and select "Options" to set it.

Figure 2.1-13 Project Settings



You can select the corresponding device by selecting "General Options->Target->Device" in the settings. If there is no Puya option in the list, please refer to the previous article on the Pack installation process to install or reinstall.

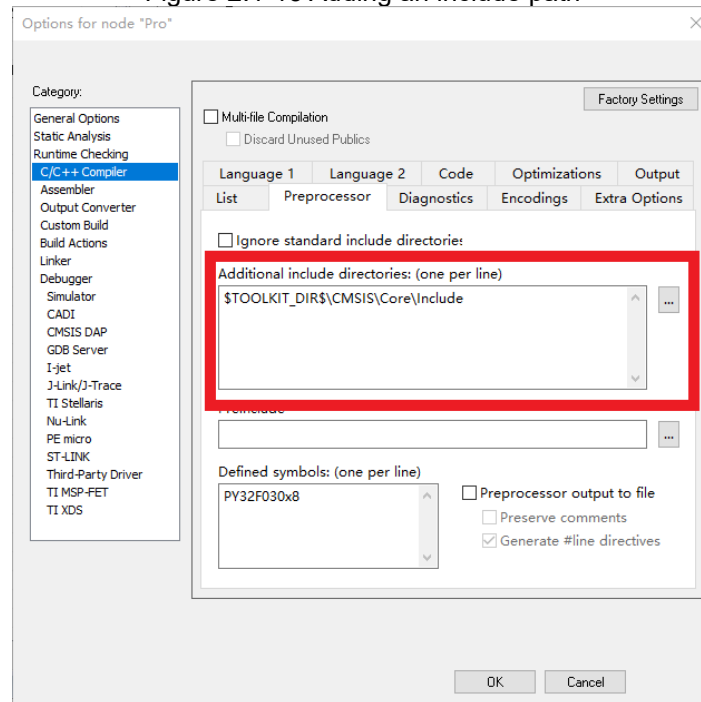
Figure 2.1-14 Selecting a device



In the setting "C/C++ Compiler>Preprocessor->Additional include directories:(one per line)", you

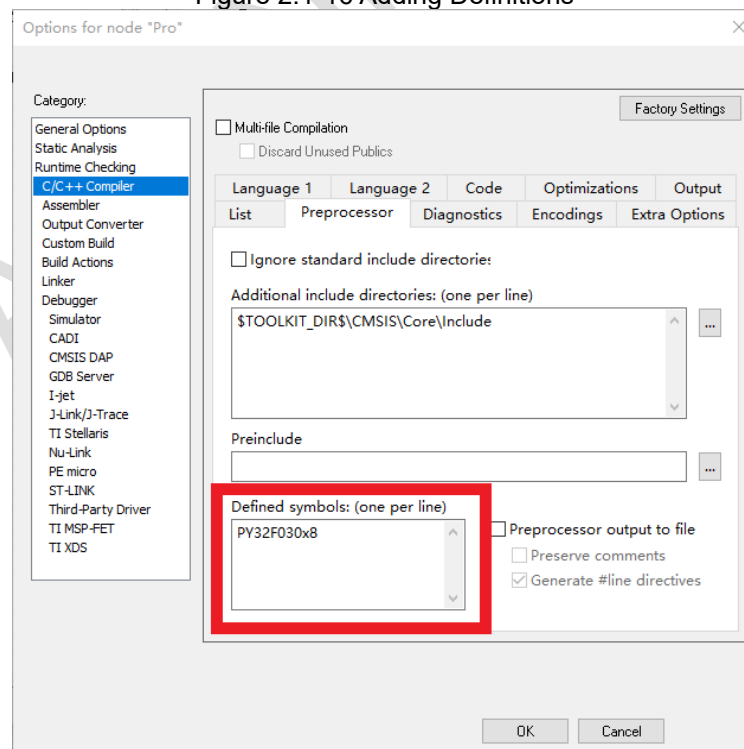
can add the include path. "\$TOOLKIT\_DIR\CMSIS\Core\Include" is the core include in the installation directory.

