How to use the SWO on STM32H7 devices?

Document created by Amel N on Jun 21, 2018 • Last modified by Amel N on Jun 22, 2018
Version 8

This tutorial is based on *Keil Cortex™-M3 Hands-On LAB featuring Serial Wire Viewer (Version 4.04 May, 2008)*

The configuration of the SWV needs to be done for each project you use whenever ITM output is required. It will be saved when you close a project or µVision.

There are two main parts to be considered in this setup:

- Select the Serial Wire Debug Port instead of the JTAG port (step 6 below)
- Configure the Serial Wire Output (steps 4 and 7)

Steps:

µVision must not be in debugging mode for these steps. If it is, click on the debugger icon 
The menus below will be grayed out if µVision is in debugger mode.

1- Select your project and make sure the correct processor is selected. (Project/Open Project).

2- Open the Debug setup menu. This can be accessed three ways:

- Click on Flash/Configure Flash Tools and select the Debug tab.
- Click on Project/Options for target “STM32H743ZI”. Select the Debug tab.
- Click on Options For Target icon on main µVision toolbar  . Select the Debug tab.

The third option is probably the easiest to use, but whichever one you use, Figure 1 opens up:

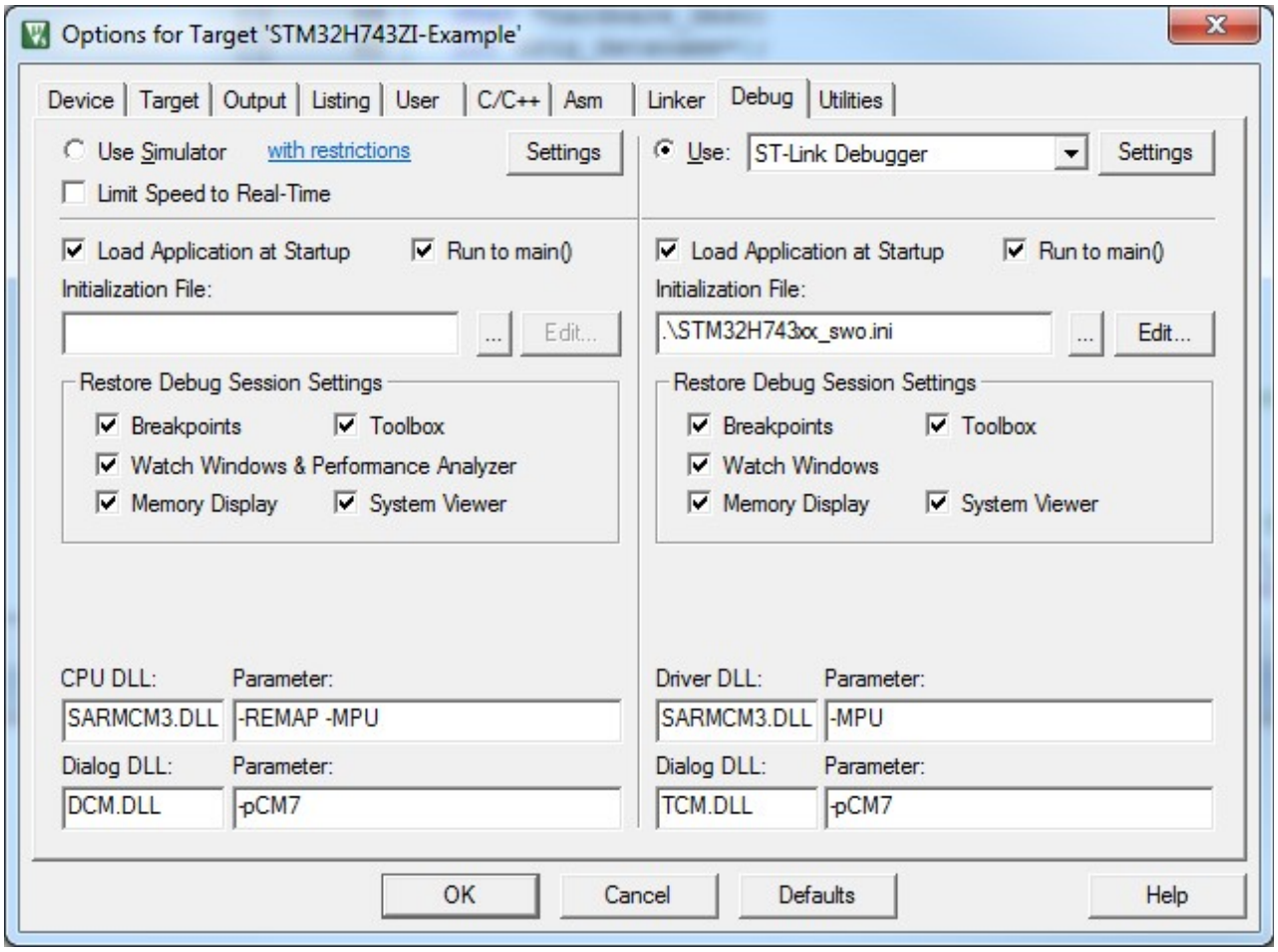


Figure 1 : Option for Target

3- Select ST-Link Debugger

4- The attached file STM32H743xx_swo.ini can be located in the the project directory. It configures the SWV ports.

In the box Initialization File: select the file STM32H743xx_swo.ini as shown in Figure 1.

You can rename and/or relocate this file if you prefer. It will not hurt if this is enabled for every exercise.

5- Click on Settings to configure the ST-Link. Figure 2 opens up.

