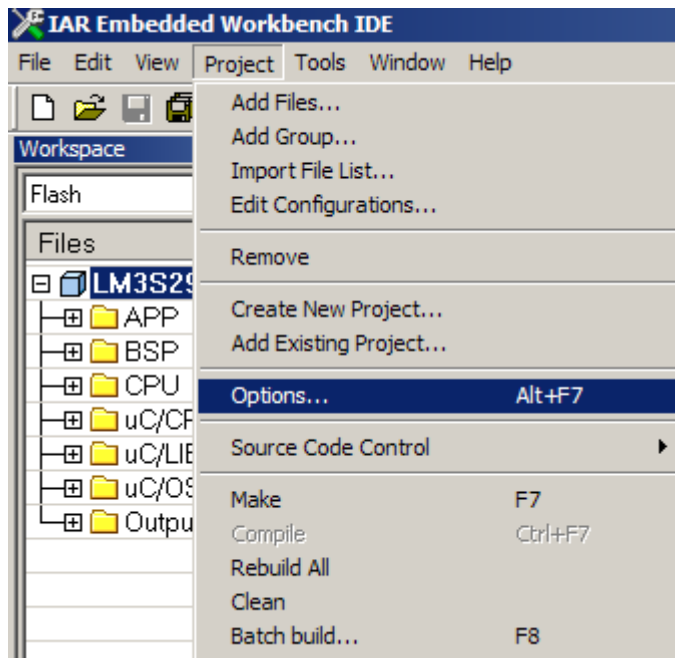


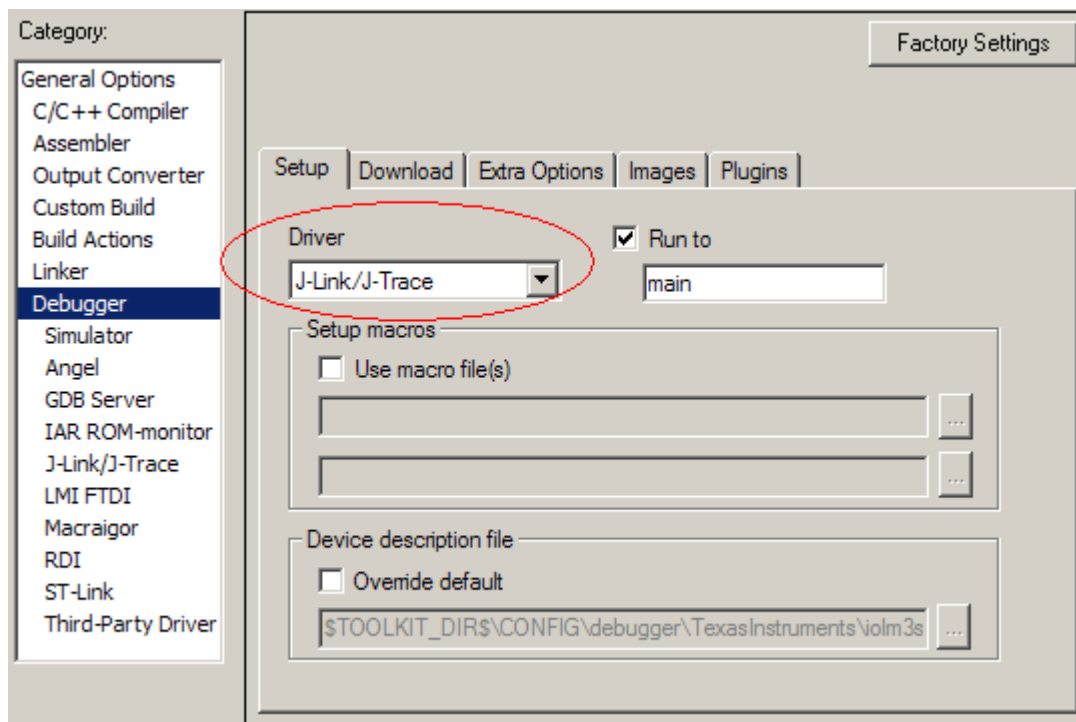
## Measuring the time consumption of a piece of code by Cortex-M3 SWD/SWV

IDE: IAR Embedded Workbench for ARM (v5.40)  
Hardware: Any target board based on ARM Cortex-M3 MCU  
Debugger: J-Link (version 6.0 or above)