Figure 2.1-15 Adding an include path



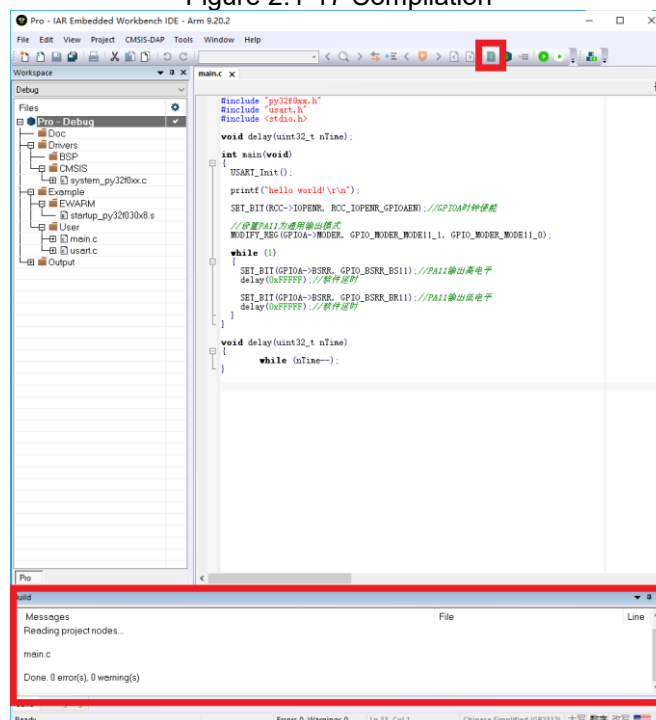
In "C/C++ Compiler>Preprocessor->Defined symbols:(one per line)" setting, you can add definitions.

Figure 2.1-16 Adding Definitions



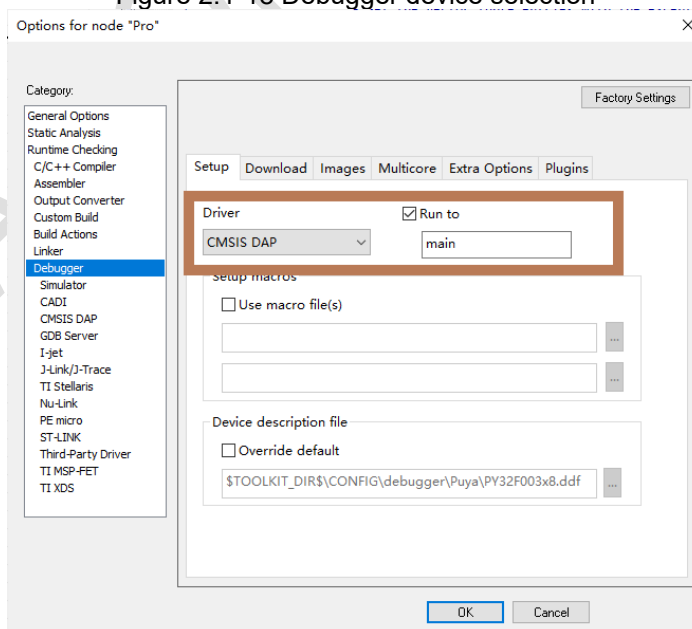
After setting all the files, paths and definitions you can compile, click the button in the red box to compile. The message window indicates Done. 0 error(s), 0 warning(s)

Figure 2.1-17 Compilation



Next, download the program to the MCU for testing, connect the PY32F030xx8 to the PY-LINK via SWD, connect the PA9 pin to the RX pin of the UART, and connect the PY-LINK emulator to the PC. In "Debugger>Setup->Driver", select "CMSIS DAP" and check "Run to main" on the right side of the project. "

Figure 2.1-18 Debugger device selection



When using PY-Link connection, you need to set "CMSIS DAP -> Interface -> Interface" to 1MHz in the settings to download, otherwise an error window may pop up.

