

W55MH32 用戶手册

Version 1.0.0

目 录

1 开发板介绍	5
1.1 W55MH32L-EVB	5
1.1.1 简介	5
1.1.2 板载资源	6
1.2 W55MH32Q-EVB	7
1.2.1 简介	7
1.2.2 板载资源	8
2 安装 IDE	10
2.1 温馨提示	10
2.2 获取 KEIL5 安装包	10
2.3 开始安装 KEIL5	11
2.4 安装 W55MH32 芯片包	12
3 打开示例工程	13
3.1 外设示例目录	13
3.2 TOE 引擎示例目录	14
3.3 KEIL 主界面介绍	15
3.4 常用快捷键按钮	15
4 仿真器介绍	17
4.1 仿真器简介	17
4.2 仿真器配置	18
4.3 使用其他仿真器和串口调试工具	21

插图清单

图 1 W55MH32L-EVB 开发板正面	5
图 2 W55MH32L-EVB 开发板背面	5
图 3 W55MH32L-EVB 硬件资源配置图	6
图 4 W55MH32Q-EVB 开发板正面	7
图 5 W55MH32Q-EVB 开发板背面	8
图 6 W55MH32Q-EVB 硬件资源配置图	9
图 7 Keil 官网安装包示意图	10
图 8 开始安装 KEIL5	11
图 9 安装完成 KEIL5	11
图 10 安装 pack 包	12
图 11 示例程序	13
图 12 TOE 示例程序	14
图 13 KEIL5 主界面	15
图 14 常用工具快捷键按钮	15
图 15 WIZ-Link	17
图 16 WIZ-Link 设备管理器示意图	18
图 17 Debug 选项配置	19
图 18 CMSIS-DAP Debugger 配置	20
图 19 W55MH32 跳线帽连接 WIZ-Link	21
图 20 JLINK 修改内容示意图	21
图 21 添加 W55MH32.FLM 文件	22
图 22 W55MH32.FLM 文件位置	22

列表清单

表 1 W55MH32L-EVB 硬件资源配置表	6
表 2 硬件资源配置表	9
表 3 仿真类按钮.....	15
表 4 编译类按钮.....	16

1 开发板介绍

1.1 W55MH32L-EVB

1.1.1 简介

W55MH32L-EVB 是基于 W55MH32L 芯片开发的一款开发板，主频为 216MHz，1MB 的闪存以及 96KB 的 SRAM，同时还具有一个完整的硬件 TCP/IP 卸载引擎，只需要简单的 socket 编程即可实现以太网应用。

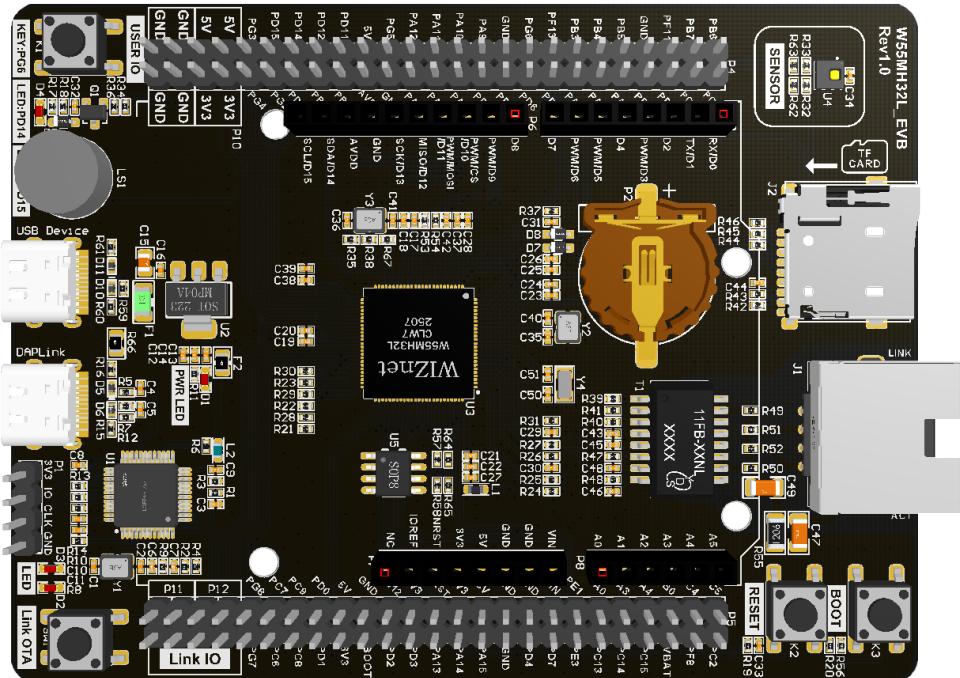


图 1 W55MH32L-EVB 开发板正面

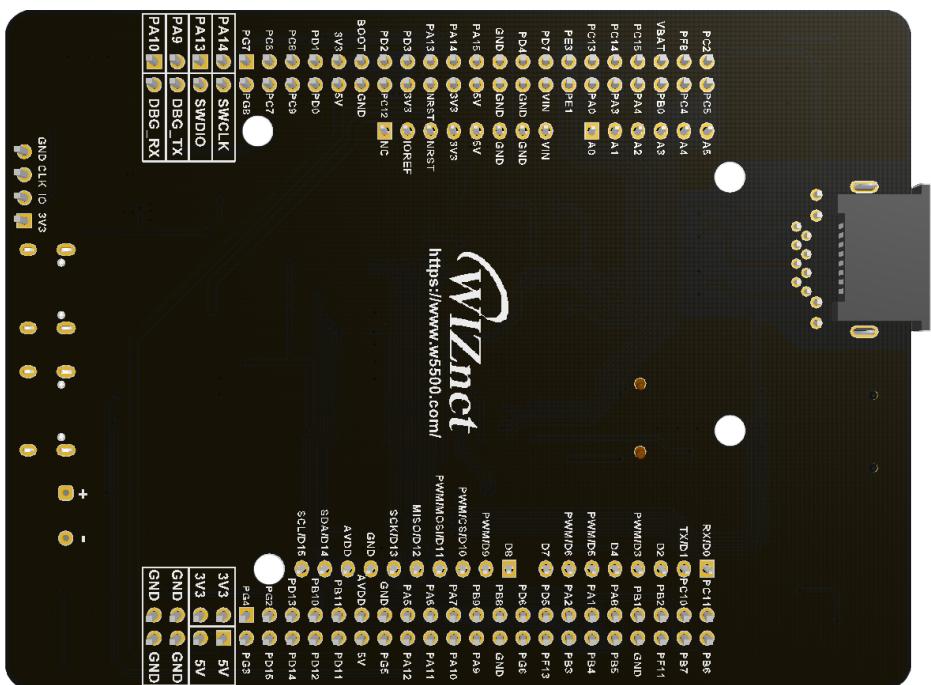


图 2 W55MH32L-EVB 开发板背面

1.1.2 板载资源

开发板上集成了一套 ARDUINO 插排，可以外接 ARDUINO 接口的扩展板，兼容性和拓展性都非常强大。

此外，开发板上还集成了一个 WIZ-Link 仿真器，支持调试，程序下载以及虚拟串口功能。再也无须外接仿真器和串口调试工具，只需一块开发板即可轻松搞定单片机+以太网部分的学习内容。

