

OPL1000

ULTRA-LOW POWER 2.4GHZ WI-FI + BLUETOOTH SMART SOC

DEVKIT 电路板入门使用



OPULINKS

<http://www.opulinks.com/>

Copyright © 2017-2020, Opulinks. All Rights Reserved.

OPL1000-DEVKIT 电路板入门使用 | Version V2.0

版本记录

日期	版本	更新内容
2018-05-10	0.1	<ul style="list-style-type: none">初版
2018-05-17	0.2	<ul style="list-style-type: none">更新章节 3.1.2, 3.2.2 因为下载工具适用更法已更新.加入 Figure10 介绍如何验证 AT UART 执行更新 Figure 12 内容加入章节 3.4 介绍如何启动 APS 串口打印 log 信息章节 3.3 拆成 3.3.1 和 3.3.2.加入章节 3.3.2 介绍 J-link 仿真器
2018-05-31	0.3	<ul style="list-style-type: none">Mini USB 串口将视为 APS UART 串口, 当再使用 v1.0.1.19 SDK 之后下载使用者 APP 时
2018-06-05	0.4	<ul style="list-style-type: none">加入新章节 3.1更新章节 3.2.2.
2018-07-13	0.5	<ul style="list-style-type: none">更新章节 3.2
2018-08-03	0.6	<ul style="list-style-type: none">更新章节 2 及 章节 3.5 符合 A2 芯片
2018-08-15	0.7	<ul style="list-style-type: none">更新章节 2, figure2更新章节 3.2 及 章节 3.3
2018-11-09	0.8	<ul style="list-style-type: none">更新章节 3.4.1 及 章节 3.4.2 启动 ICE 接口
2018-11-20	0.9	<ul style="list-style-type: none">更新章节 3.4.1 不作用 watchdog
2019-04-22	1.0	<ul style="list-style-type: none">加入 LM80 介绍
2019-06-21	1.1	<ul style="list-style-type: none">更新 AT 和 APS 串联接口描述.
2019-07-08	1.2	<ul style="list-style-type: none">更新 Fw_binary 目录到 FW_Pack 目录
2020-02-17	2.0	<ul style="list-style-type: none">同步到 A2 板子, 更新 IOTech DEVKIT 介绍加入 IO uart 和 APS 切换功能.

目录

1. 介绍 1

1.1. 文档应用范围 1

1.2. 缩略语 1

1.3. 参考文献 1

2. DEVKIT 概要介绍 2

2.1. DEVKIT 概要介绍 2

3. 使用 DEVKIT 4

3.1. 安装 USB 驱动 4

3.1.1. Win10/Win7 系统驱动安装 4

3.1.2. WinXP/Vista 系统驱动安装 8

3.2. APS 和 UART AT 设定与开发 10

3.2.1. 设定初始的状态值 10

3.2.2. 指令函式 11

3.2.3. 设定 IO 功能 11

3.3. APS 和 UART AT 串口转换 13

3.4. APS 串口连接和使用 14

3.4.1. APS 串口连接 14

3.4.2. 通过 APS 观察固件调试打印信息 14

3.5. AT 串口连接和使用 15

3.5.1. 通过 AT 串口更新固件 15

3.5.2. 使用 AT 串口执行 AT 指令 18

3.6. SWD 端口 18

3.6.1. 禁用 watchdog 18

3.6.2. M3 ICE 端口连接 19

3.7. 外接供电 20

3.8. DEVKIT 复位 21

图目录

Figure 1: DEVKIT 板组成介绍.....2

Figure 2: DEVKIT 扩展 IO map.....3

Figure 3: DEVKIT 设备端口4

Figure 4: Win10/7 驱动安装搜索.....5

Figure 5: Win10/Win7 驱动安装 – 选择从驱动程序列表中选取5

Figure 6: Win10/Win7 驱动安装 – 为 CP210x 更新驱动程序.....6

Figure 7: Win10/Win7 驱动安装 –指定安装目录.....6

Figure 8: Win10/Win7 驱动安装 – 选择 USB inf 文件7

Figure 9: Win10/Win7 驱动安装 – 驱动更新完成.....7

Figure 10: Win10/Win7 驱动安装 – 检查驱动版本信息8

Figure 11: WinXP/Vista 驱动安装 – 执行驱动安装程序9

Figure 12: WinXP/Vista 驱动安装 – 接受协议9

Figure 13: WinXP/Vista 驱动安装 – 检查驱动版本信息..... 10

Figure 14: 预设初始化 IO01 为 AT UART..... 10

Figure 15: 预设初始化 IO01 为 APS..... 11

Figure 16 : AT UART work mode..... 13

Figure 17: AT UART switch to Debug UART 13

Figure 18: APS 串口接线实例..... 14

Figure 19: APS 串口输出打印信息 15

Figure 20: DEVKIT AT 串口设备 15

Figure 21:载入 M3/M0 Bin 文件进行合并操作..... 16

Figure 22: Patch Bin 文件下载..... 17

Figure 23: 启动后 APS 串口输出 log 信息..... 17

Figure 24: 执行 AT 指令 18

Figure 25: 禁用 watchdog..... 19

Figure 26: DEVKIT 板上 M3 ICE 连线图..... 20

Figure 27: J-link ICE 仿真器正确识别 20

Figure 28: J-link ICE 仿真器正确识别 21

表目录

Table 1: AT/Debug UART Switch Command 13

Table 2: DEVKIT M3 SWD 信号连接..... 19

1. 介绍

1.1. 文档应用范围

OPL1000 DEVKIT 用于评估 OPL1000 芯片的功能、开发应用程序。本文介绍了 DEVKIT 板子的组成，如何使用 DEVKIT 提供的端口进行固件下载和应用程序调试。DEVKIT 采用 OPL1000 低功耗芯片，均由开发母版和子模块组成，而子板作为最小系统，可以单独使用，母板保证了电压的稳定，以及 USB 转串口功能。

1.2. 缩略语

Abbr.	Explanation
APP	APPLication 应用程序
APS	Application Sub-system 应用子系统，在本文中亦指 M3 MCU
AT	Attention 终端命令指令集
DevKit	Development Kit OPL1000 评估开发板
EVB	Evaluation Board 评估板
FW	FirmWare 固件，处理器上运行的嵌入式软件
ICE	In-Circuit Emulator 在线仿真调试工具
RX	Receive 接收
SWD	Serial Wire Debug 串行线调试
TX	Transmit 发送