Figure 2.1-19 Interface frequency selection

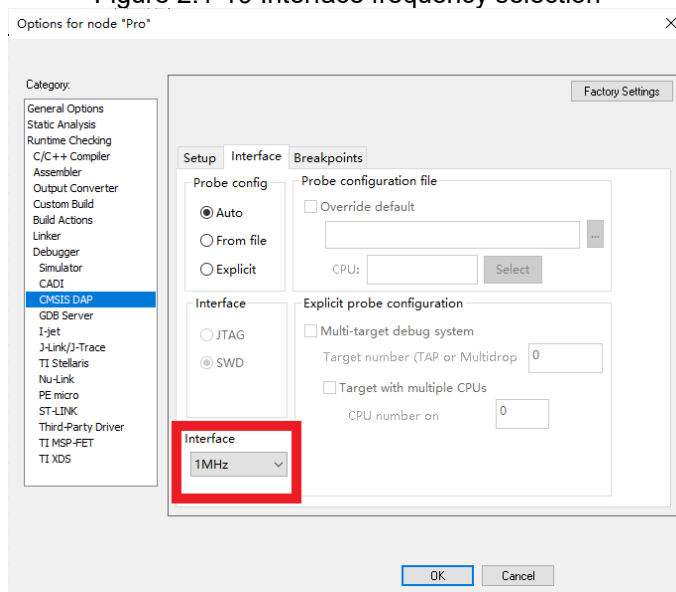
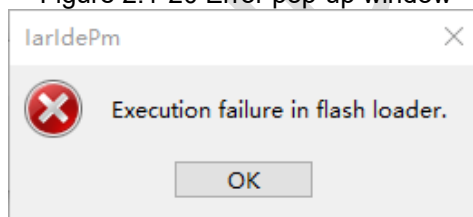
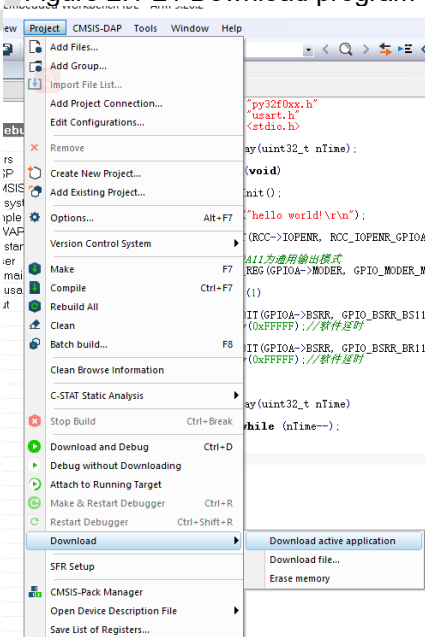


Figure 2.1-20 Error pop-up window



After setting, click "Project->Download-> Download active application" to download. (If you use this function often, please refer to section 2.3.3 and place this function in the toolbar.)

Figure 2.1-21 Download program




After successful download, open the serial port tool on your computer. Since the routine specifies the checksum method, baud rate and other information in the file, you need to select the same



options as the file in the tool.

Figure 2.1-22 Serial Receive Options



The dialog box for serial receive options. It contains several settings: 'COM6: USB-SERIAL' is selected in a dropdown; '波特率' (Baud rate) is set to 115200; '停止位' (Stop bits) is set to 1; '数据位' (Data bits) is set to 8; '校验位' (Parity) is set to Even; and '串口操作' (Serial port operation) has a radio button selected next to '关闭串口' (Close serial port).

COM6: USB-SERIAL	▼
波特率	115200 ▼
停止位	1 ▼
数据位	8 ▼
校验位	Even ▼
串口操作	<input checked="" type="radio"/> 关闭串口

Press the Reset button of the development board, the serial tool displays the following figure.