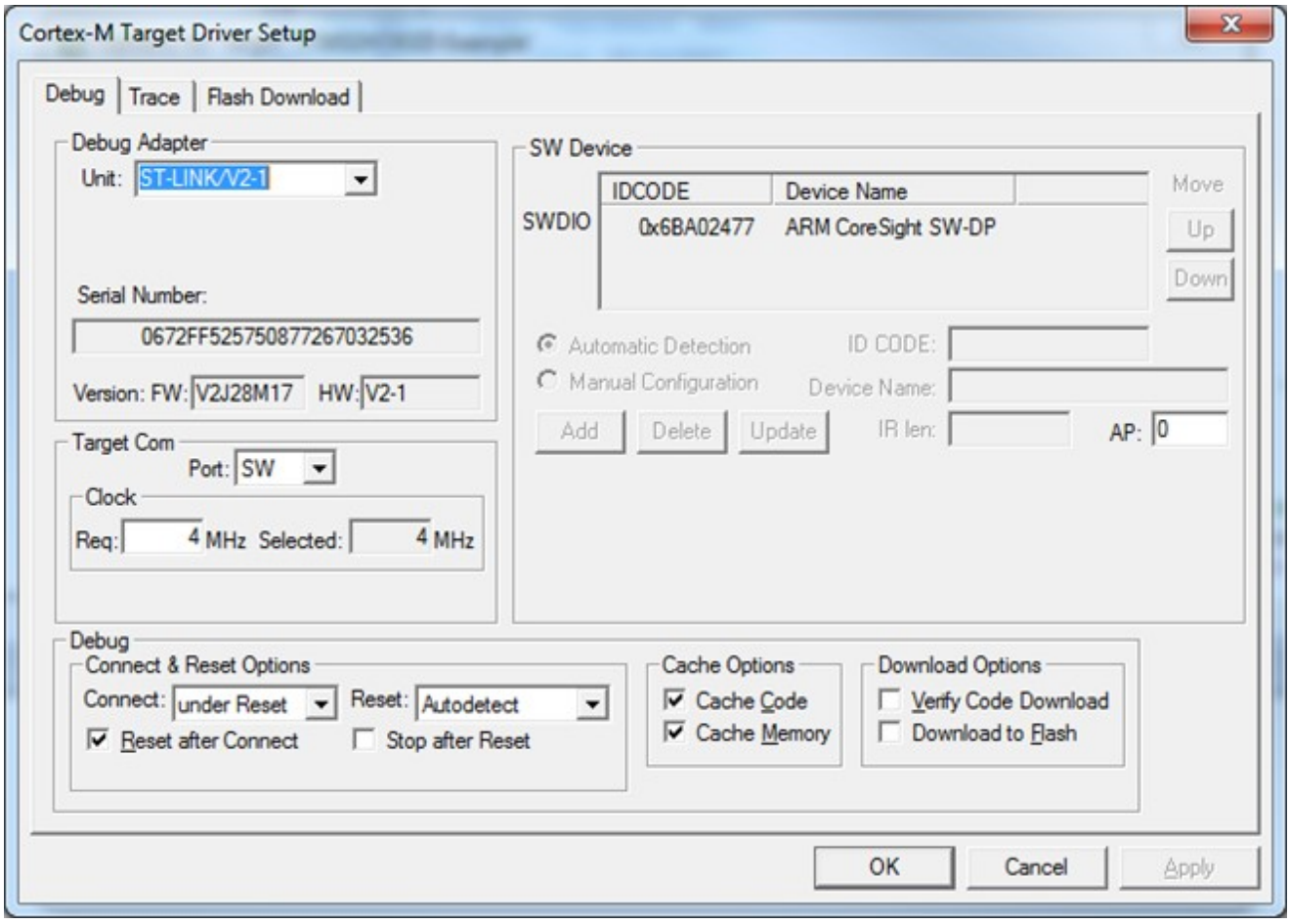


Figure 2 : Target Driver Setup

6- Select the SWJ box and set Port: to SW as shown in Figure 2. This must not be set to JTAG.

SWV operates only through the Serial Wire debug port (SW). Max clock @ 4 MHz is correct.

7- Select the Trace tab and Figure 3 opens up to configure the SWV trace:

