
How To Use SWO Trace With MCUXpresso IDE

[一 . Introduction](#)

[二 . Requirements](#)

[三 . Using SWO Trace Based on NEW SDK Project and SDK Demo](#)

[四 . Reference:](#)

一 . Introduction

SWO is Serial Wire output, it is a debug interface of Cortec-M3 and M4 core.

The SWO Trace provide access to the memory of a running target without needing to interrupt target. It just requires one extra pin from MCU in addition to the standard SWD connection. It is easy to send debug information. If want to know more detail about SWO Trace, you can have a look at <MCUXpresso_IDE_SWO_Trace> .

The document of <MCUXpresso_IDE_SWO_Trace> has descripted using SWO Trace function in MCUXpresso IDE detailly, so in my article , I just demonstrate the steps using SWO Trace on MCUXpresso IDE, based on LPCXpresso54114 board.

二. Requirements

[MCUXpresso IDE v10.2.1 or later](#)

[MCUXpresso SDK v2.4.1 for LPC54114](#)

[LPCXpresso54114 development board](#)



SECURE CONNECTIONS
FOR A SMARTER WORLD

Micro USB Cable

Personal Computer

≡. Using SWO Trace Based on NEW SDK Project and SDK Demo

3.1 Using SWO Trace Based on NEW SDK Project

Step1. Enable “Redirect printf/scanf to ITM”/Add “retarget_itm.c”

Enable the option of “Redirect printf/scanf to ITM” in “Advanced project settings” page of new SDK project wizard (Figure 1). If you forget to check it when creating new project, it doesn't matter, just add the file “retarget_itm.c” into project, you can find this file inside your MCUXpresso IDE installation: MCUXpressoIDE_10.2.1_795\ide\Examples\Misc (Figure 2).