1.3. 参考文献

[1] Download tool 使用指南 OPL1000-patch-download-tool-user-guide.pdf

[2] HDK 开发文档 https://github.com/Opulinks-Tech/OPL1000-HDK/blob/master/OPL1000-HDK%20Development%20Guide_UG1-01-1.pdf

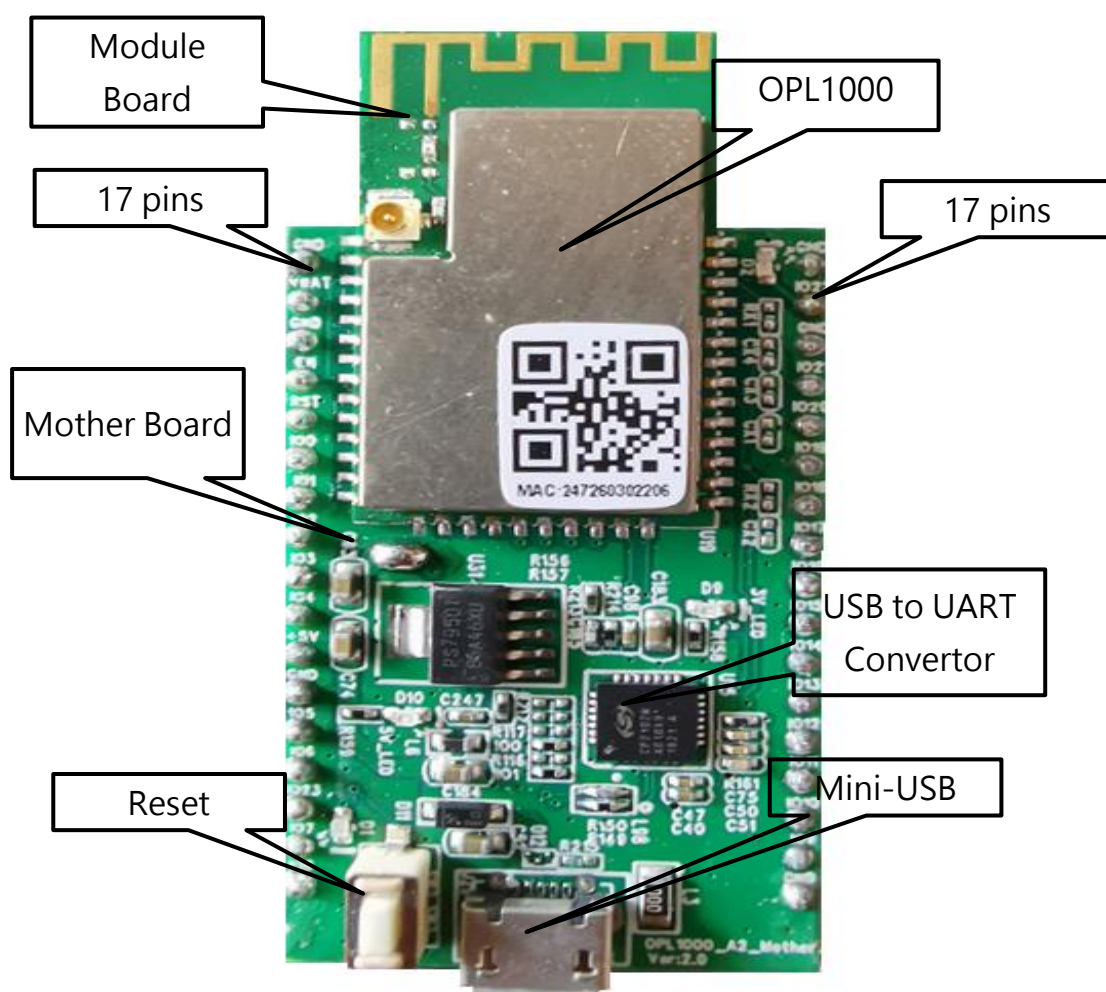
2. DEVKIT 概要介绍

2.1. DEVKIT 概要介绍

DEVKIT 板包括一个开发母板和 OPL1000 模块子板。母板包括 USB 转 UART 转换芯片、Flash 芯片和电源适配模块。OPL1000 模块子板包括 OPL1000 SOC 芯片和外部晶振。如

Figure 1 所示：

Figure 1: DEVKIT 板组成介绍



Opulinks DEVKIT 提供 mini USB 转串口功能，mini USB 同时提供供电功能。用户可以轻松使用 USB 控制 OPL1000，快速进行功能评估以及完成产品开发。DEVKIT 母板提供了若干扩充 GPIO 管脚，在线开发用的 ICE mode 管脚及 flash 烧录用的 UART Tx 及 Rx 管脚。OPL1000 提供 flash 烧录软件。扩展 GPIO 管脚可配置为 GPIO、ADC、SPI、I2C 等功能。扩充排针 J2,J3 配置底视图 (Bottom view)，如 Figure 2 所示。

Figure 2: DEVKIT 扩展 IO map

J2						ANT	J3					
ICE Mode	PWM	I2C	ADC	Pin Name	Pin No		Pin No	Pin Name	ADC	SPI	UART	Flash Prg
				GND	pin 17	Bottom View	pin 17	GND				
	Yes			GPIO22	pin 16		pin 16	+3V				
				GND	pin 15		pin 15	GND				
M3_CLK				GPIO21	pin 14		pin 14	CHIP_EN				
M3_DAT				GPIO20	pin 13		pin 13	RST_N				
M0_DAT				GPIO19	pin 12		pin 12	GPIO0(REV)				UART_Prg_Tx
M0_CLK				GPIO18	pin 11		pin 11	GPIO1(REV)				UART_Prg_Rx
		SDA	Yes	GPIO17	pin 10		pin 10	GPIO2	Yes	MOSI	TxD	
		SCLK	Yes	GPIO16	pin 9		pin 9	GPIO3	Yes	MISO	RxD	
				GPIO15	pin 8		pin 8	GPIO4	Yes	CLK		
				GPIO14	pin 7		pin 7	Ex_5V				
				GPIO13	pin 6		pin 6	GND				
	Yes			GPIO12	pin 5		pin 5	GPIO5	Yes	CS		
				GPIO11	pin 4		pin 4	GPIO6	Yes			
				GPIO10	pin 3		pin 3	GPIO23				
	Yes			GPIO9	pin 2		pin 2	GPIO7	Yes	CS		
				GND	pin 1	USB	pin 1	GPIO8	Yes			