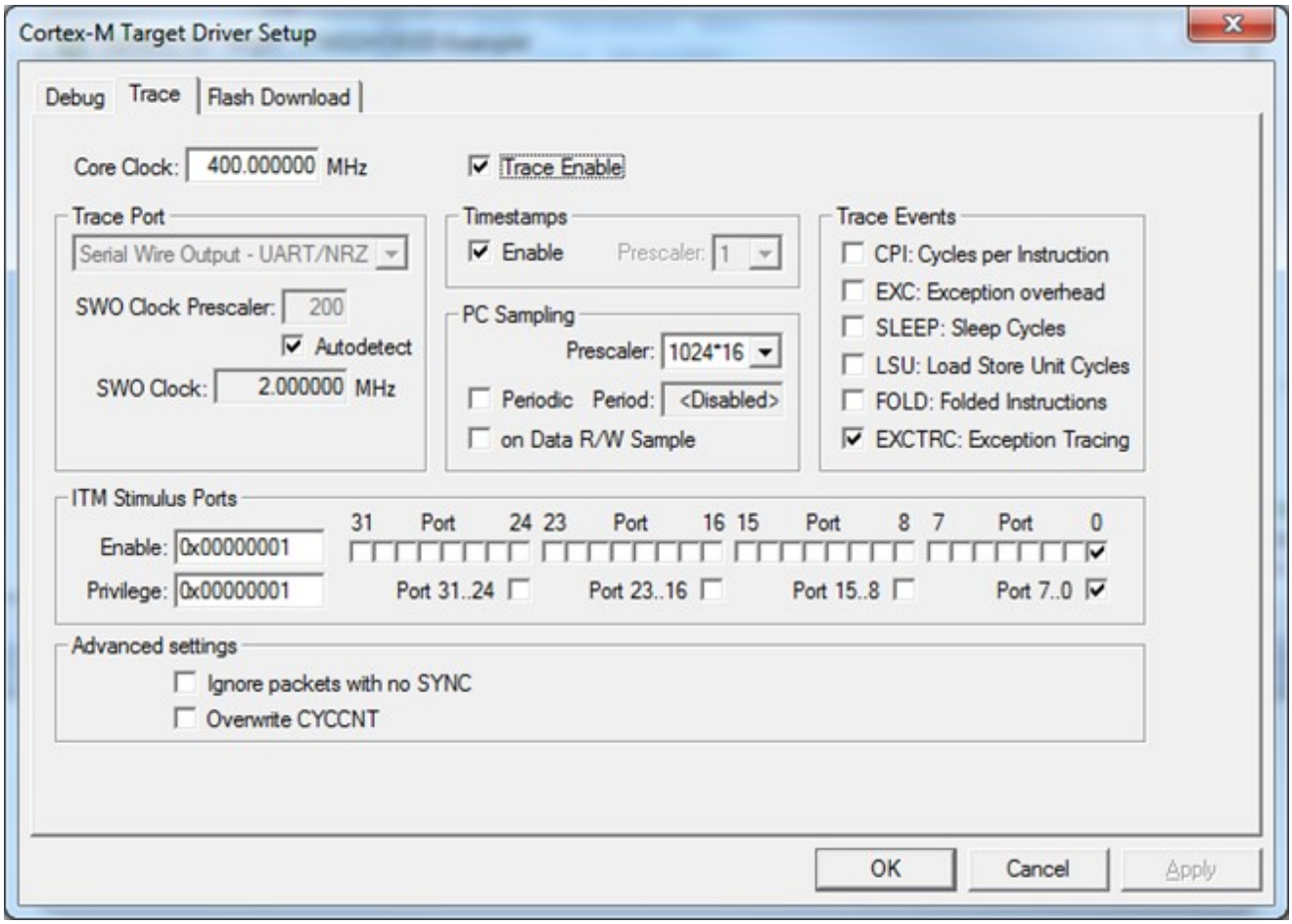


Figure 3 : Trace Setup

8- Important Step: Set the Core Clock to 400 MHz and check the Trace Enable box. Ensure in the ITM Stimulus Ports that at least Port 0 and Port 7..0 are selected. The rest are Don't Care for this exercise.

9- Click on OK twice. The Serial Wire Viewer Trace is now configured and ready to use !

10- Click on File/Save to save these settings.

1. When in the debugger mode, if the SWV windows do not update when the program is running and only do when you stop the program execution, make sure View/Periodic Window Update is activated.
2. If Core Clock is different of 400 MHz, pay attention to set correct divisor value in initialization file (In following example of .ini file, refer to line 16 to calculate the new value and line 17 to set it)

Example of Initialization file:

```
1  /*****************************************************************
2  /* STM32H743_SWO.ini: STM32H743 Debugger Initialization File
3  /*****************************************************************
4
```

```
FUNC void DebugSetup (void) {