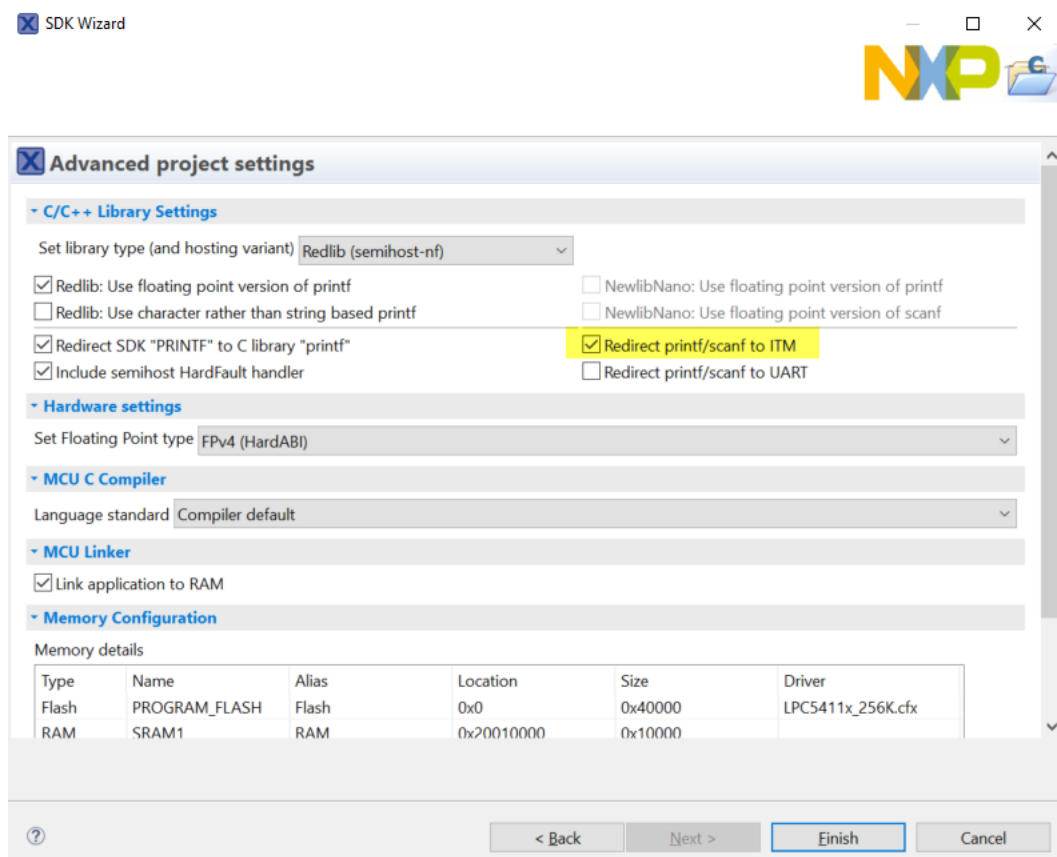


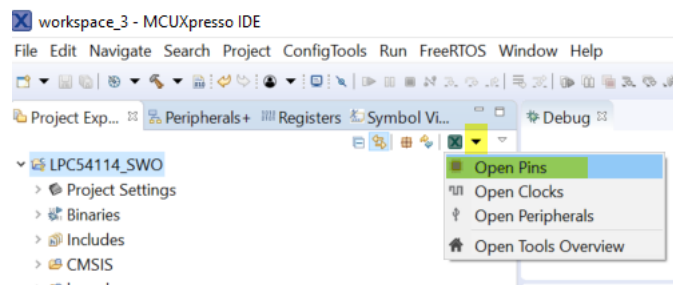
Figure 1

Name	Date modified	Type
_cr_check_heap	2018/7/25 10:02	C Source File
retarget_itm	2018/7/25 10:02	C Source File