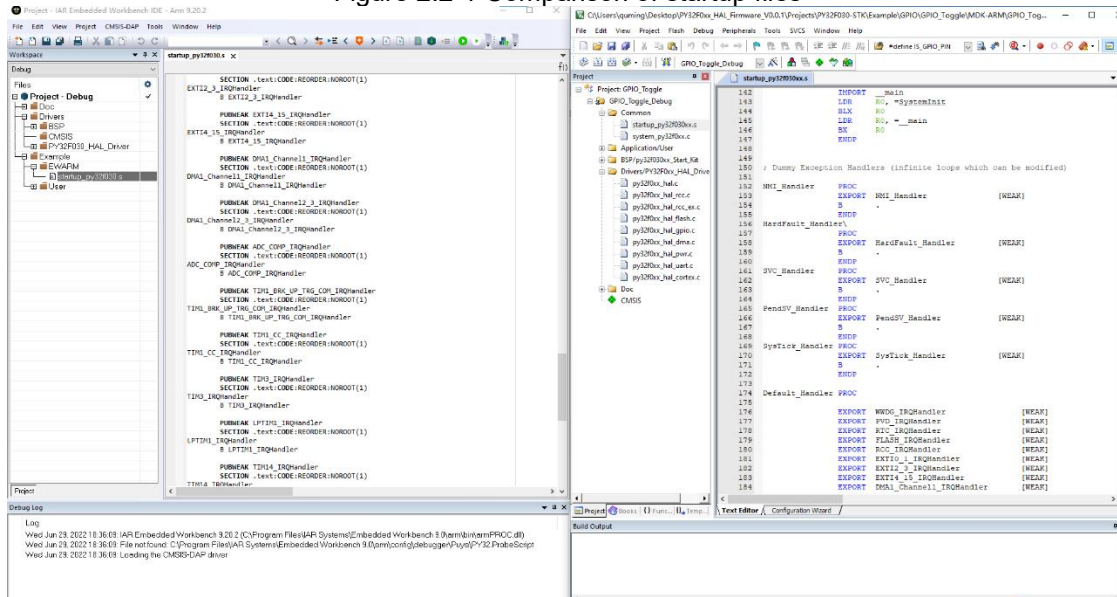
Figure 2.1-23 Serial port output

hello world!

## 2.2 Transplantation project

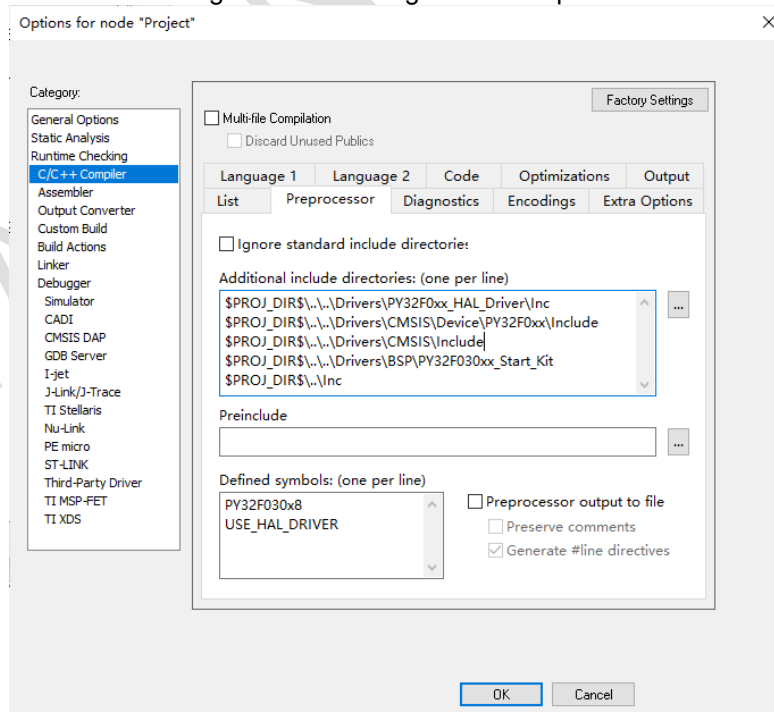
This section briefly describes porting a project from MDK, most of the operations are the same as the new project in the previous section, you need to add files to the project, but the way to write the startup file is different, especially the way to write interrupts, which should be modified when porting.