注 1：UART_Prg 串口波特率为 115200 bps。

注 2：chip Enable (CHIP_EN)和 Reset (RST_N)都可以视为 Reset 功能。

3. 使用 DEVKIT

OPL1000 DEVKIT 板提供了三个通信端口用于用户程序开发。它们分别是：

AT 串口: AT 串口用于升级固件以及发送 AT 命令给 OPL1000。在目前 DEVKIT 板使用的 USB 转串口芯片为 Silicon Labs CP210x 芯片，用户可以通过查找设备管理器“COM 和 LPT 端口”得知 mini-USB 连接的串口编号是多少。例如下图所示的设备端口列表中，OPL1000 DEVKIT AT 串口为 COM13，对应的 IO pin 是 IO01。

Figure 3: DEVKIT 设备端口



1. APS 串口：该串口用于输出内部调试信息，对应的 IO pin 是 IO8/9;
2. Cortex M3 SWD 调试接口。

注：AT UART 以及 APS debug 可以相互转换定义，但是无论如何转换串口，固件的下载都要经过 IO01 口来完成。

3.1. 安装 USB 驱动

OPL1000 DEVKIT 使用 Silicon Labs CP210x USB 转 UART 桥接芯片。为使 Mini-USB 正常工作，需要根据用户使用的操作系统类型安装对应的驱动软件。驱动软件目录为：Tool\CP210x_Windows_Drivers

3.1.1. Win10/Win7 系统驱动安装

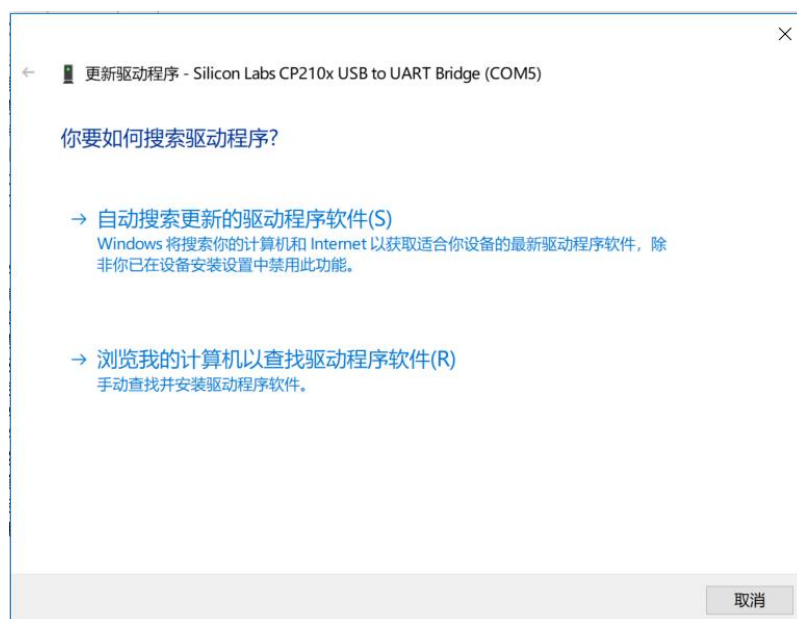
CP210x 桥接芯片的 Win10/Win7 系统驱动程序路径

Tool\CP210x_Windows_Drivers\Win7_Win10_x64

Win10 操作系统会自动安装 CP210x 桥接芯片驱动。但是这个驱动和 download 工具使用的串口模块库有冲突，需要进行版本降级。即使用 Win7_Win10_x64 目录下的驱动程序。操作步骤为：

Step1 首先打开设备管理器，选择串口设备“Silicon Labs CP210x USB to UART Bridge”，右键选择‘更新驱动程序’，点击‘浏览我的计算机以查找驱动程序软件’

Figure 4: Win10/7 驱动安装搜索



Step2 点击‘让我从计算机上的可用驱动程序列表中选择’，如下图所示。

Figure 5: Win10/Win7 驱动安装 – 选择从驱动程序列表中选择



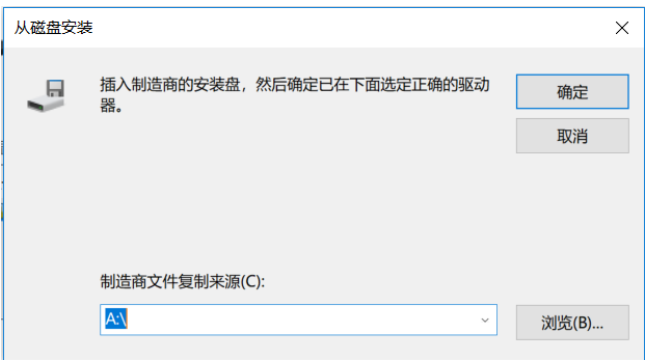
Step3 由于 CP210x 桥接芯片已经在系统中有注册，因此在 Figure 6 “显示兼容硬件” 列表中有 Silicon Labs CP210x USB to UART Bridge 。点击 ‘从磁盘安装’

Figure 6: Win10/Win7 驱动安装 – 为 CP210x 更新驱动程序



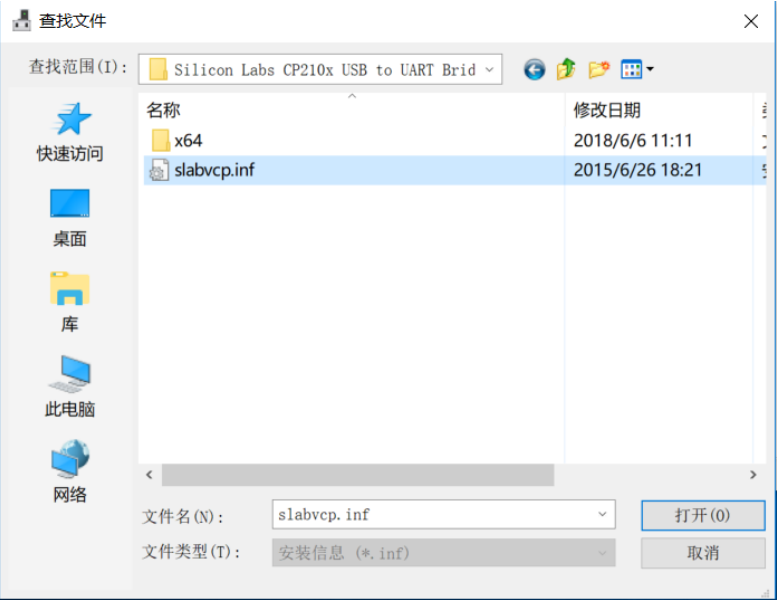
Step4 在弹出窗口中点击 ‘浏览’ 选择 Tool\CP210x_Windows_Drivers\Win7_Win10_x64 目录