还有 TF 卡座，温湿度传感器，EEPROM，用户按钮，用户 LED 等外设，资源丰富。

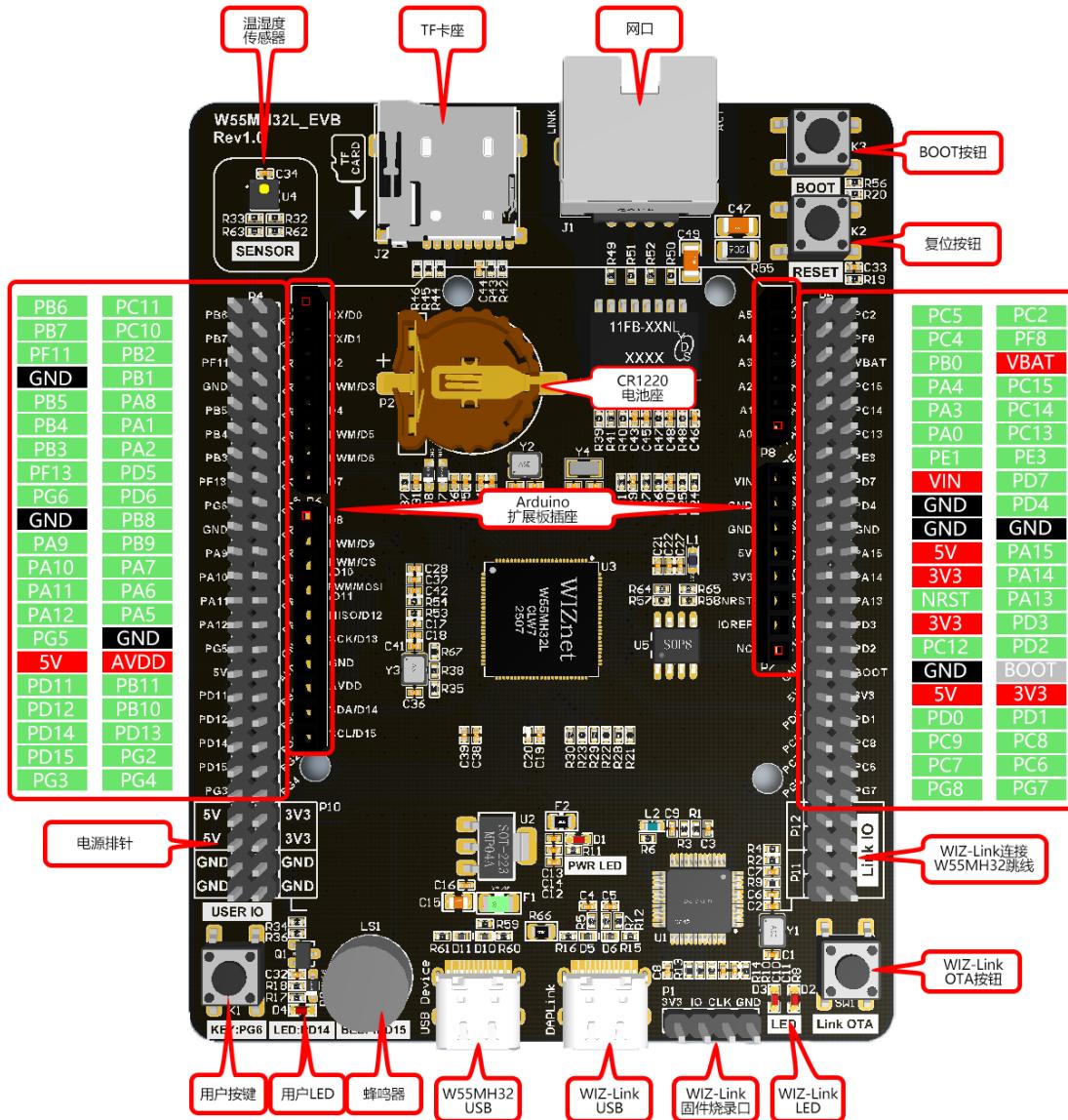


图 3 W55MH32L-EVB 硬件资源配置图

表 1 W55MH32L-EVB 硬件资源配置表

资源	描述
MCU	W55MH32L, QFN100, 1MB FLASH, 96KB RAM
主频	216MHz
GPIO	66 个

串口	5 路
SPI	2 路
I2C	2 路
USB	1 路
CAN	1 路
SDIO	1 路
Ethernet	1 路
12 位 ADC	3 路 (12 通道)
12 位 DAC	2 路 (2 通道)
随机数模块	支持
硬件加密算法单元	支持
RTC	CR1220 电池座*1
蜂鸣器	有源蜂鸣器*1
按键	复位按键*1, BOOT 按键*1, 用户按键*1, WIZ-Link OTA 按键*1
LED	PWR*1, USER*1, WIZ-Link*2
仿真器	WIZ-Link*1
温湿度传感器	AHT20*1
EEPROM	AT24C16C-SSHM-T*1
ARDUINO 插排	支持
USB	W55MH32L*1, WIZ-Link*1
以太网接口	下沉式 RJ45*1

1.2 W55MH32Q-EVB

1.2.1 简介

W55MH32Q-EVB 是基于 W55MH32Q 芯片开发的一款最小开发板，主频为 216MHz，1MB 的闪存以及 96KB 的 SRAM，同时还具有一个完整的硬件 TCP/IP 卸载引擎，只需要简单的 socket 编程即可实现以太网应用。

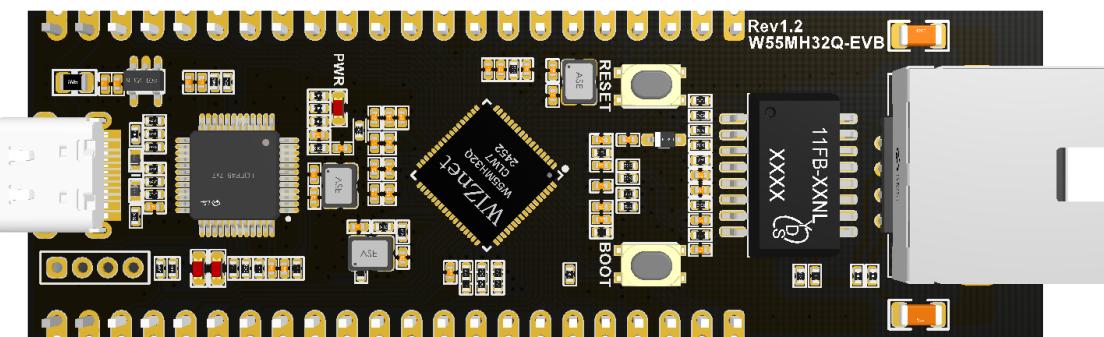


图 4 W55MH32Q-EVB 开发板正面



图 5 W55MH32Q-EVB 开发板背面

1.2.2 板载资源

芯片的所有 IO 都引出到开发板排针，芯片功能可以满足绝大多数的嵌入式应用场景和开发需求，尺寸小巧，方便测试芯片和评估功能。同时还集成了一个 WIZ-Link 仿真器，支持调试，程序下载以及虚拟串口功能。再也无须外接仿真器和串口调试工具，只需一块开发板即可轻松搞定单片机+以太网部分的学习内容。