Figure 2.2-1 Comparison of startup files



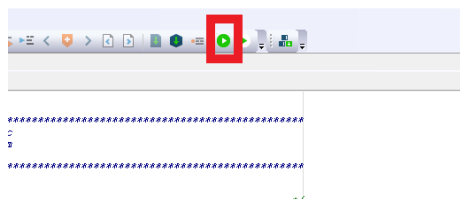
In the project settings "C/C++ Compiler>Preprocessor->Additional include directories:(one per line)" you can add include paths, such as: "\$PROJ\_DIR\$...\Drivers\CMSIS\Include".

Figure 2.2-2 Adding an include path



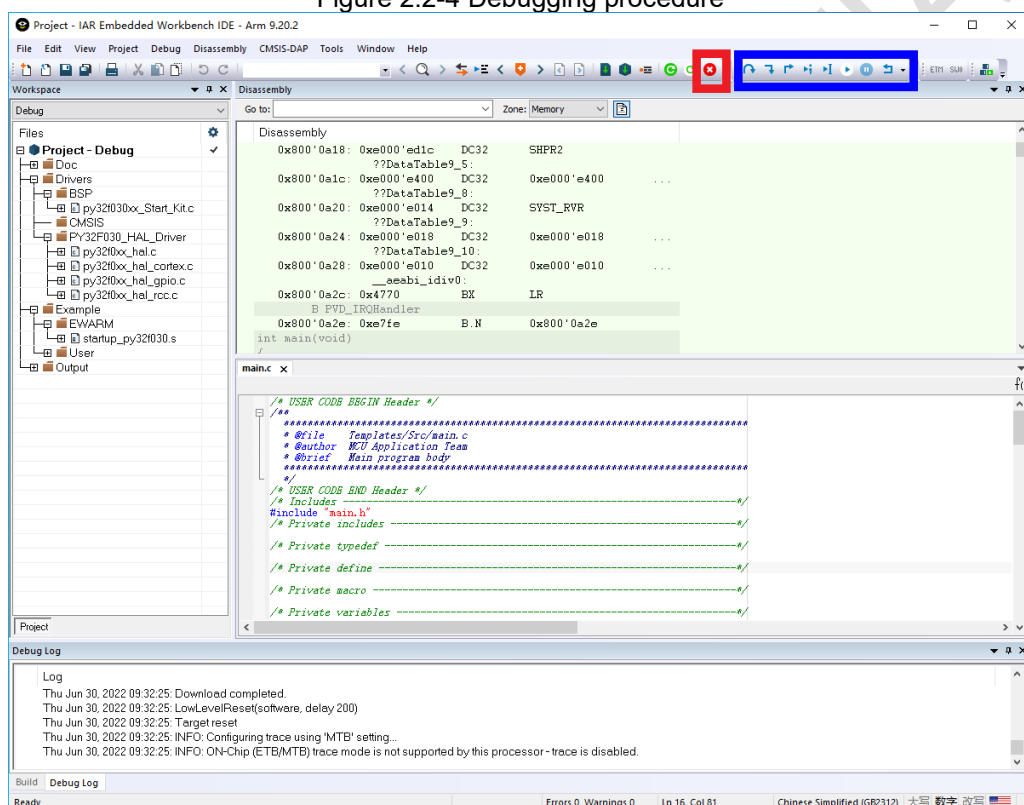
After successful compilation, you can enter debugging mode and click the button in the red box in the figure to debug.

Figure 2.2-3 Debug button



The interface during debugging is shown below. The buttons in the red box can end the simulation, and the buttons in the blue box can control the steps of the simulation (Single Step, Full Speed, Reset, etc.).