Figure 7: Win10/Win7 驱动安装 –指定安装目录



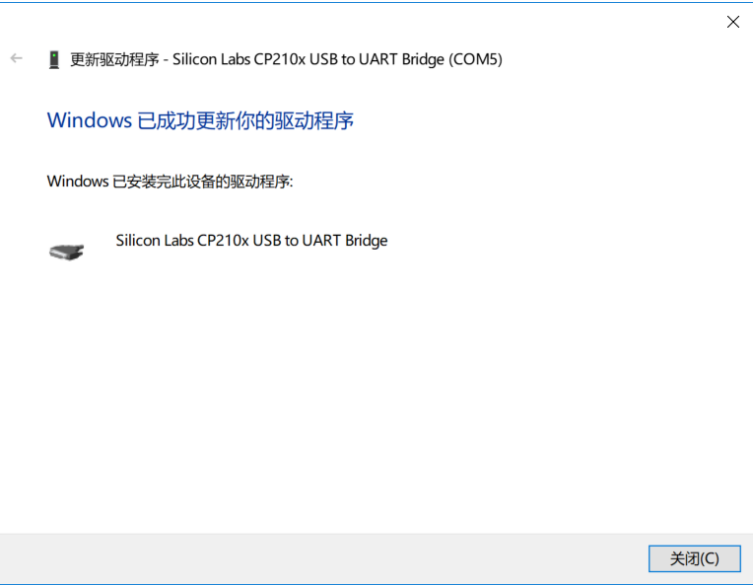
Step5 选择 Tool\CP210x_Windows_Drivers\Win7_Win10_x64 目录下的 slabcvp.inf 文件，点击 “打开” 。slabcvp.inf 包含了 CP210x 芯片的 USB 设备信息和驱动配置信息。

Figure 8: Win10/Win7 驱动安装 – 选择 USB inf 文件



Step6 点击“确定”按钮，“下一步”，按钮。点击“关闭”按钮，安装完成。得到 Figure 9 所示画面。表示驱动更新完成。

Figure 9: Win10/Win7 驱动安装 – 驱动更新完成



Step7 检查驱动版本信息，是否为 6.7.1.0。右键点击串口设备“Silicon Labs CP210x USB to UART Bridge”，在属性中选择“驱动程序”标签页。如果版本为 6.7.1.0 则表示驱动安装正确。

Figure 10: Win10/Win7 驱动安装 – 检查驱动版本信息



3.1.2. WinXP/Vista 系统驱动安装

CP210x 桥接芯片的 WinXP/Vista 系统驱动程序路径 Tool\CP210x_Windows_Drivers\WinXP_Vista 。

整个安装包括三个步骤：

Step1 运行 WinXP_Vista 目录下的 'CP210xVCPIInstaller_x86.exe' 程序

Step2 在弹出的对话框界面 Figure 11 上 点击 “下一步” 。

得到 Figure 12 所示画面。然后选择 “我接受这个协议” ， “下一步” 。

Figure 11: WinXP/Vista 驱动安装 – 执行驱动安装程序

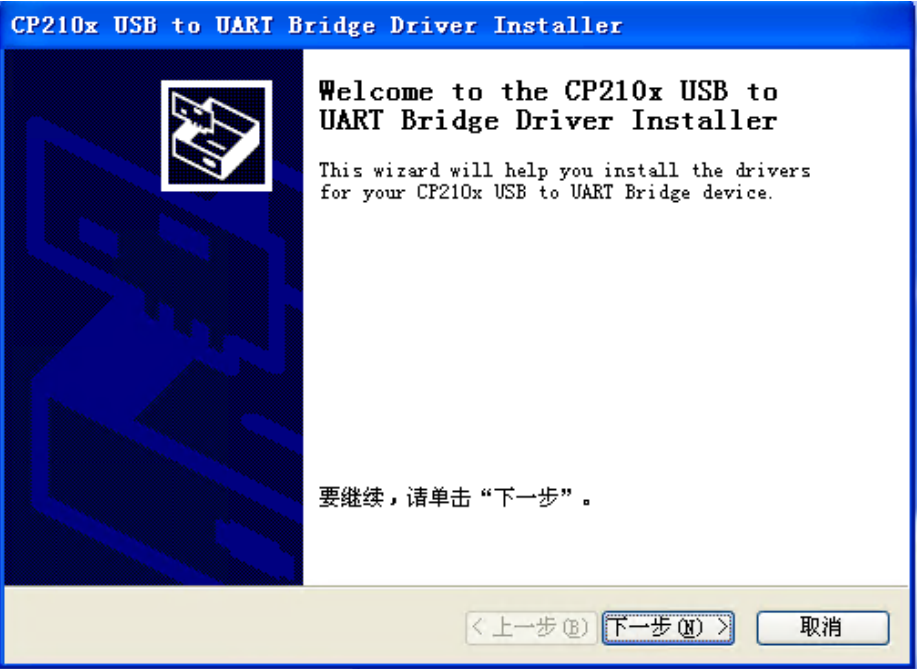


Figure 12: WinXP/Vista 驱动安装 – 接受协议



Step3 安装结束后点击“完成”

最后检查驱动版本信息。右键点击串口设备“Silicon Labs CP210x USB to UART Bridge”，在属性中选择“驱动程序”标签页。如果得到 Figure 13 所示 6.7.0.0 版本号就对了。

Figure 13: WinXP/Vista 驱动安装 – 检查驱动版本信息



3.2. APS 和 UART AT 设定与开发

3.2.1. 设定初始的状态值

初始化的定义在 hal_pin_config_project.h

预设初始化 IO0 和 IO1 为 AT UART 模式，如下：

Figure 14: 预设初始化 IO01 为 AT UART

```
/* Select IO 0/1 UART mode. For switching UART use */
#define HAL_PIN_0_1_UART_MODE      IO01_UART_MODE_AT      // IO01_UART_MODE_AT
                                                                    // IO01_UART_MODE_DBG
```


当使用者想要切换 IO0 和 IO1 为 Debug UART 时，必需修改如下：

Figure 15: 预设初始化 IO01 为 APS

```
/* Select IO 0/1 UART mode. For switching UART use */
#define HAL_PIN_0_1_UART_MODE      IO01_UART_MODE_DBG      // IO01_UART_MODE_AT

// IO01_UART_MODE_DBG
```