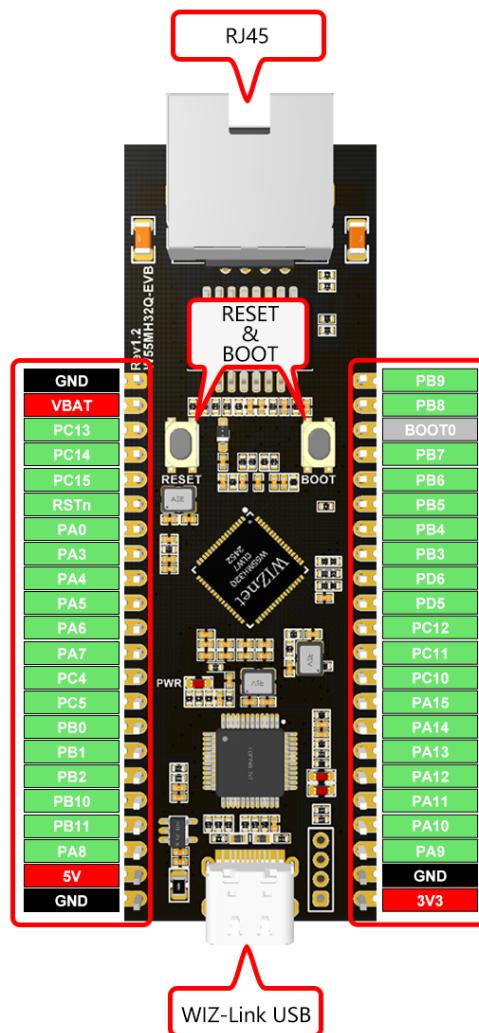


图 6 W55MH32Q-EVB 硬件资源配置图

表 2 硬件资源配置表

资源	描述
MCU	W55MH32Q, QFN68, 1MB FLASH, 96KB RAM
主频	216MHz
GPIO	36 个
串口	3 路
SPI	2 路
I2C	2 路
USB	1 路
CAN	1 路
Ethernet	1 路
12 位 ADC	3 路 (12 通道)
12 位 DAC	2 路 (2 通道)
随机数模块	支持
硬件加密算法单元	支持
按键	复位按键*1, BOOT 按键*1
LED	PWR*1
USB	W55MH32Q*1
以太网接口	下沉式 RJ45*1

2 安装 IDE

2.1 温馨提示

1. 安装路径不能带中文，必须是英文路径。
2. 安装目录不能跟 51 的 KEIL 或者 KEIL4 冲突，三者目录必须分开。
3. KEIL5 的安装比起 KEIL4 多了一个步骤，必须添加 MCU 库，不然没法使用。