Figure 2.2-4 Debugging procedure

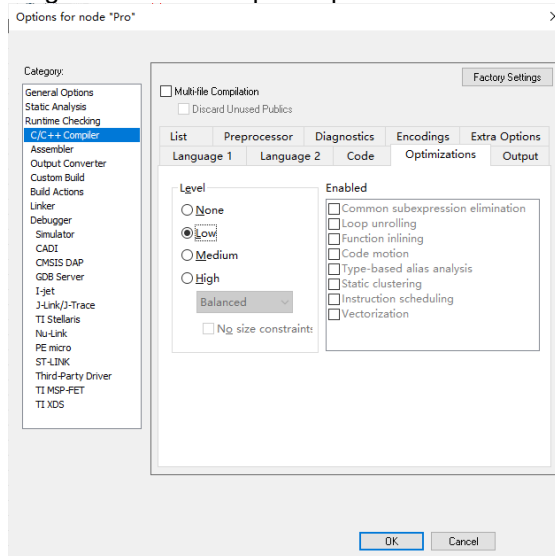


## 2.3 Common Functions

### 2.3.1 Compiler Optimization

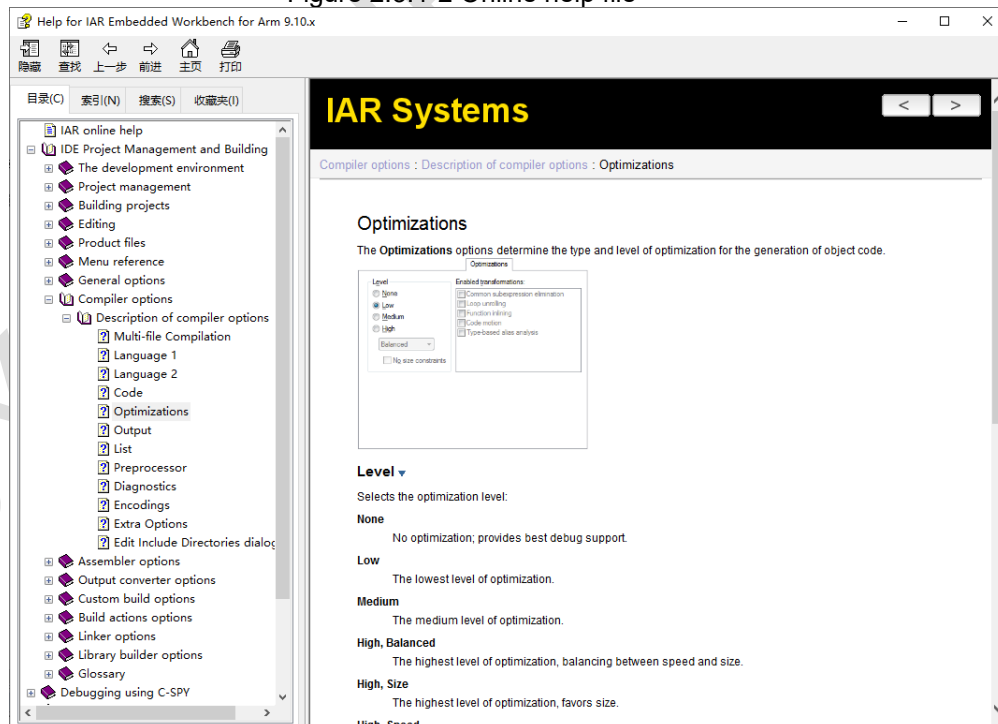
The project settings "C/C++ Compiler>Optimizations->Level" allows you to adjust the optimization level of the compiler.

Figure 2.3.1-1 Compiler Optimization



For details, please refer to the instructions in the IAR online help, which can be found in the menu bar "Help->Content", by following the path shown below.