3.2.2. 指令函式

下列函式皆为控制交换 AT UART 和 Debug UART 的行为：

- `at_cmd_at_switch_to_dbg` 函式在 `at_cmd_table_ext.c` 里头。
- `ParseSwitchAT_DBGCommand_patch` 函式在 `cli_patch.c` 里头。

3.2.3. 设定 IO 功能

- 下表中的源码定义在 `at_cmd_switch_uart1_dbguart` 的函式之中，在 `at_cmd_common_patch.c` 里头。
- 当前转换范例的 IO 数字为 8、9，但用户可以依据用户的设计，设定成符合用户的 IO 脚位，不局限于 IO 脚位为 8、9。
 - UART 行为中，拉高电位表示不作动。Tx 会自主把 pin 脚位拉高或低，但 Rx 不会自主把 pin 脚位的电流拉高或拉低，并且外部也没有接 pull high 的原件。要让 Rx 的脚位在预设的情况下不作动，故在设定上必须设定为 `PIN_DRIVING_HIGH`。
 - 其它脚位设定为 float 的原因，当脚位设定为 HIGH 或 LOW 时，脚位必须提供较大的电流让脚位作动或不作动，这也是因为脚位设定成 `PIN_DRIVING_FLOAT` 的原因。

```
if (g_eIO01UartMode == UART_AT)
{
    /* AT UART 在 0、1 switch to Debug UART */
    Hal_Pin_ConfigSet(0, PIN_TYPE_UART_APS_TX, PIN_DRIVING_FLOAT);
    Hal_Pin_ConfigSet(1, PIN_TYPE_UART_APS_RX, PIN_DRIVING_FLOAT);

    /* Debug UART 在 8、9 switch to AT UART */
    Hal_Pin_ConfigSet(8, PIN_TYPE_UART1_TX, PIN_DRIVING_FLOAT);
    Hal_Pin_ConfigSet(9, PIN_TYPE_UART1_RX, PIN_DRIVING_HIGH);
}
else
{
    /* AT UART 在 8、9 switch to Debug UART */
    Hal_Pin_ConfigSet(8, PIN_TYPE_UART_APS_TX, PIN_DRIVING_FLOAT);
    Hal_Pin_ConfigSet(9, PIN_TYPE_UART_APS_RX, PIN_DRIVING_HIGH);

    /* Debug UART 在 0、1 switch to AT UART */
```

```
Hal_Pin_ConfigSet(0, PIN_TYPE_UART1_TX, PIN_DRIVING_FLOAT);  
Hal_Pin_ConfigSet(1, PIN_TYPE_UART1_RX, PIN_DRIVING_FLOAT);  
}
```

3.3. APS 和 UART AT 串口转换

当固件设定的 IO0 和 IO1 是 AT UART，IO8 和 IO9 是 Debug UART，IO8 和 IO9 可以依照用户设计而制定其它 IO 脚位。在 AT UART 打入“at+switchdbg”可以改变 IO0 和 IO1 到 Debug UART。在 Debug UART 中，打入“switchat”可以改变 IO0 和 IO1 到 AT UART。可以参考，用户在使用上可以更清楚明白。

· 用户在使用上可以更清楚明白。

Table 1: AT/Debug UART Switch Command

指令 \ IO 脚位	IO 0、1	IO 8、9
at+switchdbg	Debug UART	AT UART
Switchat	AT UART	Debug UART

当用户一开始开机的时候，IO0 和 IO1 是 AT UART 功能。如 Figure 16 所示。敲入 at 的指令时，会出现 ok 的响应。表示当前工作模式是 AT UART 功能。

Figure 16 : AT UART work mode

```
><<CHECK>
SPI load patch, last index 614 result 2
BootMode 10
>at
OK
```

在 AT UART 功能下，用户可以透过 at+switchdbg 的指令切换到 Debug UART。当用户切换到 Debug UART 时，可以敲 at 指令，会得到 at 的响应，如 Figure 17 所示。当用户重新启动板子时，必须在重新敲入切换 Debug UART 的指令。

Figure 17: AT UART switch to Debug UART

```
>at+switchdbg
C!
Switch: Dbg UART
>
>atat
```

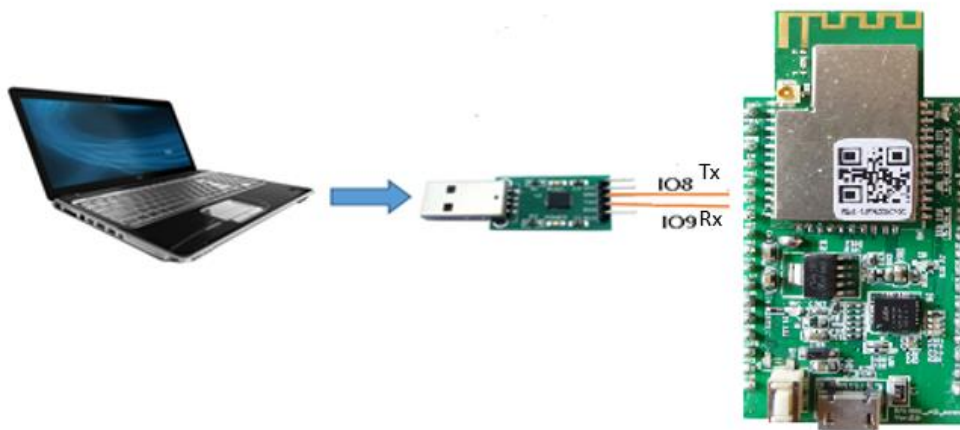
3.4. APS 串口连接和使用

DEVKIT 板两侧提供了两排扩展接口，其中包含 APS（Debug_prg）串口，实现和 M3 MCU 串口通信功能。APS 串口可以输出固件 log 打印功能。

3.4.1. APS 串口连接

APS 串口连接使用 IO8 和 IO9 两根管脚。IO8 是 APS 串口的 TX 输出信号线，接 UART 转接板的输入 RX 信号线。IO9 是 APS 串口 RX 信号线，接 UART 转接板的 TX 信号线。接线如下图所示。

Figure 18: APS 串口接线实例



3.4.2. 通过 APS 观察固件调试打印信息

使用 Tera Term 连接 Debug_Prg 串口（本例中为 COM58），波特率设置为 115200。复位 DEVKIT 板在信息输出窗口可以得到固件的打印信息。如 Figure 19 所示。