2.2 获取 KEIL5 安装包

W55MH32 开发板支持 KEIL5 MDK 微控制器开发套件开发，可前往官网 www.keil.com 下载安装包。



图 7 Keil 官网安装包示意图

2.3 开始安装 KEIL5

双击官网下载的安装包，点击 Next 按钮开始安装。

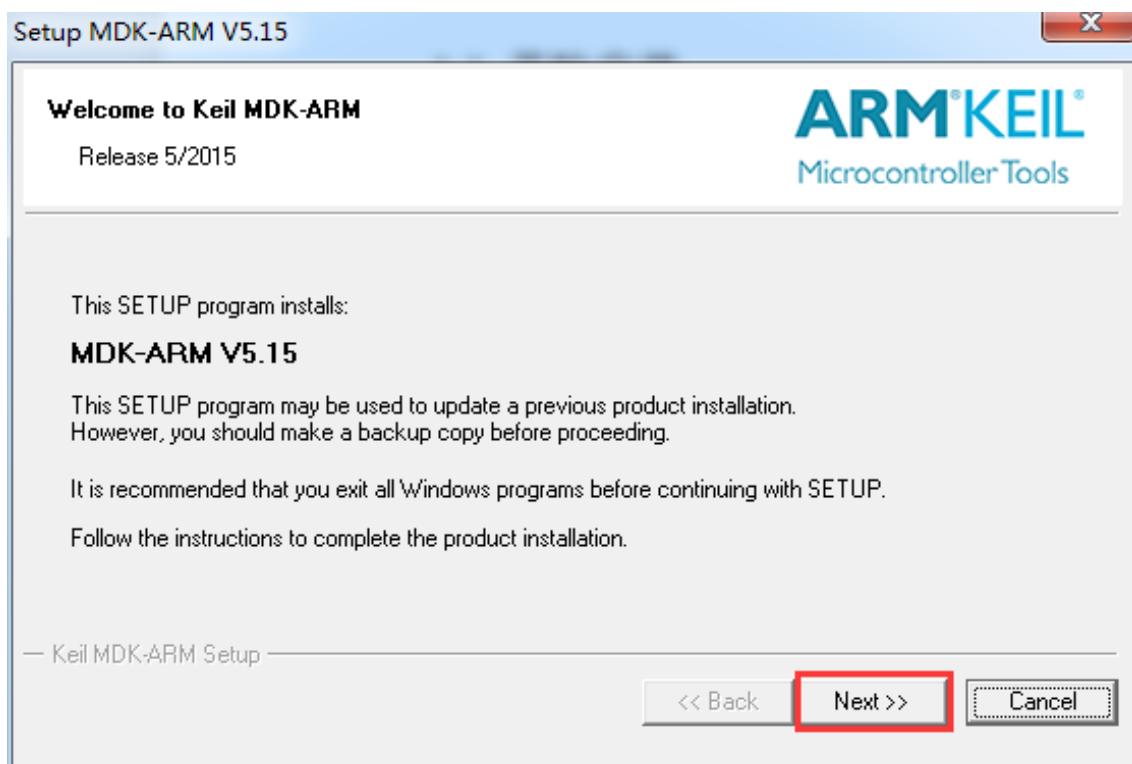


图 8 开始安装 KEIL5

按照提示逐步进行安装，安装完成后如下图所示。

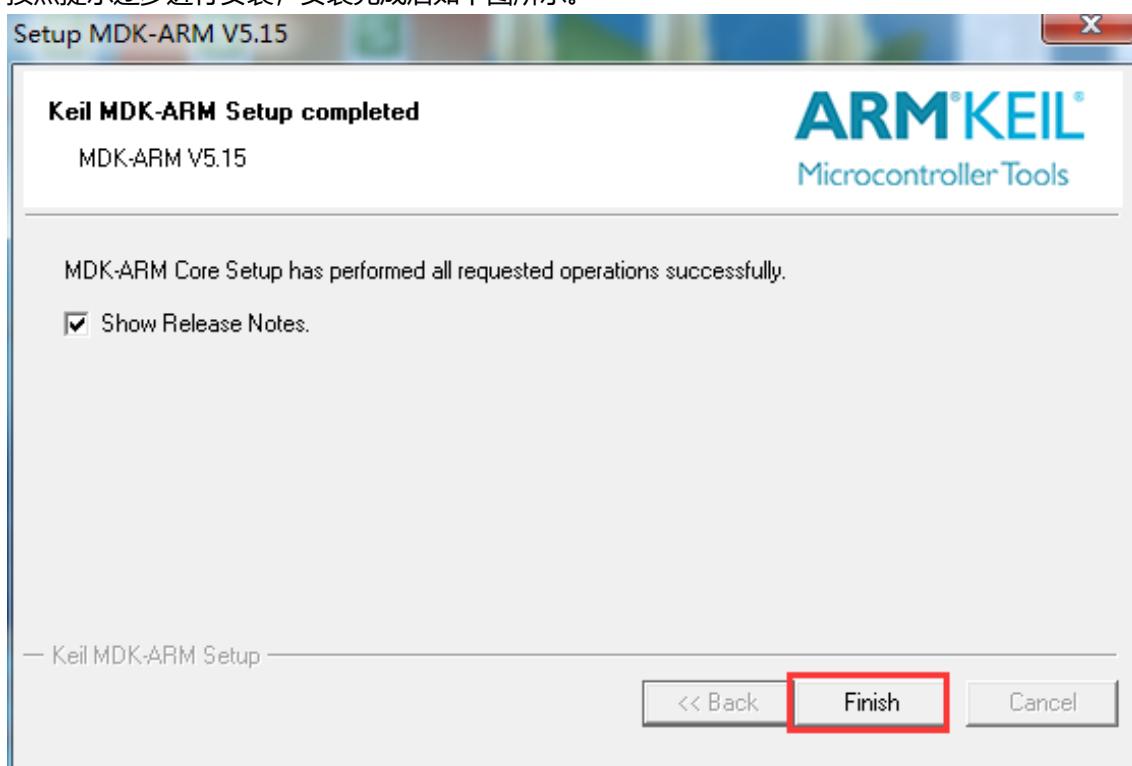


图 9 安装完成 KEIL5

2.4 安装 W55MH32 芯片包

打开软件包目录下的 W55MH32_DFP.1.1.0.pack，点击 Next 即可完成安装。

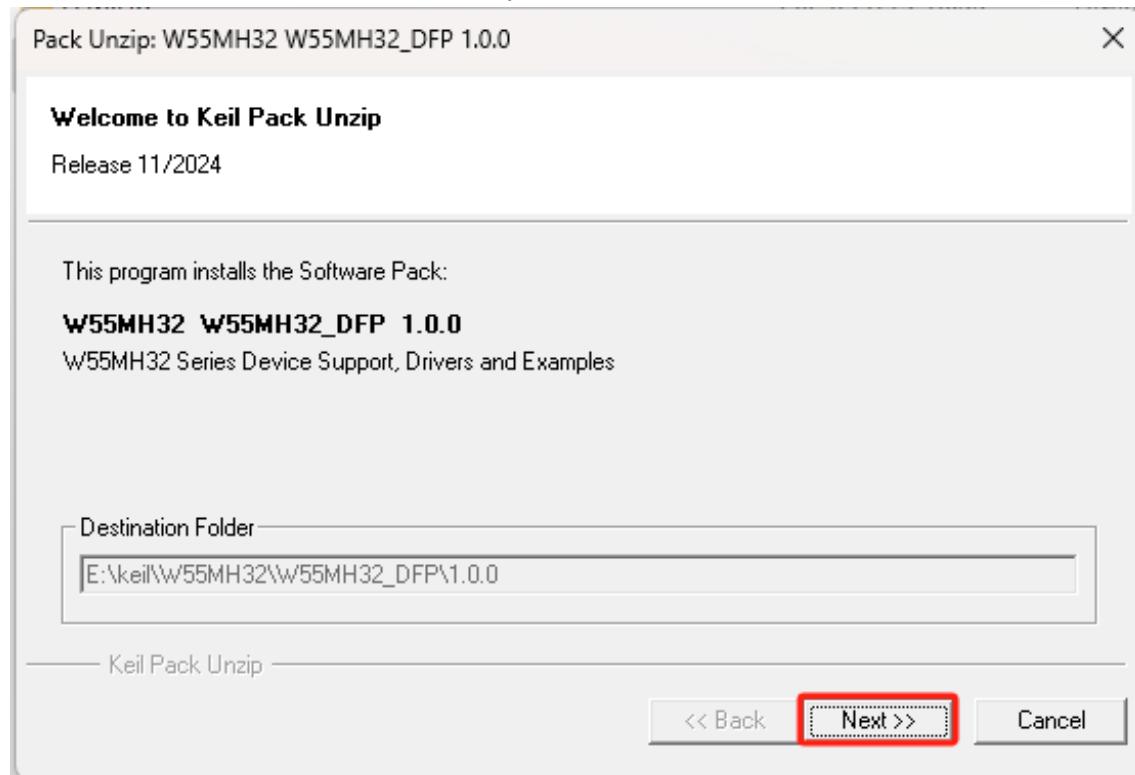


图 10 安装 pack 包

3 打开示例工程

3.1 外设示例目录

打开资料包中的 SDK\ModuleDemo 文件夹，里面包含了大部分外设示例以及 TOE 引擎示例。

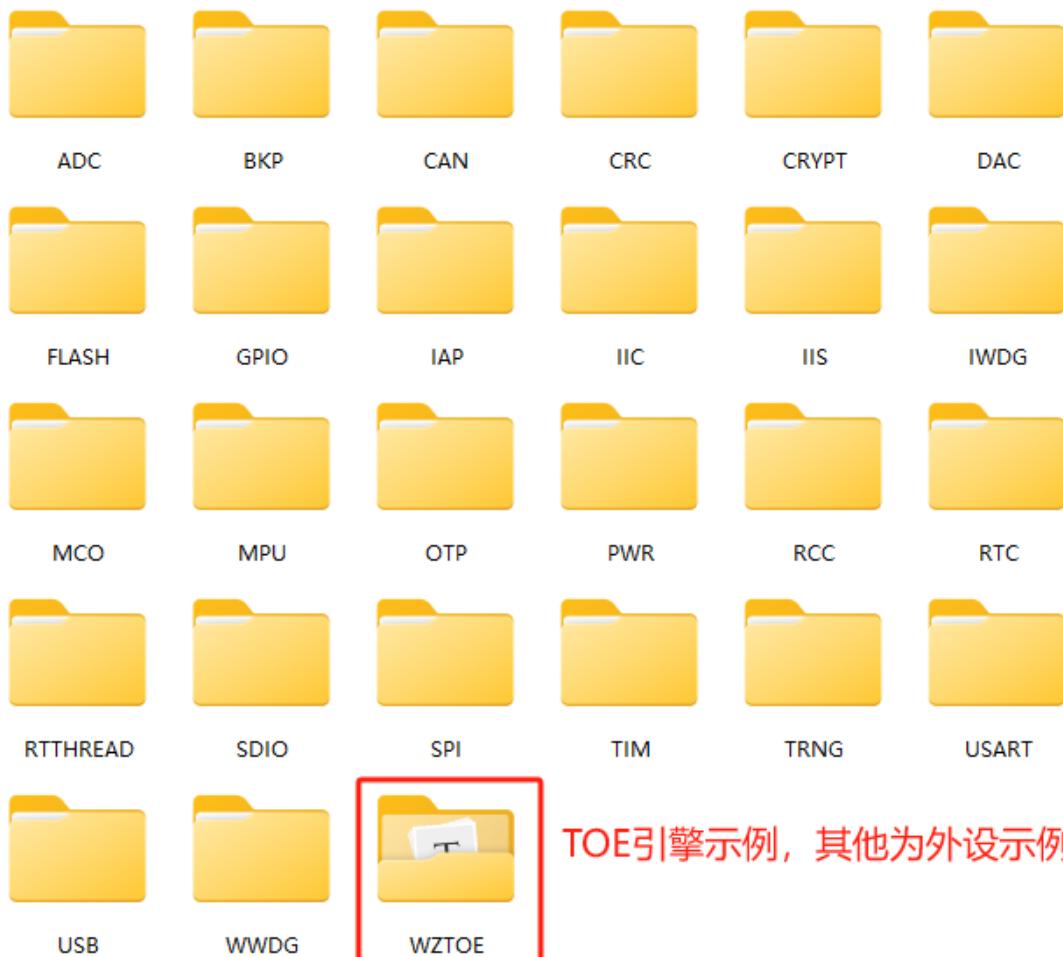