  _WWORD(0x5C001004,0x00700000);

  //UNLOCK FUNNEL
  _WWORD(0x5C004FB0,0xC5ACCE55);
  _WWORD(0x5C003FB0,0xC5ACCE55);

  //SWO current output divisor register
  //This divisor value (0x000000C6) corresponds to 400Mhz
  //To change it, you can use the following rule
  // value = (CPU Freq/2000)-1
  _WWORD(0x5C003010,((_RWORD(0x5C003010) & 0xfffff000) | 0x000000C7))

  //SWO selected pin protocol register
  _WWORD(0x5C0030F0, 0x00000002);

  //Enable ITM input of SWO trace funnel
  _WWORD(0x5C004000, (_RWORD(0x5C004000) | 0x00000001));

  //RCC_AHB4ENR enable GPIOB clock
  _WWORD(0x580244E0, (_RWORD(0x580244E0) | 0x00000002));

  // Configure GPIOB pin 3 as AF
  _WWORD(0x58020400,((_RWORD(0x58020400) & 0xffffff3f) | 0x00000080))

  // Configure GPIOB pin 3 Speed
  _WWORD(0x58020408, (_RWORD(0x58020408) | 0x00000080));

  // Force AF0 for GPIOB pin 3
  _WWORD(0x58020420, (_RWORD(0x58020420) & 0xFFFF0FFF));
}

DebugSetup();                          // Debugger Setup

FUNC void OnResetExec (void)  {         // executes upon software RESE
  DebugSetup();                        // call the initialization func
}
```