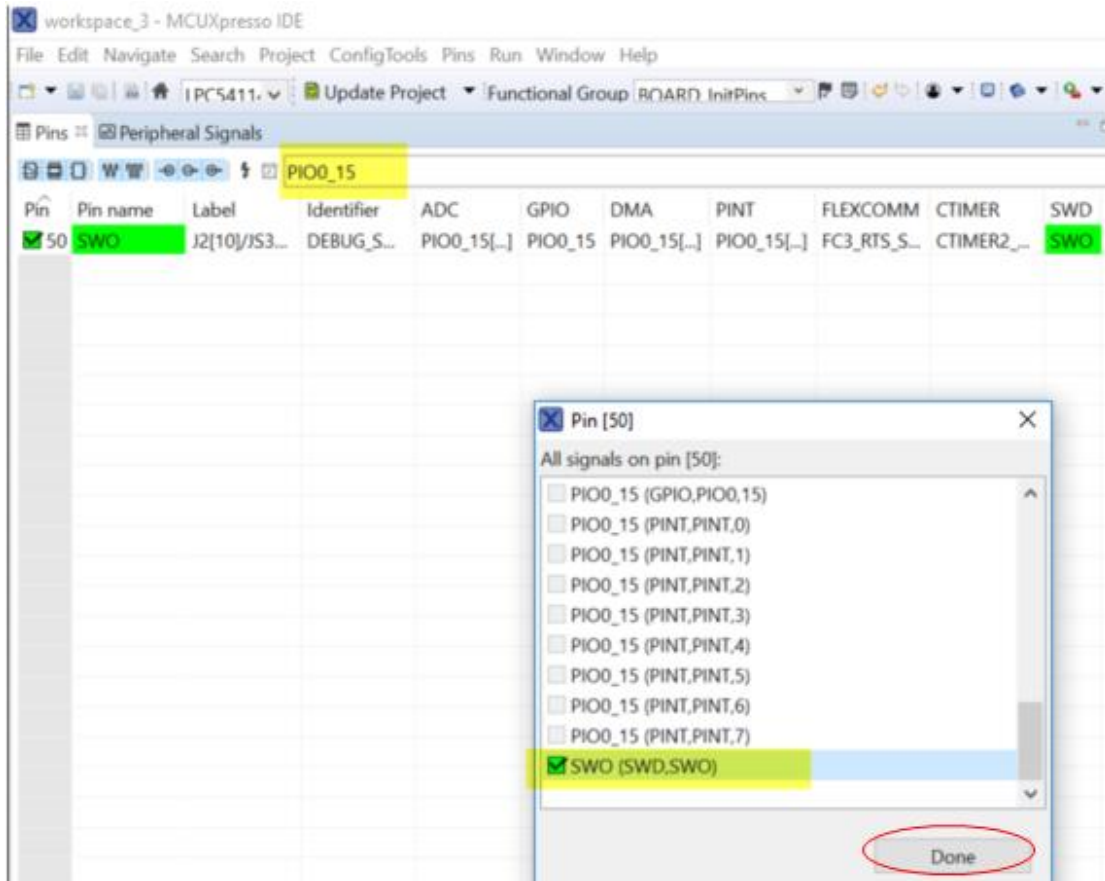
Figure 2

Step2. Enable SWO Pin

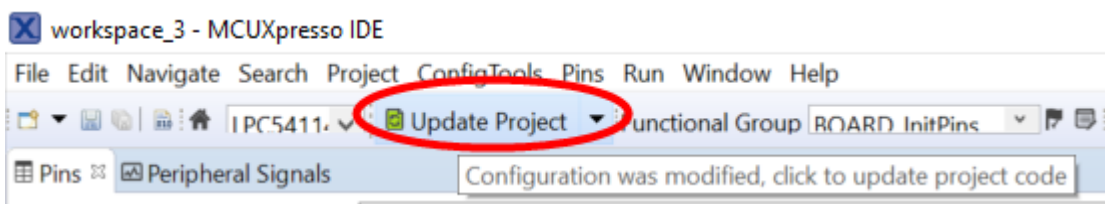
Open Pins configuration view:



From board schematic, PIO0_15 is set to SWO pin, so configure this pin to SWO function in Pins configuration view:



Update Project code through "Update Project" button, then you can find the generated code in pin_mux.c file:



```

76 /* PORT0 PIN0 (coords: 31) is configured as FC0_RXD_SDA_MOSI */
77 IOCON_PinMuxSet(IOCON, 0U, 0U, port0_pin0_config);
78
79 const uint32_t port0_pin1_config = (/* Pin is configured as FC0_TXD_SCL_MISO */
80 IOCON_PIO_FUNC1 |
81 /* No addition pin function */
82 IOCON_PIO_MODE_INACT |
83 /* Input function is not inverted */
84 IOCON_PIO_INV_DI |
85 /* Enables digital function */
86 IOCON_PIO_DIGITAL_EN |
87 /* Input filter disabled */
88 IOCON_PIO_INPFILT_OFF |
89 /* Standard mode, output slew rate control is en
90 IOCON_PIO_SLEW_STANDARD |
91 /* Open drain is disabled */
92 IOCON_PIO_OPENDRAIN_DI);
93
94 /* PORT0 PIN1 (coords: 32) is configured as FC0_TXD_SCL_MISO */
95 IOCON_PinMuxSet(IOCON, 0U, 1U, port0_pin1_config);
96
97 IOCON->PIO[0][15] = ((IOCON->PIO[0][15] &
98 /* Mask bits to zero which are setting */
99 (~((IOCON_PIO_FUNC_MASK | IOCON_PIO_DIGIMODE_MASK)))
100
101 /* Selects pin function.
102 * : PORT015 (pin 50) is configured as SW0. */
103 | IOCON_PIO_FUNC(PIO015_FUNC_ALT2)
104
105 /* Select Analog/Digital mode.
106 * : Digital mode. */
107 | IOCON_PIO_DIGIMODE(PIO015_DIGIMODE_DIGITAL));
108 }