Figure 19: APS 串口输出打印信息

```
COM58 - Tera Term VT
文件(F) 编辑(E) 设置(S) 控制(O) 窗口(W) 帮助(H)
@<CHECK>
SPI load patch, last index 656 result 2

BootMode 10

The init of MW_FIM is done.
[OPL1000 A2 PATCH]
wifiMac Task create successful

Supplicant task is created successfully!

controller_queue creates successful!

controller_queue_ble creates successful!

controller_task_create successful!

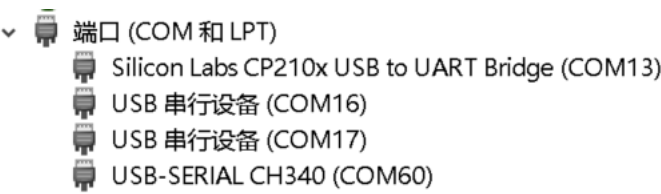
LE Task create successful

There is no any OTA image.
Applnit
```

3.5. AT 串口连接和使用

DEVKIT 板上的 mini USB 提供供电，固件下载。AT 串口所采用的 USB 转串口控制芯片为 CP210X，正确安装芯片驱动后，连接 DEVKIT 板。在 PC 设备管理器中可以观察到 CP210x 串口设备。下图给出的例子中 COM13 为 AT 串口，另外一个 CH340 串口设备 COM60 连接的是 APS 串口。

Figure 20: DEVKIT AT 串口设备



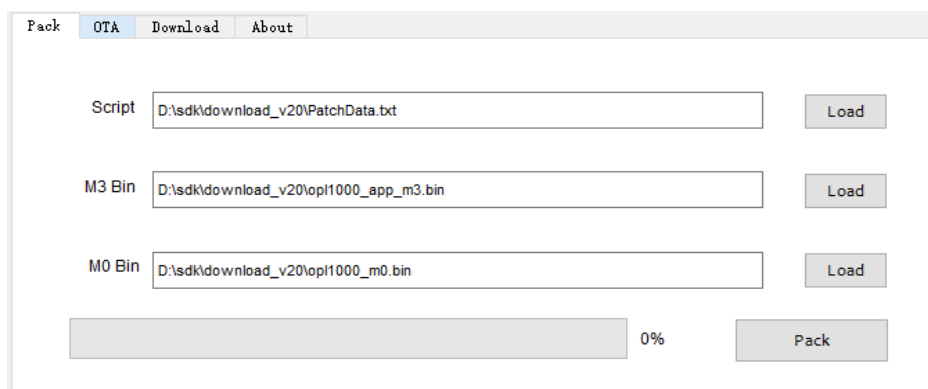
3.5.1. 通过 AT 串口更新固件

从编译工程到下载固件至 DEVKIT 板有 4 个步骤，以编译下载 hello_world 示例工程为例:

Copyright © 2017-2020, Opulinks. All Rights Reserved.
OPL1000--DEVKIT 电路板入门使用, UG1-03-2

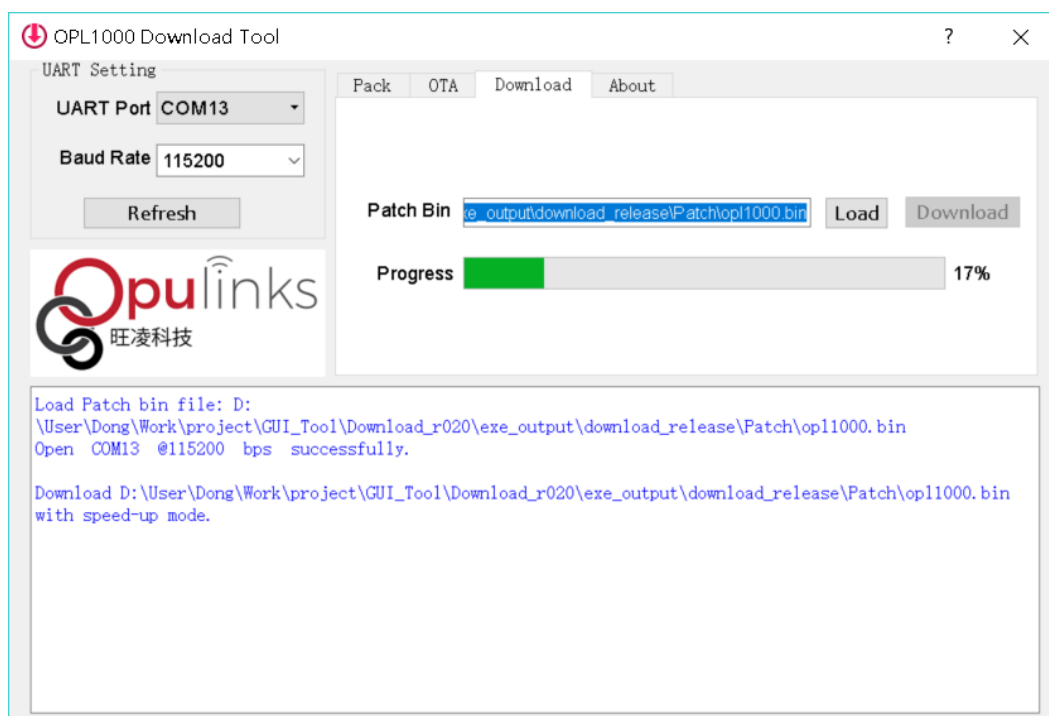
1. 使用 keil uVision(建议版本不低于 5.23)软件编译 SDK 的示例工程 hello_world。
目录：SDK\APS_PATCH\examples\get_started\hello_world
编译完成以后，在工程目录 Output\Objects 获得 opl1000_app_m3.bin。
2. 将编译得到的 opl1000_app_m3.bin 保存到 FW_Binary 目录下面。在 FW_Pack 目录下有固件合并脚本文件 PatchData.txt 和 M0 Bin。在 Pack 标签页点击 Script,M3 Bin, M0 Bin 文本框右侧的 Load 按钮，依次载入固件合并脚本文件、M3 bin 和 M0 bin 文件。点击 Pack 按钮。Pack 动作会把几个独立的 bin 文件合成为一个可供下载的 opl1000.bin 文件，存放在 download tool 同目录下的 Patch 子文件夹。

Figure 21:载入 M3/M0 Bin 文件进行合并操作



3. 选择 AT 串口端口号，波特率默认 115200bps。切换到 Download 选项，Patch Bin 路径已经正确填充为 opl1000.bin 文件，点击 Download 按钮，并在 5 秒之内复位 DEVKIT 板。download tool 自动识别到 DEVKIT 板复位以后，开始下载 opl1000.bin。进度条到达 100%，表示下载 opl1000.bin 成功。
对 OTA 固件下载过程和上面的一样。只是在 load 固件文件的时候，需要选取 opl1000_ota.bin (OTA image 文件)。