4 people found this helpful

ATTACHMENTS

STM32H743xx_SWO.ini.zip
736 bytes

OUTCOMES

Helpful(2)

Visibility: STM32 MCUs Forum • **207 Views**

Last modified on Jun 22, 2018 2:53 PM

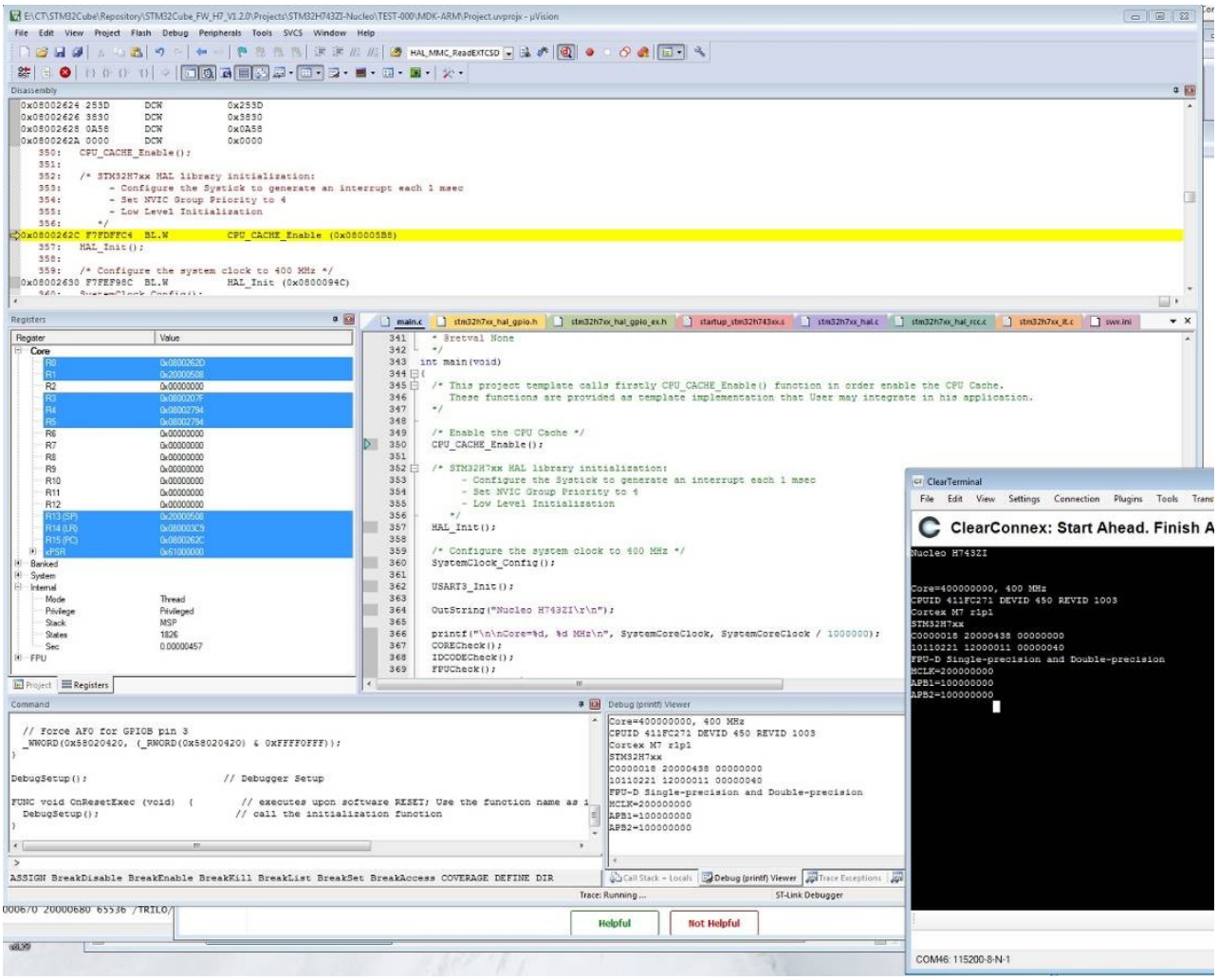
Tags: stm32h7 itm swo trace

0

8 Comments

Clive Two.Zero
Jun 21, 2018 6:28 PM

Amel N Thanks, that seems to be working at 400 MHz, didn't seem to at 200 MHZ, but might need to tweak the settings.