1. Open the “Options” dialog of your project in EWARM:

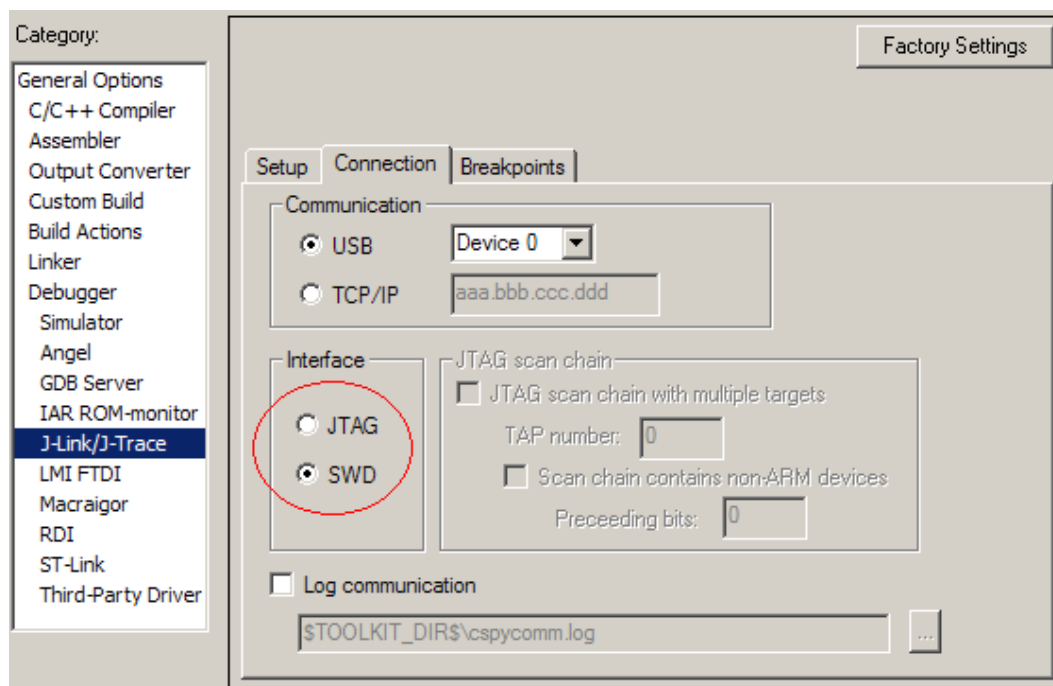


2. Select “J-Link/J-Trace” as the debug probe:



## Measuring the time consumption of a piece of code by Cortex-M3 SWD/SWV

3. Select “SWD” as the debug interface:



4. Add a macro “ITM\_Port8” into your code:

```
#define ITM_Port8(n)    (*((volatile unsigned char *) (0xE0000000+4*n)))
```

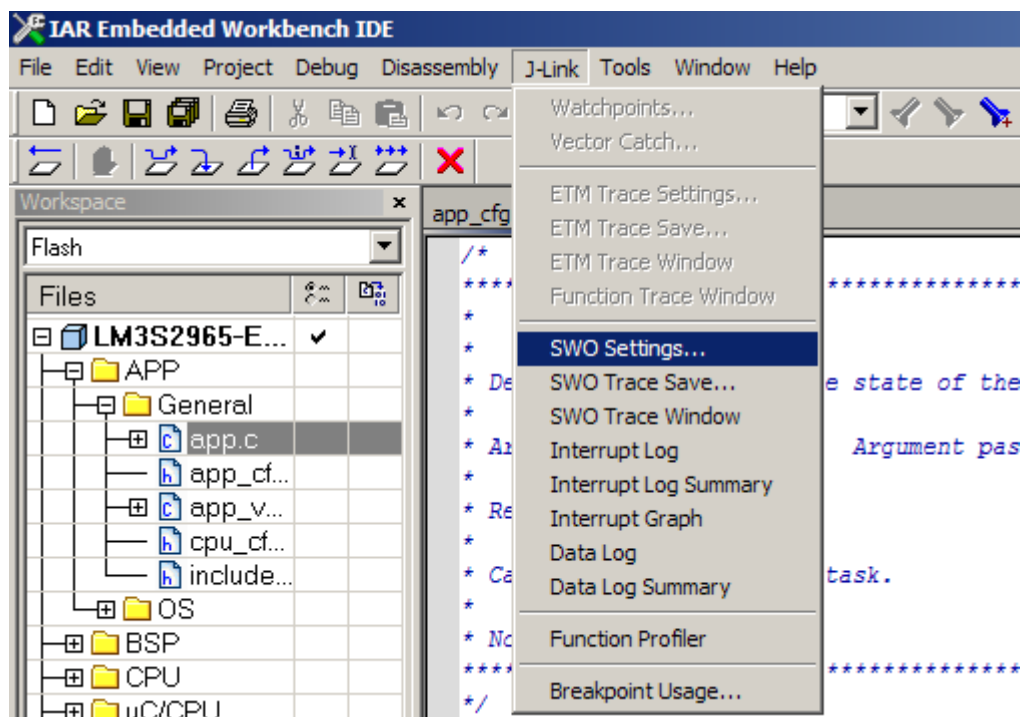
Add following statement at the beginning of the code to be measured:

```
ITM_Port8(0) = 0;
```