Figure 22: Patch Bin 文件下载



4. 下载完成后 DEVKIT 板会自动复位，Flash 的固件载入到 RAM 中执行。在“UART Port”选择 APS 串口端口，选择 115200 波特率，再次复位 DEVKIT 板，在串口调试工具中看到下图输出信息则表明固件下载正确。

Figure 23: 启动后 APS 串口输出 log 信息

```

BootMode 10 go to normal path

The init of MW_FIM is done.

[Lib] SDK version info: 1516
[Lib] Compile time: 2018/05/10 17:49:03

[SVN REV] SVN_REVISION:809
wifiMac Task create successful
Supplicant task is created successfully!
controller_queue creates successful!
controller_queue_ble creates successful!
controller_task_create successful!
LE Task create successful
Sw patch is changed successfully.
Hello world 1
Hello world 2
Hello world 3
Hello world 4
Hello world 5

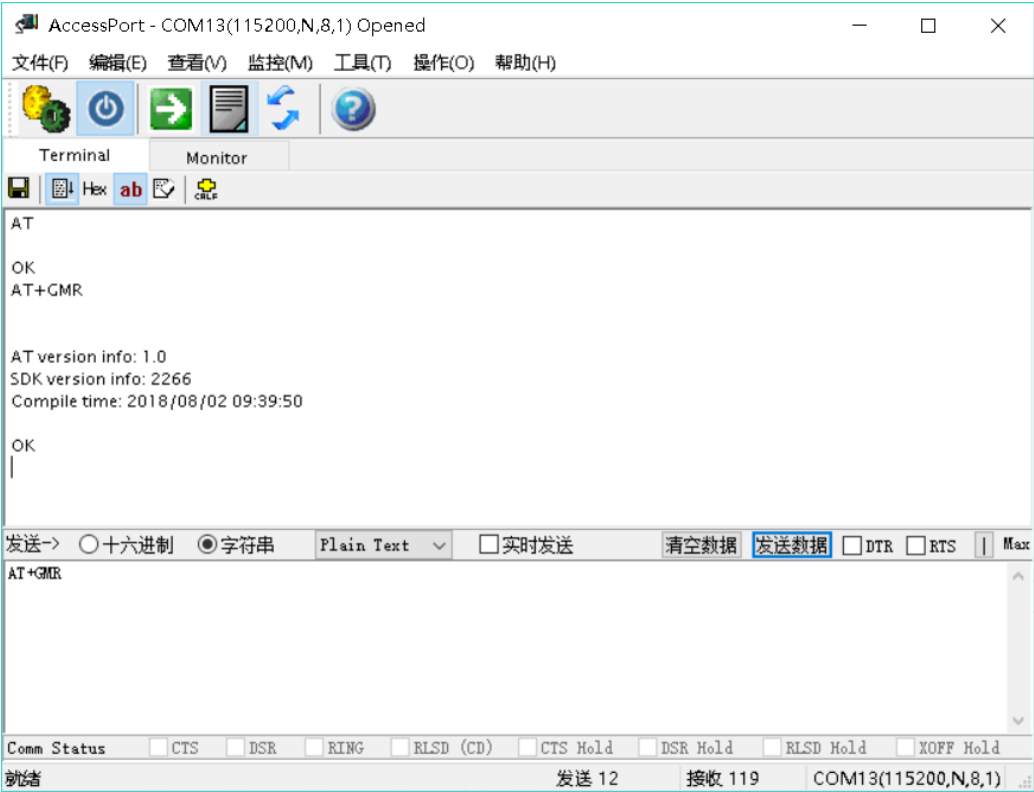
```

3.5.2. 使用 AT 串口执行 AT 指令

打开串口工具，选择 AT 串口号，波特率 115200。正常情况下，键入 ENTER (回车)，出现命令提示符 >，输入 at，得到 OK 的返回，则说明 AT 功能正常。

注：AT 命令的执行在 MP1.8 及其之后版本中要在 APS 口输入。

Figure 24: 执行 AT 指令



3.6. SWD 端口

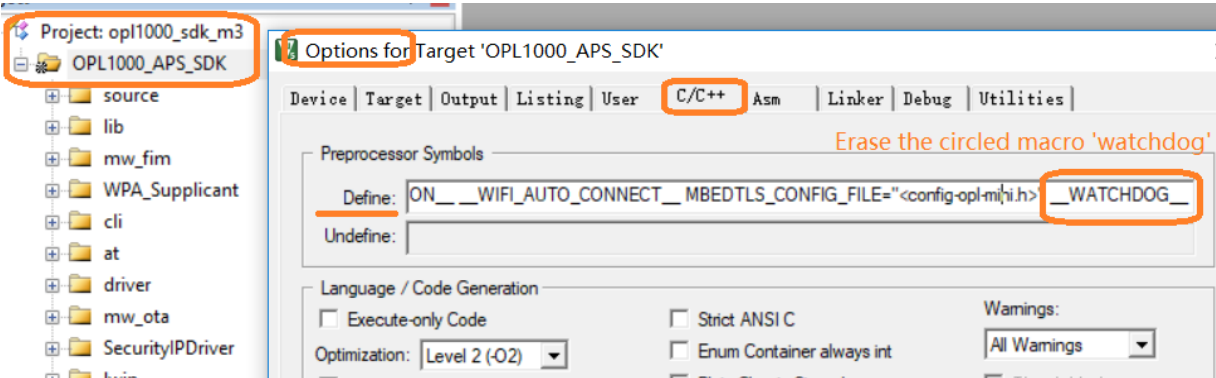
如果需要使用在线调试应用程序，则需要使用 ICE 仿真器，DEVKIT 板支持 M0 和 M3 四线 SWD 调试方式。由于用户 APP 在 M3 上执行，因此实际需要使用 M3 ICE 端口。后续章节将介绍如何禁用 watchdog 以及 M3 ICE 端口连接方式。

3.6.1. 禁用 watchdog

在默认情况下 OPL1000 A2 开启了 watchdog 功能。但用户可根据需要将 watchdog 功能关闭，以免在调试过程中被复位，方便使用 J-link 仿真器在线调试。方法如下：