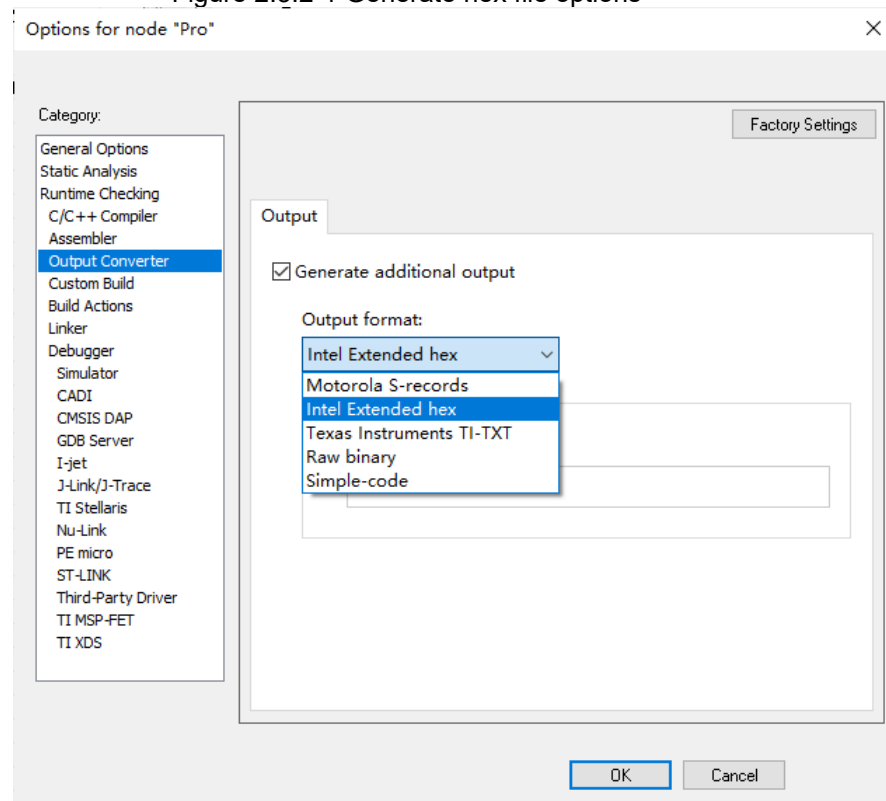
Figure 2.3.1-2 Online help file



### 2.3.2 Generate hex file

In the project settings "Output Converter> Output", you can check "Generate additional output", and in the following "Output format" select "Intel Extended hex" to generate a ".hex" file at compile time, the generated file is located under "\Debug\Exe" path in the project folder.

Figure 2.3.2-1 Generate hex file options



### 2.3.3 Customize Toolbar

Click the small triangle on the rightmost side of the toolbar, as shown in the figure, and click "Add or Remove Button -> Customize" in order to pop up the customization window, as shown in the figure below.