Add following statement at the end of the code to be measured:

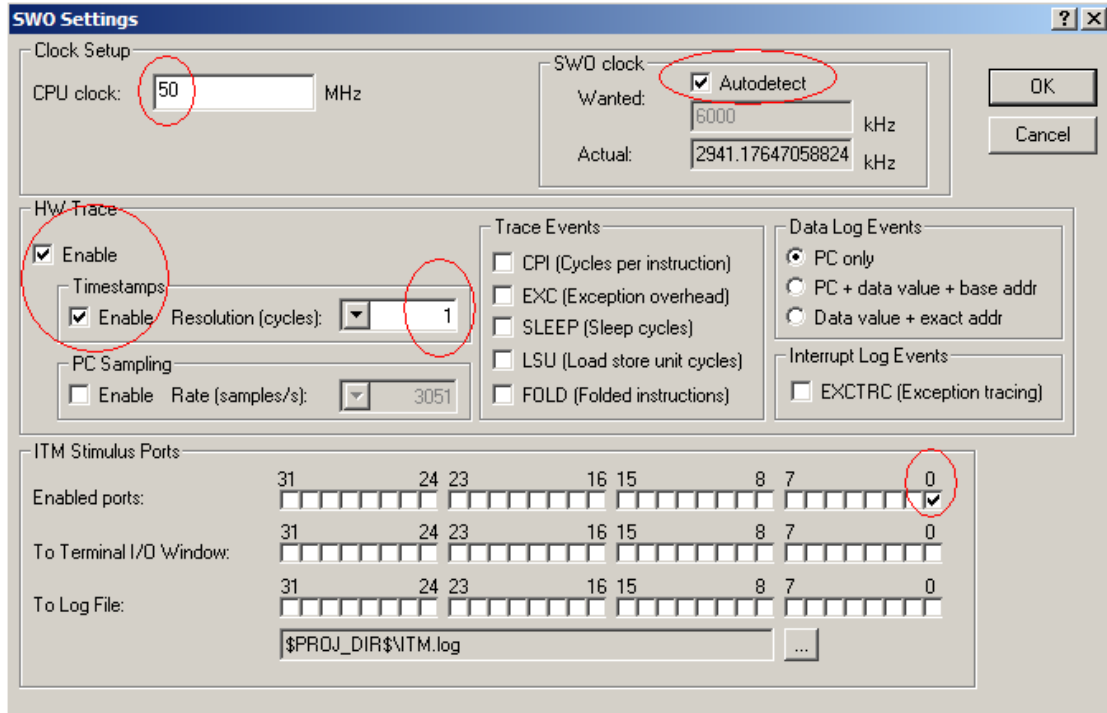
```
ITM_Port8(0) = 1;
```

5. Rebuild your project. Start the C-SPY debugger. Open the “SWO Settings” dialog:



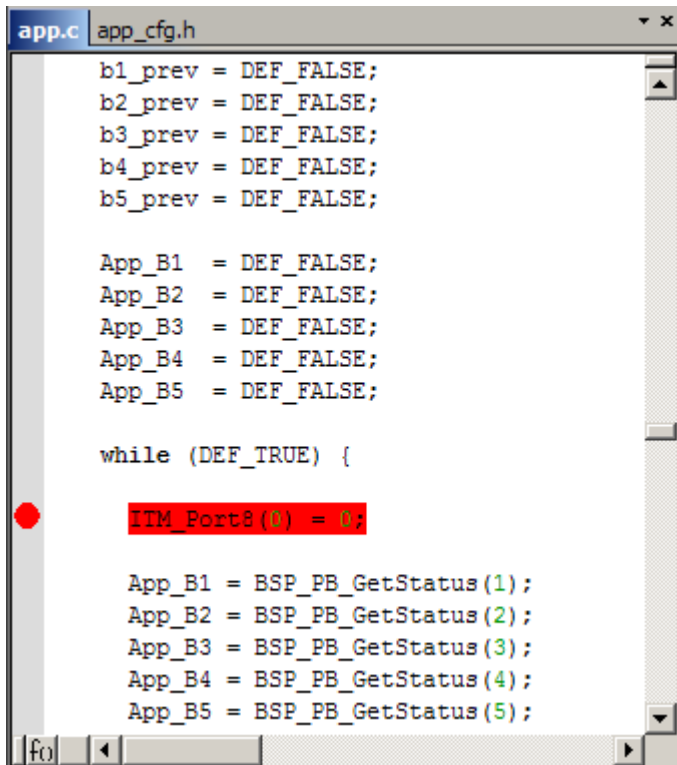
## Measuring the time consumption of a piece of code by Cortex-M3 SWD/SWV

- Configure necessary options for SWO Settings:
  - Set “CPU clock” according to the actual frequency of your target system;
  - Enable “HW Trace” and “Timestamps”. Set “Resolution (cycles)” to 1;
  - Enable “Autodetect” for “SWO clock”;
  - Enable the ITM Stimulus Port 0.



## Measuring the time consumption of a piece of code by Cortex-M3 SWD/SWV

7. Add two breakpoints before and after the code to be measured. The latter one should be set on the next statement of "ITM\_Port8(0) = 1".