1. 使用 keil uVision(建议版本不低于 5.23)软件打开 opl1000_sdk_m3 工程文件 (目录 : A2\SDK\APS\project\opl1000\) , 打开该工程的 options , 点击 'C/C++' tag, 在 'Define' 域中删除 ' __WATCHDOG__ ' 宏定义 , 然后 rebuild 该工程 ;

Figure 25: 禁用 watchdog



2. 用 keil 打开 hello_world 工程 (目录 : SDK\APS_PATCH\examples\get_started\hello_world) , rebuild 它。再参照 3.3.1 节的方法用 download tool 把生成的 opl1000_app_m3.bin 和 m0 bin 文件合并成为 opl1000.bin 文件 , 再将它烧录到 A2 板上。

烧录完毕 , 重启 A2 板后 , watchdog 功能就被禁用了。

注 : 在 rebuild 该工程前 , 请确认以下设置 :

在 hal_pin_config_project.h 文件中 IO20,IO21 做如下设置

```
#define HAL_PIN_TYPE_IO_20  PIN_TYPE_ICE_M3_DAT           // PIN_TYPE_NONE
#define HAL_PIN_TYPE_IO_21  PIN_TYPE_ICE_M3_CLK           // PIN_TYPE_NONE
```

3.6.2. M3 ICE 端口连接

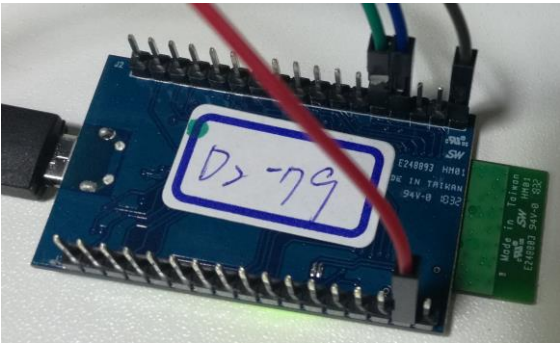
M3 ICE 端口接线对应关系如表 Table 2 所示。

Table 2: DEVKIT M3 SWD 信号连接

编号	SWD 信号	DEVKIT 板 J2 排针	J-Link 仿真器管脚	说明
1	3.3V	--	1	3.3 V 电源
2	GND	GND	4 - 20	GND
3	SWD_CLK	IO21	9	时钟信号
4	SWD_DAT	IO20	7	数据线

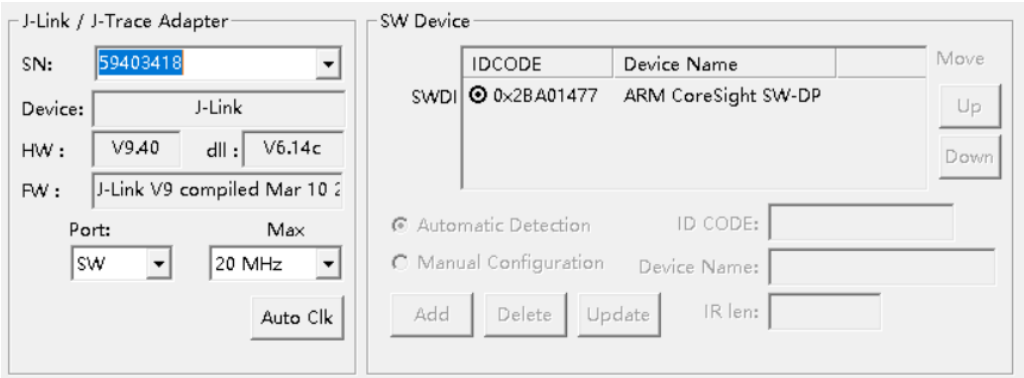
Notes: 对于多芯片 J-Link 仿真器 , 需确保编号为 1 的电源线也正确地连接。

Figure 26: DEVKIT 板上 M3 ICE 连线图



连接好 J-Link 仿真器和 DEVKIT 板，在 keil 工程的 debug 界面里面如果检测到 SW Device 的序列号，则说明连接正确，可以正常使用 SWD 开发和调试工程。如图 Figure 27 所示。

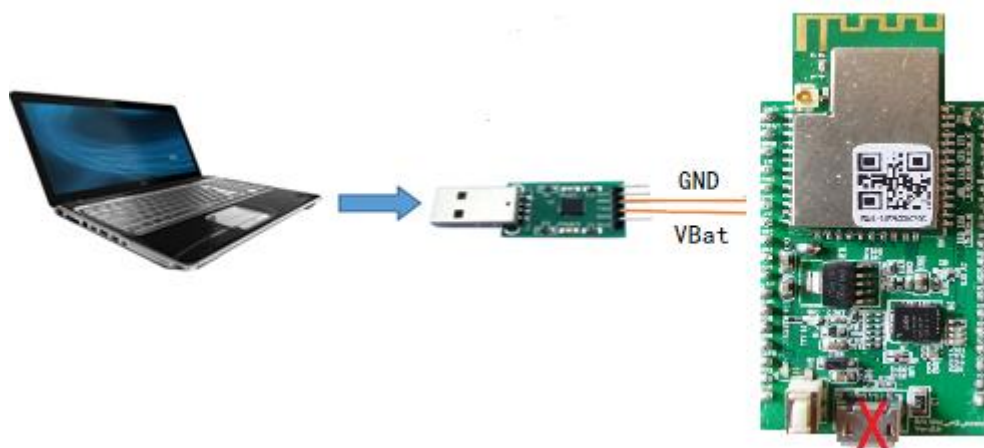
Figure 27: J-link ICE 仿真器正确识别



3.7. 外接供电

由于子板是最小模块，可以不经过母板，直接外部供电方式使其工作，这样可以利于低功耗的测量。此处以 USB 转接板接线为例，可以将转板供电改为 power monitor 供电，将 USB 转串口板的+3.3V 和 GND 脚位分别接到 DEVKIT 板的+3.3V 引脚和 GND，将 DEVKIT 的 USB 接口悬空，不接入任何设备

Figure 28: J-link ICE 仿真器正确识别



3.8. DEVKIT 复位

DEVKIT 复位有三种复位方式，均可达到设备重启的目的。

- 使用板子上硬件按钮直接 reset。
- 使用外部低脉冲信号触发复位，将要触发的低脉冲信号接到 DEVKIT 板子的 chip_en 或者 rst，同时保持共地，脉冲信号触发的时序图，可以参考 [HDK 文档](#)。
- 使用 AT 命令 at+rst 完成复位，打开串口工具，通过串口转板，发送 AT 命令到 AT UART 口。

CONTACT

sales@Opulinks.com