Figure 2.3.3-33 Customizing the toolbar

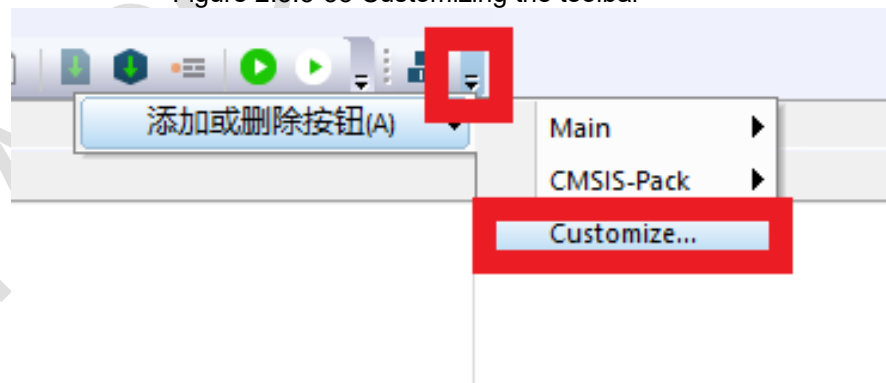
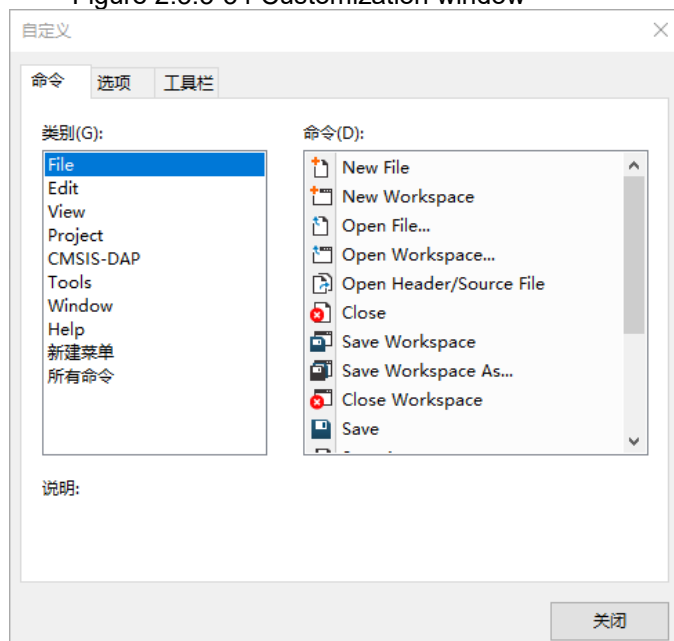
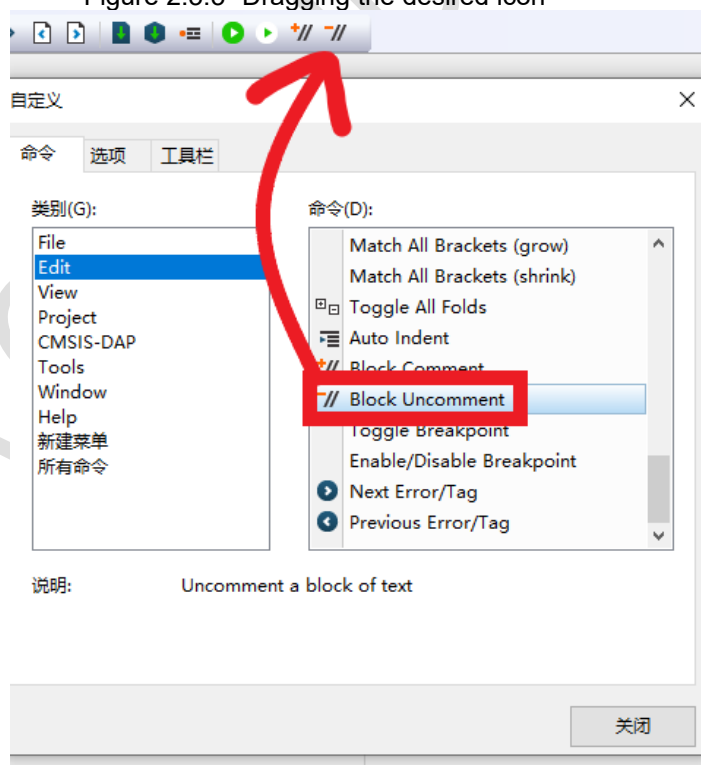


Figure 2.3.3-34 Customization window



Find the icon you need to use, and drag it into the toolbar by holding down the mouse to fix the icon to the toolbar, as shown in the figure.

Figure 2.3.3- Dragging the desired icon



3 Version History

Version	Date	Description
V1.0	2024-06-12	Initial version



Puya Semiconductor Co., Ltd.

**IMPORTANT NOTICE**

Puya Semiconductor reserves the right to make changes without further notice to any products or specifications herein. Puya Semiconductor does not assume any responsibility for use of any of its products for any particular purpose, nor does Puya Semiconductor assume any liability arising out of the application or use of any of its products or circuits. Puya Semiconductor does not convey any license under its patent rights or other rights nor the rights of Puya Semiconductor does not convey any license