图 11 示例程序

3.2 TOE 引擎示例目录

其中 TOE 引擎示例包含多种网络协议应用。

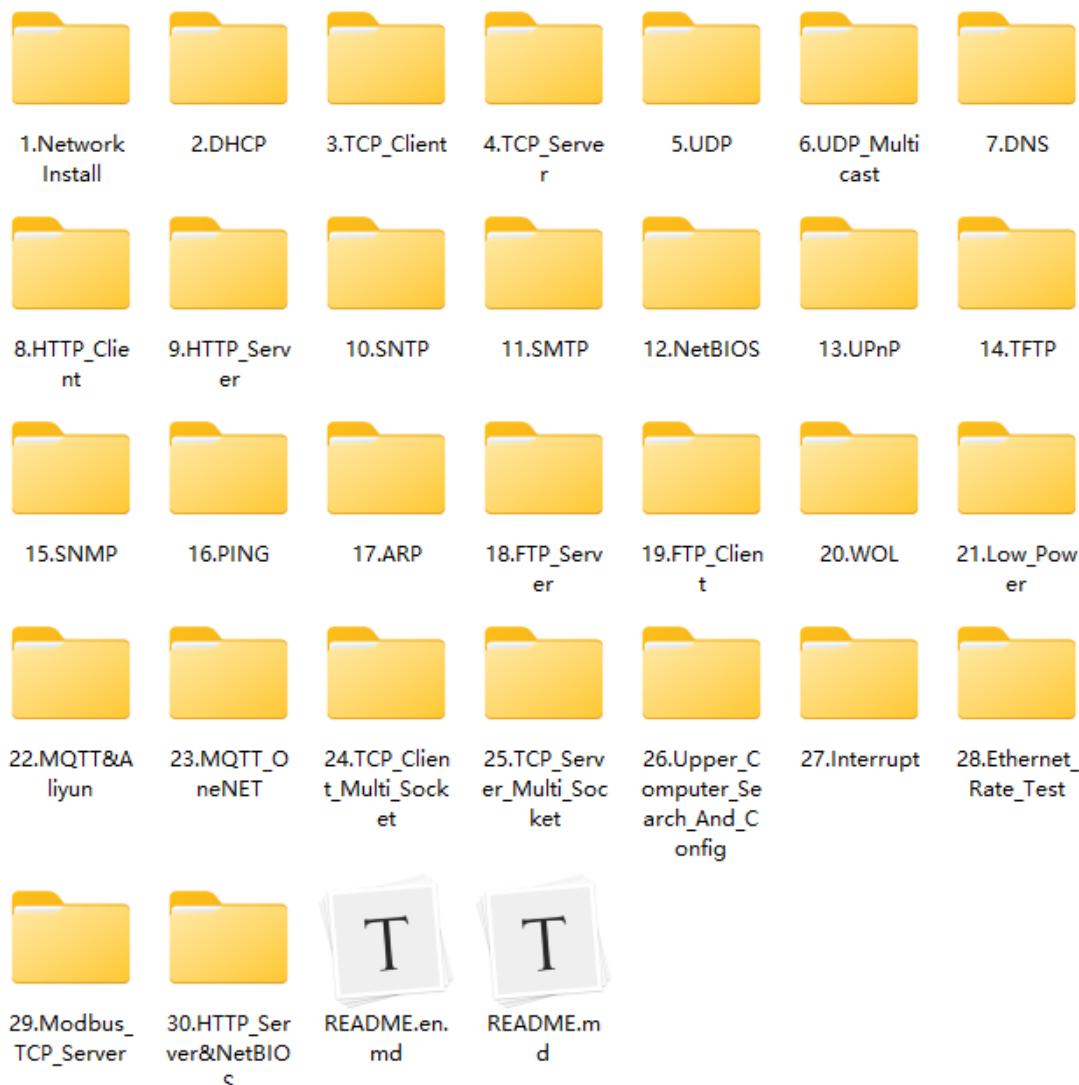


图 12 TOE 示例程序

3.3 KEIL 主界面介绍

KEIL 工程后缀为.uvprojx，外设例程的 KEIL 工程文件在对应外设示例的 USER 目录中，TOE 示例的 KEIL 工程文件在 Project 目录下。双击打开后出现 KEIL5 的主窗口界面，如下图所示。

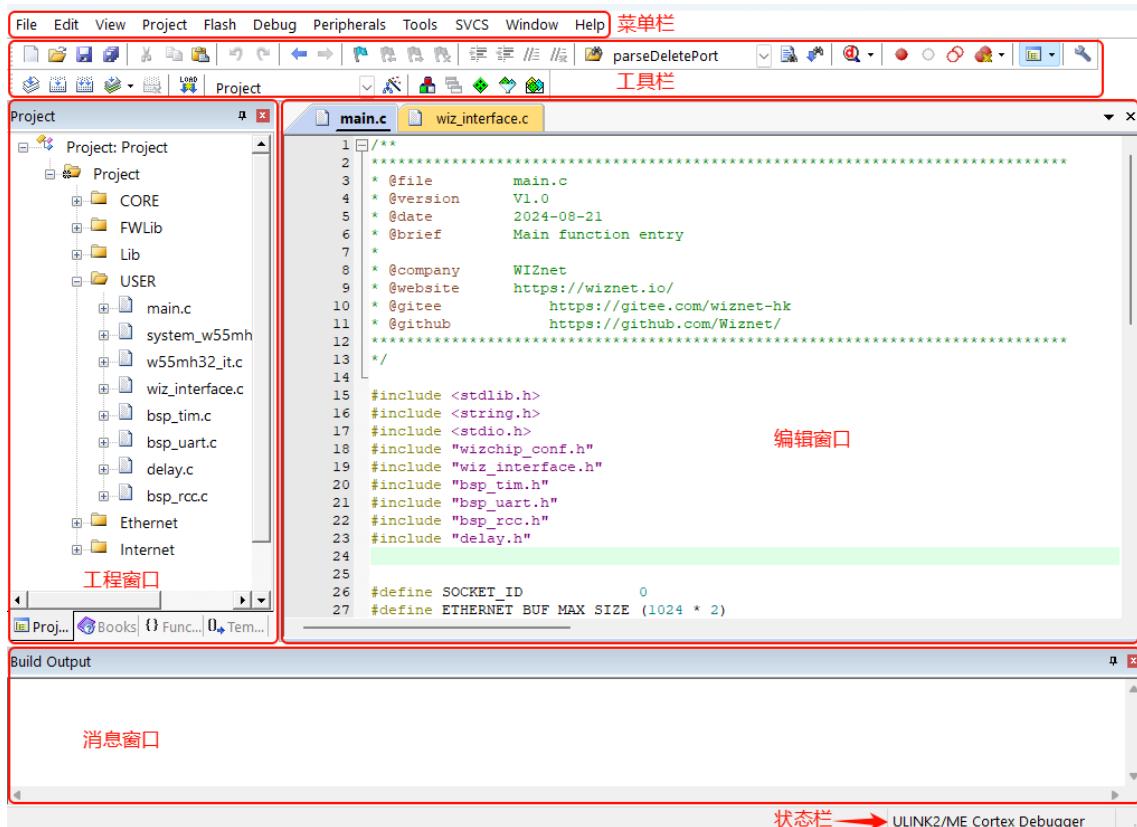


图 13 KEIL5 主界面

- 菜单栏**: 包含 File 文件、Edit 编辑、View 视图、Project 工程、Help 帮助等。
- 工具栏**: 常见工具的快捷按钮，下面会重点介绍一些常用的工具。
- 工程窗口**: 主要显示项目内容，文件组、源文件和头文件等。
- 编辑窗口**: 编写代码的地方。
- 消息窗口**: 反馈编译信息、烧录信息等。
- 状态栏**: 光标的行列位置、字符编码、键盘 NUM 锁定等一些状态信息。