Actions

Like • 1

STM 32

@ Clive Two.Zero on Jun 21, 2018 6:46 PM

Dear Clive Two.Zero,

I think you have prescaler value issue. I think you have used .ini file as it is. If you look at the line 13 in the .ini file:

```
//SWO current output divisor register
  //This divisor value (0x000000C6) corresponds to 400Mhz
  //To change it, you can use the following rule
  // value = (CPU Freq/2000)-1
  _WWORD(0x5C003010,((_RWORD(0x5C003010) & 0xfffff000) | 0x000000C6));
```

The divisor you are using corresponds to 400MHz. So please replace 0x000000C6 by 0x00000064 for 200MHz.

Best Regards,
STM32

Actions Like • 0

Clive Two.Zero
@ STM 32 on Jun 21, 2018 7:04 PM

>>The divisor you are using corresponds to 400MHz. So please replace 0x000000C6 by 0x00000064 for 200MHz.

For the math and the formulas to be coherent here

400 -> 200-1 -> 0x0C7
200 -> 100-1 -> 0x063

You used the 0x0C7 value here https://community.st.com/message/203009-re-stm32h7-swo-printf-not-working?commentID=203009#comment-203009

Actions Like • 0

STM 32
@ Clive Two.Zero on Jun 21, 2018 7:10 PM

Sorry for the typo..
Yes the right value is 0xC7 for 400MHz.
0x63 corresponds to 200MHz.

B.R.
STM32

Actions Like • 0

Clive Two.Zero
@ Clive Two.Zero on Jun 21, 2018 7:12 PM

Seems to be tolerant of 0x0C7 or 0x0C6 in Keil when running core at 400 MHz, 0x063 works for a 200 MHz SYSCLK

Both 200 and 400 MHz clocking schemes AHB clock (and FLASH) at 200 MHz

Actions Like • 0

Clive Two.Zero
@ Clive Two.Zero on Jun 21, 2018 7:20 PM

The script Amel posted above uses the 0xC6 value, it does appear to work, and I've confirmed that Keil/ST-LINK doesn't change the SWV settings (computes 200 divisor in the debug dialog).

Printing out the register application side I see 0x000000C6 set by the script.

Actions Like • 0

Amel N
@ Clive Two.Zero on Jun 22, 2018 2:57 PM

Good catch Clive! I updated the .ini file to put the correct value (0xC7) calculated based on the formula although 0x0C6 is working for 400 MHz.

-Amel

Actions Like • 0

Clive Two.Zero
@ Clive Two.Zero on Jun 21, 2018 7:36 PM

Takes a licking and keeps on ticking...

Core=500000000, 500 MHz
CPUID 411FC271 DEVID 450 REVID 1003
Cortex M7 r1p1
STM32H7xx
C0000018 20000438 00000000
10110221 12000011 00000040
FPU-D Single-precision and Double-precision
HCLK=250000000
APB1=125000000

APB2=125000000

0x5C003010 -> 000000F9

1 person found this helpful

Actions Like • 2

Related Content

- Wireless Programming and Debugging with STM32 and RPi
- FAQ: Register Protection of SPC560Dxx
- How to run&debug from RAM without flashing flash STM32 devices (Keil)
- HAL_labs.pdf
- How to create an open STM32 Project

Recommended Content

- STM32H7 CubeMX FatFs + uSD + DMA + RTOS Success
- STEVAL-FCU001V1 - BLE app for smartphone
- First prototype of frame 3D printed
- FAQ for FCU (STEVAL-FCU001V1)
- Smart Ball using STM32 IoT node and Node-Red

Incoming Links

- Re: No traceswo output on stm32H7xx
- Re: STM32H743 NUCLEO SWO printf debug issue
- Re: STM32H7 SWO printf not working

Home | Top of page | Help