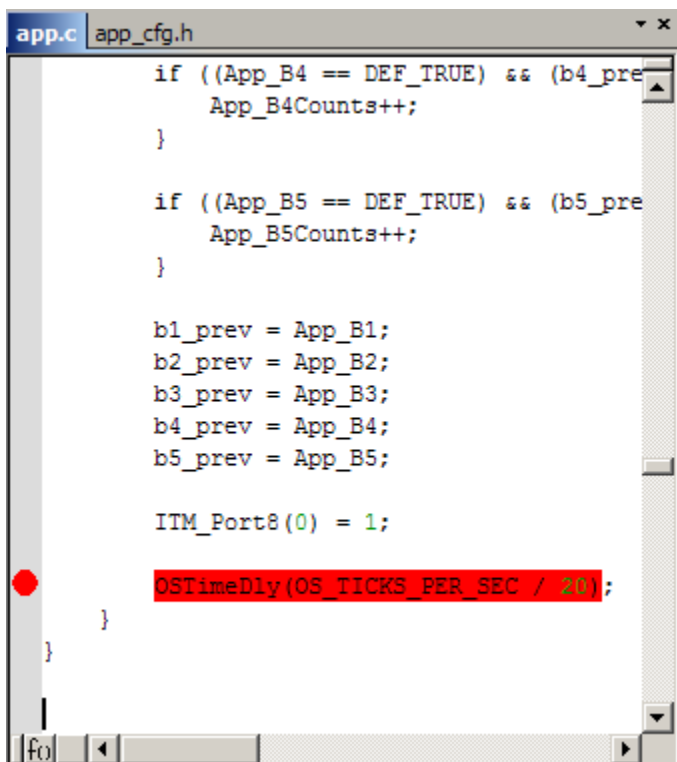


```
app.c app_cfg.h
b1_prev = DEF_FALSE;
b2_prev = DEF_FALSE;
b3_prev = DEF_FALSE;
b4_prev = DEF_FALSE;
b5_prev = DEF_FALSE;

App_B1 = DEF_FALSE;
App_B2 = DEF_FALSE;
App_B3 = DEF_FALSE;
App_B4 = DEF_FALSE;
App_B5 = DEF_FALSE;

while (DEF_TRUE) {
    ITM_Port8(0) = 0;

    App_B1 = BSP_PB_GetStatus(1);
    App_B2 = BSP_PB_GetStatus(2);
    App_B3 = BSP_PB_GetStatus(3);
    App_B4 = BSP_PB_GetStatus(4);
    App_B5 = BSP_PB_GetStatus(5);
```



```
app.c app_cfg.h
    if ((App_B4 == DEF_TRUE) && (b4_pre
        App_B4Counts++;
    }

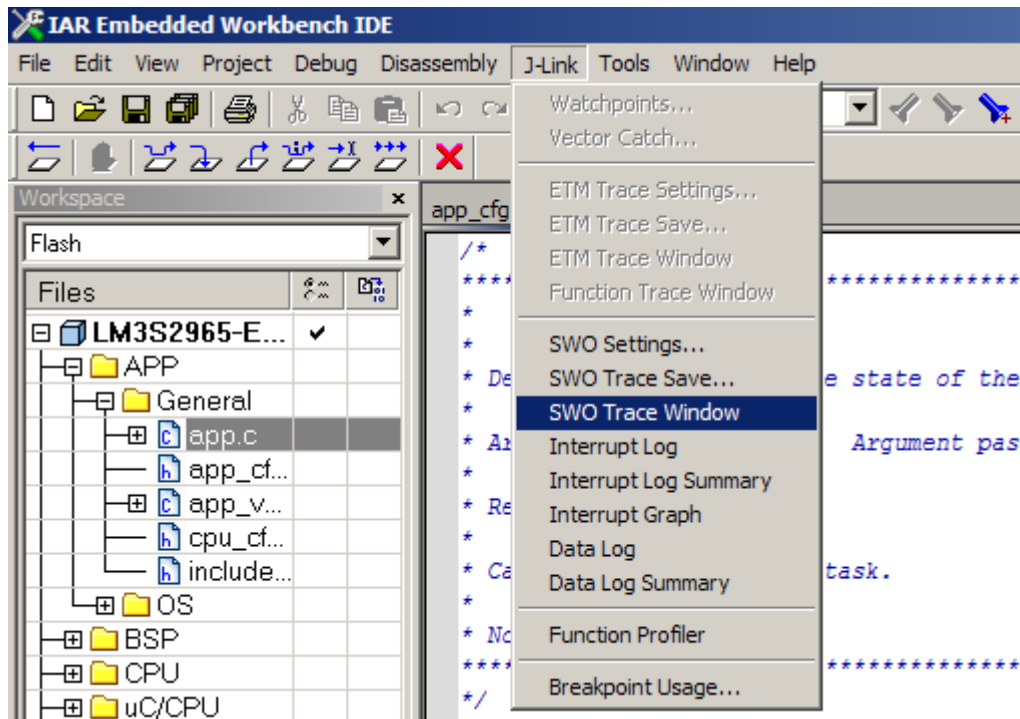
    if ((App_B5 == DEF_TRUE) && (b5_pre
        App_B5Counts++;
    }

    b1_prev = App_B1;
    b2_prev = App_B2;
    b3_prev = App_B3;
    b4_prev = App_B4;
    b5_prev = App_B5;

    ITM_Port8(0) = 1;
    OSTimeDly(OS_TICKS_PER_SEC / 20);
}
}
```