3.4 常用快捷键按钮

三类常用工具快捷键按钮。



图 14 常用工具快捷键按钮

- 仿真类快捷按钮**: 在仿真时使用，具有断电标记作用。

表 3 仿真类按钮

英文名称	中文描述	快捷键
Start/Stop Debug Session	打开/关闭调试	Ctrl+F5
Insert/Remove Breakpoint	插入/移除断点	F9
Enable/Disable Breakpoint	使能/失能断点	Ctrl+F9

Disable All Breakpoints	失能所有断点	-
Kill All Breakpoints	取消所有断点	Ctrl+Shift+F9

2. **编译类快捷按钮**: 对代码进行编译下载。

表 4 编译类按钮

英文名称	中文描述	快捷键
Translate	编译当前文件	Ctrl+F7
Build	编译工程目标	F7
Rebuild	重新编译所有目标文件	-
Batch Build	分批编译(多工程)	-
Stop Build	停止编译(正在编译时有效)	-
Download	下载	F8

3. **工程目标选项又称魔术棒**: 即对工程目标的配置, 如芯片设备选择、C/C++选项、仿真配置等等。

4 仿真器介绍

4.1 仿真器简介

开发板集成了一个仿真器 WIZ-Link，如图 12 所示。WIZ-Link 遵循 ARM 公司的 CMSIS-DAP 标准，支持下载和在线仿真程序，支持在 XP/WIN7/WIN8/WIN10 这四个操作系统上免驱使用，支持 KEIL 和 IAR 直接下载。