```

Set3. Enable SWO Trace clock

The startup code in new SDK project has contain the enable SWO code:

```

470 SectionLen = *SectionTableAddr++;
471 bss_init(ExeAddr, SectionLen);
472 }
473
474 // Optionally enable Cortex-M4 SWV trace (off by default at reset)
475 // Note - your board support must also set up pinmuxing such that
476 // SWO is output on GPIO PIO0-15 (FUNC2) or PIO1_1 (FUNC2).
477 #if !defined(DONT_ENABLE_SWTRACECLK) && !defined(CORE_M0PLUS)
478 volatile unsigned int *TRACECLKDIV = (unsigned int *) 0x40000304;
479 volatile unsigned int *SYSAHBCLKCTRLSET = (unsigned int *) 0x40000220;
480 volatile unsigned int *SYSAHBCLKCTRL = (unsigned int *) 0x40000200;
481 // Write 0x00000000 to TRACECLKDIV - Trace divider
482 *TRACECLKDIV = 0;
483 // Enable IOCON peripheral clock (for SWO on PIO0-15 or PIO1_1)
484 // by setting bit13 via SYSAHBCLKCTRLSET[0]
485 *SYSAHBCLKCTRLSET = 1 << 13; // 0x2000
486 // Read SYSAHBCLKCTRL[0] and check bit 13
487 __SWVtrace_Enabled = ((*SYSAHBCLKCTRL & 1 << 13) == 1 << 13);
488 #endif

```

Step4. Add Code to Send Text

Use “printf()” to output some message you want to send when debugging. For example, in this simple demo:

```
printf("SWO Trace demonstration!\n");
```

Step5. Download project

Using USB cable connect board interface J7 to PC, download your project.

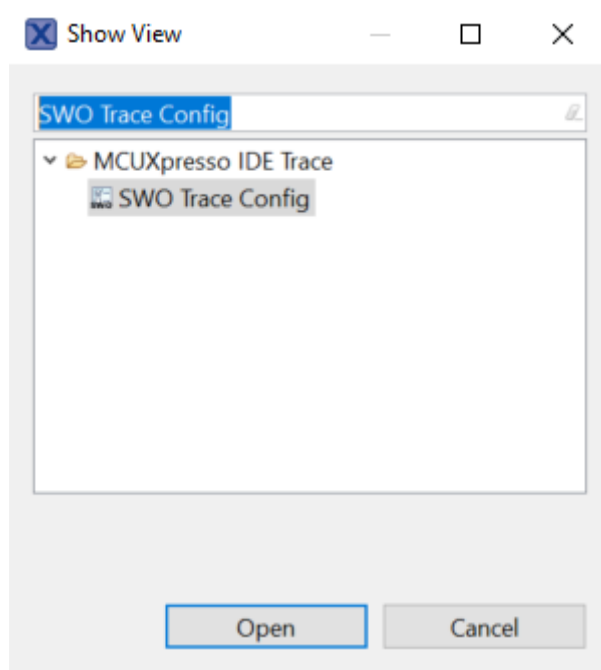


SECURE CONNECTIONS
FOR A SMARTER WORLD

Step6. Configure MCUXpresso IDE

- Configure SWO Trace clock speed:

Open SWO Trace configuration view through “Window->Show View->Other->SWO Trace Config”