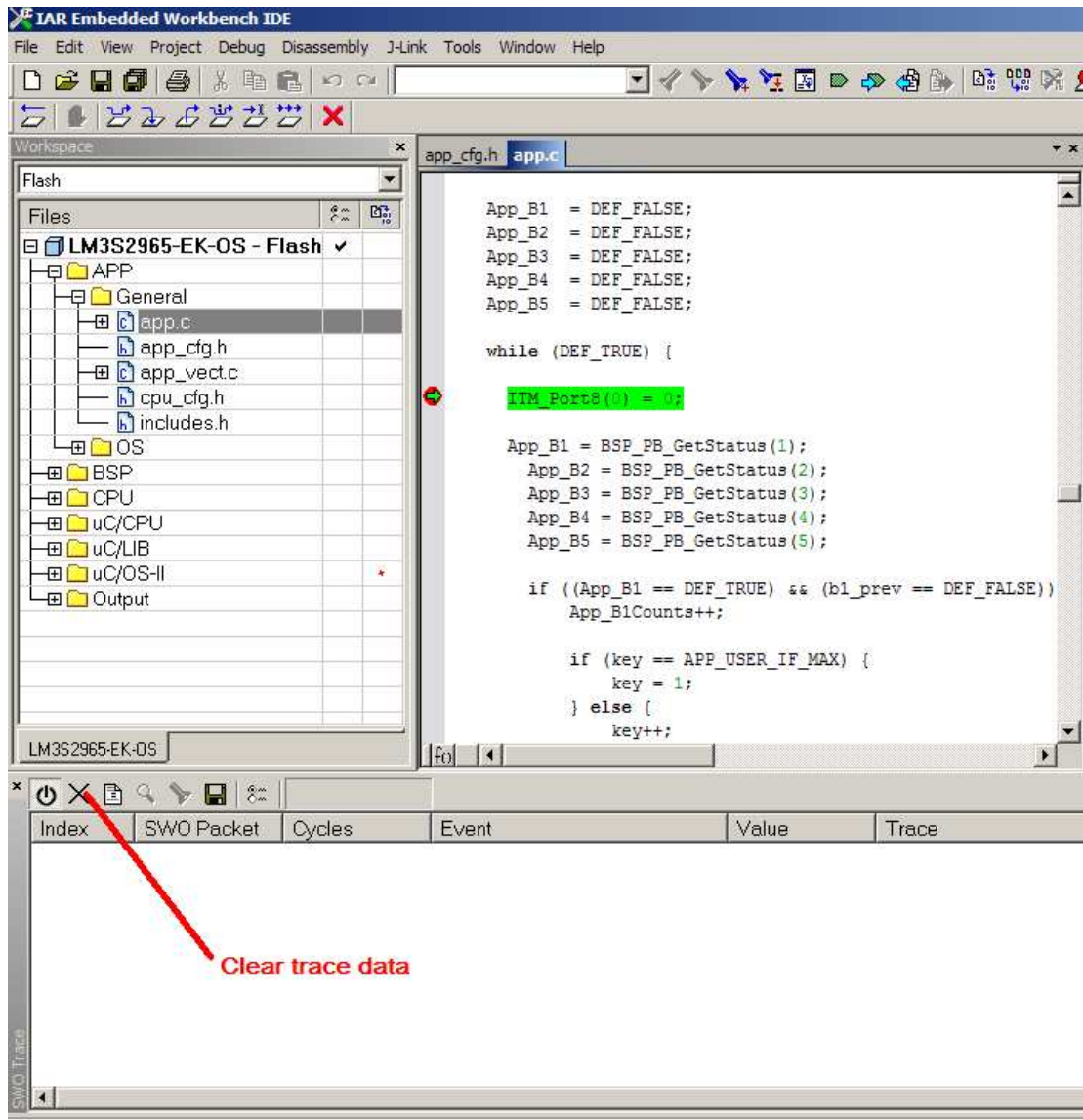
## Measuring the time consumption of a piece of code by Cortex-M3 SWD/SWV

8. Open the “SWO Trace” window:



## Measuring the time consumption of a piece of code by Cortex-M3 SWD/SWV

- Execute your program in full speed until the first breakpoint is reached. Click the “Clear trace data” button to clean the “SWO Trace” window:



## Measuring the time consumption of a piece of code by Cortex-M3 SWD/SWV

10. Execute your program in full speed again, until the second breakpoint is reached:

The screenshot shows the IAR Embedded Workbench IDE. The workspace contains a project named 'LM3S2965-EK-OS - Flash'. The source file 'app.c' is open, showing the following code:

```
if ((App_B5 == DEF_TRUE) && (b5_prev == DEF_FALSE))
    App_B5Counts++;
}

b1_prev = App_B1;
b2_prev = App_B2;
b3_prev = App_B3;
b4_prev = App_B4;
b5_prev = App_B5;

ITM_Port8(0) = 1;
OSTimeDly(OS_TICKS_PER_SEC / 20);
}

/*
*****
*
* Description : Updates LCD.
*
* Argument(s) : p_arg      Argument passed to 'AppTaskUserIF()'
*/
```

The SWO Trace window at the bottom shows the following data:

Index	SWO Packet	Cycles	Event	Value	Trace
000000	C0FF887A	810140964	Timestamp synch	1999999	
000001	C0FF887A	812140963	Timestamp synch	1999999	
000002	C0FF887A	814140962	Timestamp synch	1999999	
000003	C0FF887A	816140961	Timestamp synch	1999999	
000004	C0FF887A	818140960	Timestamp synch	1999999	
000005	0100	818140960	ITM(0)	0x00	
000006	C0AAAF48	819326666	Timestamp synch	1185706	
000007	0101	819326666	ITM(0)	0x01	
000008	C0BD02	819326983	Timestamp synch	317	

11. Find the two ITM(0) events in “SWO Trace” window. The time consumption can be calculated from the difference of the two “Cycles” value, and the main frequency of your target system.