此外，WIZ-Link 还支持一个虚拟串口，内部连接至 W55MH32 的 PA9 和 PA10。

下载，日志，仿真皆可通过 WIZ-Link 实现，非常方便。

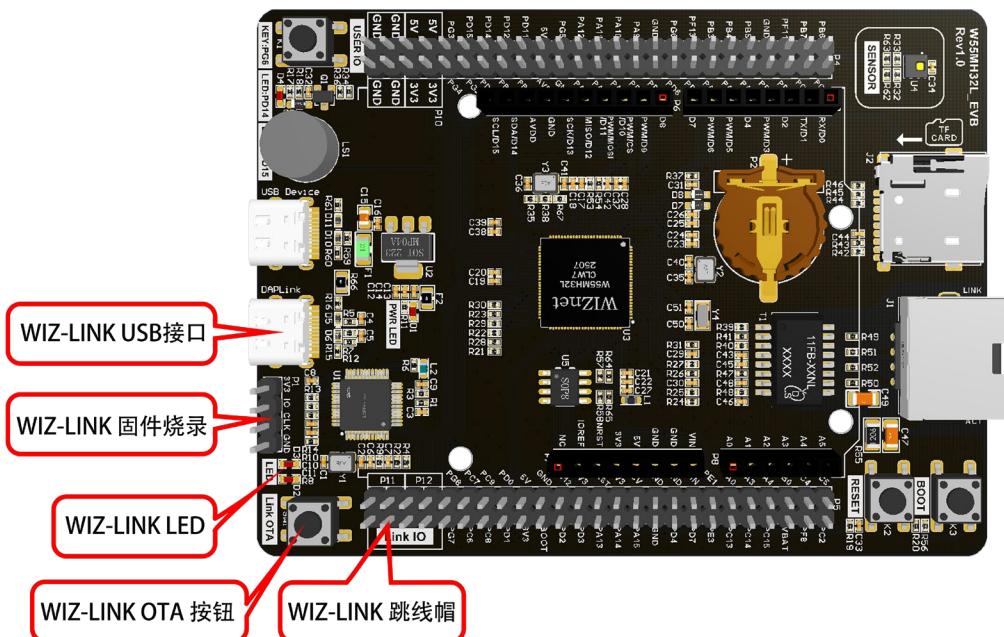


图 15 WIZ-Link

4.2 仿真器配置

在通过 WIZ-Link 连接好开发板和电脑后，打开电脑的设备管理器，可以看到 WIZ-Link 及虚拟的串口。如下图所示。

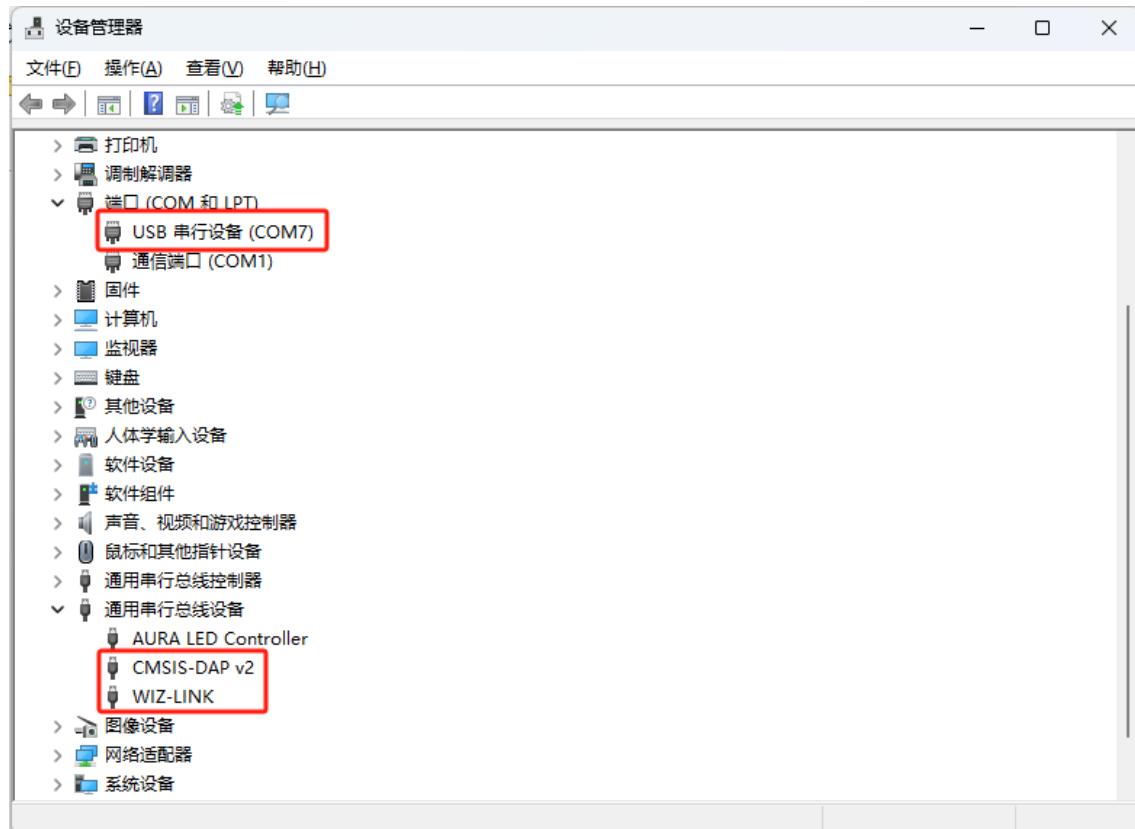


图 16 WIZ-Link 设备管理器示意图

打开 KEIL，在魔术棒选项卡里面选择 CMSIS-DAP 仿真器即可，如下图所示。

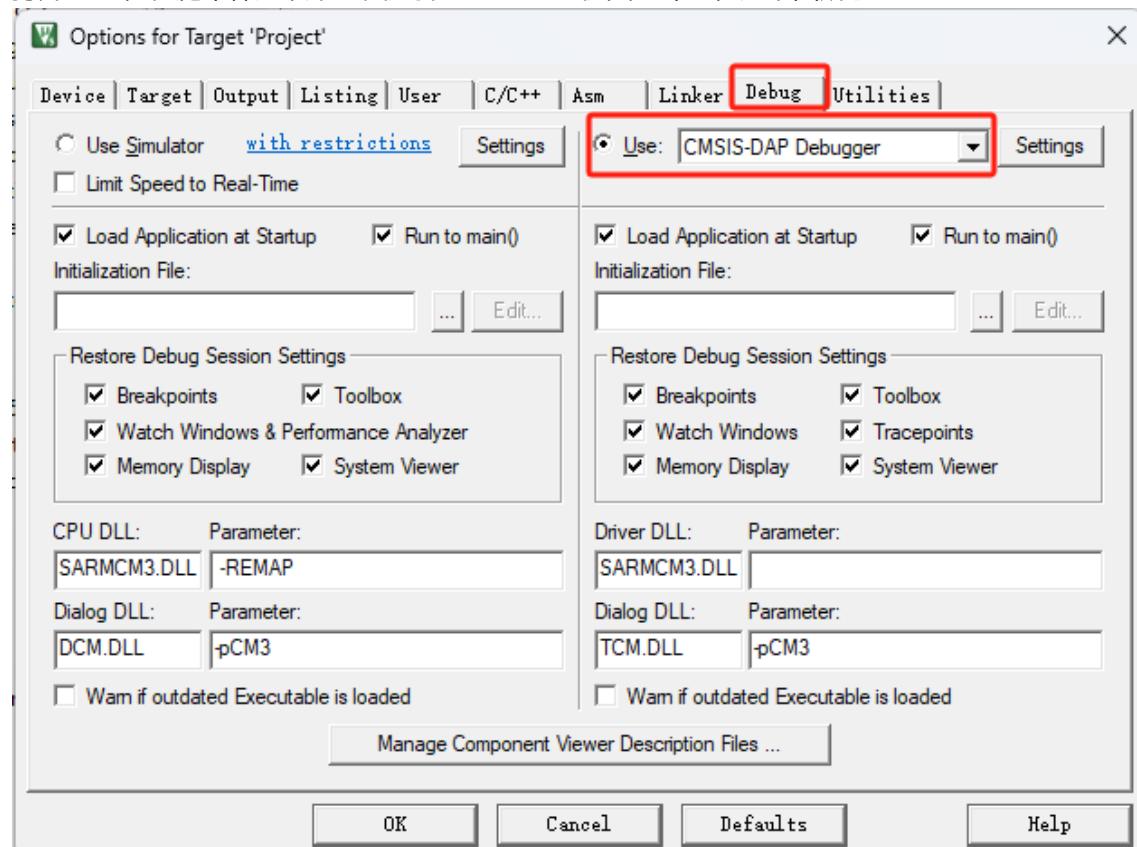


图 17 Debug 选项配置

点击 Settings 按钮后，进入 CMSIS-DAP Debugger 配置界面。

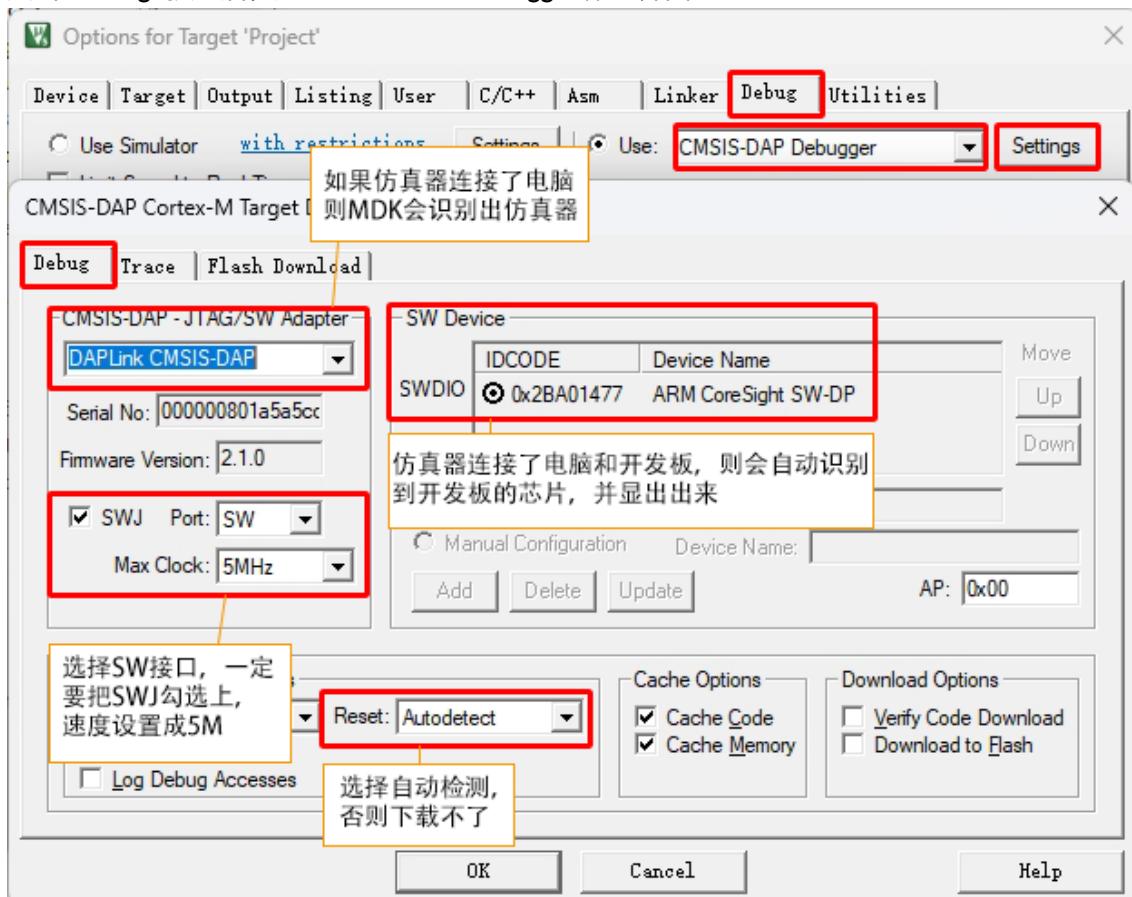


图 18 CMSIS-DAP Debugger 配置

注意：如果设备管理器识别到了 WIZ-Link，但是在 CMSIS-DAP Debugger 设置界面没有识别到开发板，可以将 KEIL5 升级到最新版本。

4.3 使用其他仿真器和串口调试工具

W55MH32 支持市面上大部分仿真器，如 ST-LINKV2、JLINK、ARM Emulator 等。

注意：不支持 ST-LINKV3。

开发板上使用跳线帽连接 W55MH32 和 WIZ-Link 的串口以及调试口，如图 16 所示，如您需要使用其他串口调试工具或仿真器，可以拔掉跳线帽以避免同时连接。