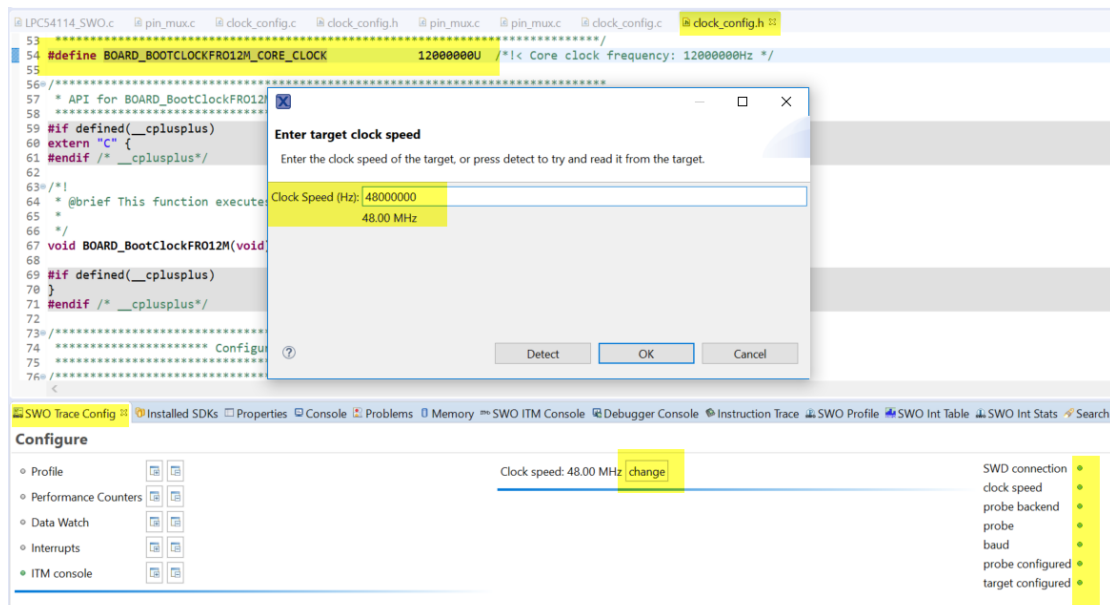
Click “change” button in “SWO Trace Config”, shows Clock Speed options, you can use “Detect” button to automatically read system clock of MCU from the “SystemCoreClock” global variable. When new a project based on SDK, this variable has been set to the system clock of processor. The default system clock in SDK new project is 48MHZ.

There is an explanation about why Clock Speed need equal to system speed from < MCUXpresso_IDE_SWO_Trace>:

“Due to the way the Trace data is transferred by the Cortex CPU within the MCU, setting the correct clock speed within the SWO Trace interface is essential to

determine the correct baud rate for the data transfer. If the clock speed setting does not match the actual clock speed of the processor then data will be lost and/or corrupted. This can result in no data being visualized, or unexpected Trace data.”

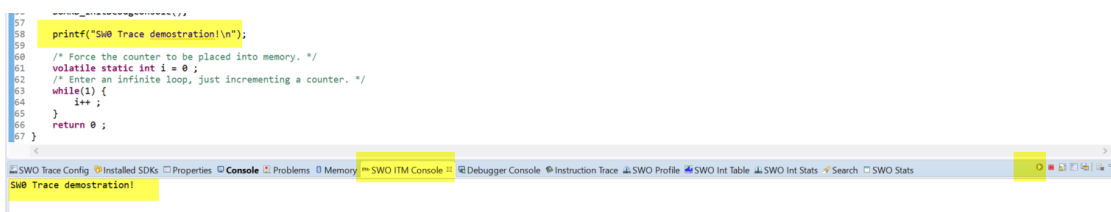
And if it can't detect the system clock correctly, you can also input it by manually.



- Open Trace Views to observe result.

You can open all the Trace views by “Window->Show View->Other-> (input the view name to search)”

For example Open “SWO ITM Console” view to read debug information:



“SWO Int Stats” provides counts and timing information for interrupts.

There are also “SWO Int Trace” “SWO Int Table” “SWO Data” and others SWO

Trace view, about functions of them , please refer to <MCUXpresso_IDE_SWO_Trace> -> 1.2 SWO Trace: Views.

3.2 Using SWO Trace Based on SDK Demo

If you development based on SDK demo, the steps using SWO Trace are the same with “3.1 Using SWO Trace Based on NEW SDK Project”. Just pay attention some points: There isn't “Redirect printf/sanf to ITM” option when import SDK demo, so you need add the file “retarget_itm.c” into project as 3.1 step1. Some of the SDK demos have already enable SWO Pin, please check pin_mux.c file. Also double check the actual system clock in SDK demo.

四. Reference:

< MCUXpresso_IDE_SWO_Trace >

< MCUXpresso_IDE_User_Guide >



SECURE CONNECTIONS
FOR A SMARTER WORLD