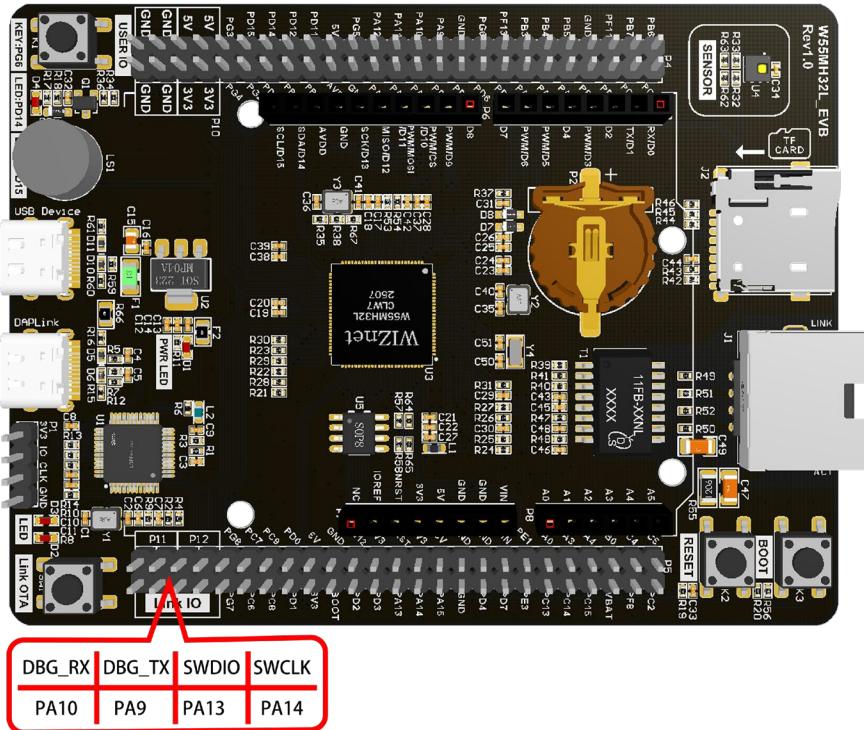


图 19 W55MH32 跳线帽连接 WIZ-Link

在使用 JLINK 时，在 JLINK 配置界面会出现无法识别芯片的情况，此时，需要做出如下改动。

1. 修改 JLINK 安装目录下的 JLinkDevices.xml 文件，添加如下内容。

```
<Device>
    <DeviceInfo Vendor="WIZnet" Name="W55MH32-100LQFP" Core="JLINK_CORE_CORTEX_M3"
    WorkRAMAddr="0x20000000" WorkRAMSize="0x00018000"/>
    <FlashBankInfo Loader="Devices/WIZnet/W55MH32/W55MH32.FLM"
    LoaderType="FLASH_ALGO_TYPE_OPEN" MaxSize="0x00100000" Name="Internal Flash"
    BaseAddr="0x08000000" AlwaysPresent="1"/>
</Device>
```

```
D:\software>SEGGER>JLink_V796g> JLinkDevices.xml ... 
1   <DataBase>
2     <Device>
3       <DeviceInfo WorkRAMAddr="0x20000000" JLinkScriptFile="Devices/ArteryTek/AT32FLM/AT32F4xx.JLinkScript" Vendor="ArteryTek" Name="AT32A423_APP0E_256" Core="JLINK_CORE_CORTEX_M4" WorkRAMSize="0x00C000"/>
4       <flashBankInfo Loader="Devices/ArteryTek/AT32A423_256.FLM" LoaderType="FLASH_ALGO_TYPE_OPEN" MaxSize="0x00040000" Name="Flash Bank1" BaseAddr="0x8000000" AlwaysPresent="1"/>
5       <flashBankInfo Loader="Devices/ArteryTek/AT32FLM/AT32A423_BOOTMEM_AP.FLM" LoaderType="FLASH_ALGO_TYPE_OPEN" MaxSize="0x0005000" Name="Flash Boot Memory" BaseAddr="0x1FFFA400" AlwaysPresent="1"/>
6     </Device>
7     <Device>
8       <DeviceInfo WorkRAMAddr="0x20000000" JLinkScriptFile="Devices/ArteryTek/AT32FLM/AT32F4xx.JLinkScript" Vendor="ArteryTek" Name="AT32M23U5D" Core="JLINK_CORE_CORTEX_M4" WorkRAMSize="0x00C000"/>
9       <flashBankInfo Loader="Devices/ArteryTek/AT32FLM/AT32A423_USD.FLM" LoaderType="FLASH_ALGO_TYPE_OPEN" MaxSize="0x200" Name="Flash USD" BaseAddr="0x1FFF800" AlwaysPresent="1"/>
10    </Device>
11    <Device>
12      <DeviceInfo Vendor="W55MH32" Name="W55MH32_W2153AVG16" Core="JLINK_CORE_CORTEX_M3" WorkRAMAddr="0x20000000" WorkRAMSize="0x00018000"/>
13      <flashBankInfo Loader="Devices/W55MH32/W2153AVG16.FLM" LoaderType="FLASH_ALGO_TYPE_OPEN" MaxSize="0x00100000" Name="Internal Flash" BaseAddr="0x08000000" AlwaysPresent="1"/>
14    </Device>
15    <Device>
16      <DeviceInfo Vendor="WIZnet" Name="W55MH32-100LQFP" Core="JLINK_CORE_CORTEX_M3" WorkRAMAddr="0x20000000" WorkRAMSize="0x00018000"/>
17      <flashBankInfo Loader="Devices/WIZnet/W55MH32/W55MH32.FLM" LoaderType="FLASH_ALGO_TYPE_OPEN" MaxSize="0x000100000" Name="Internal Flash" BaseAddr="0x08000000" AlwaysPresent="1"/>
18    </Device>
19  </DataBase>
```

图 20 JLINK 修改内容示意图

2. 在 Devices/WIZnet/W55MH32/路径下，添加 W55MH32.FLM 文件。

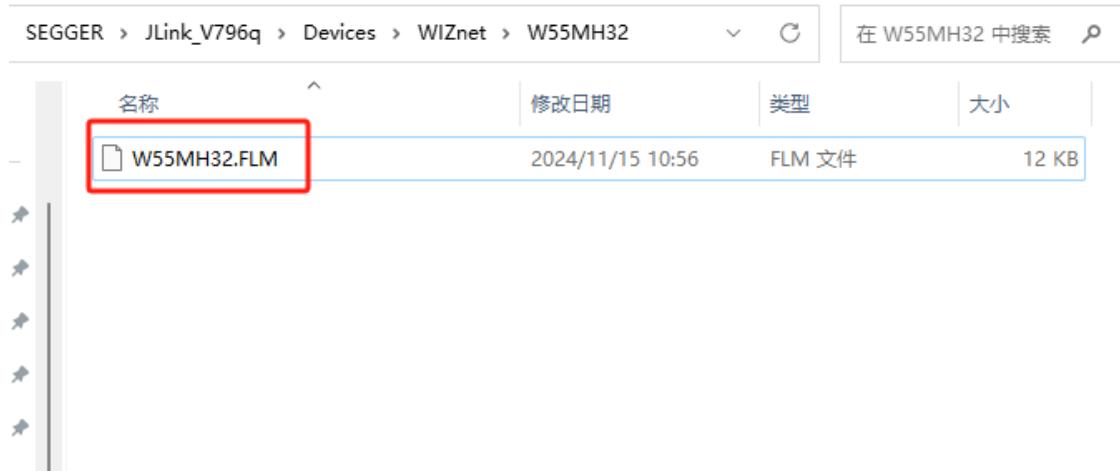


图 21 添加 W55MH32.FLM 文件

其中，W55MH32.FLM 文件在 keil 安装目录下的 W55MH32\W55MH32_DFP\1.0.0\CMSIS\Flash 路径中，如下图所示。



图 22 W55MH32.FLM 文件位置