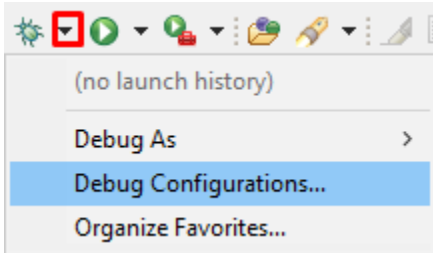


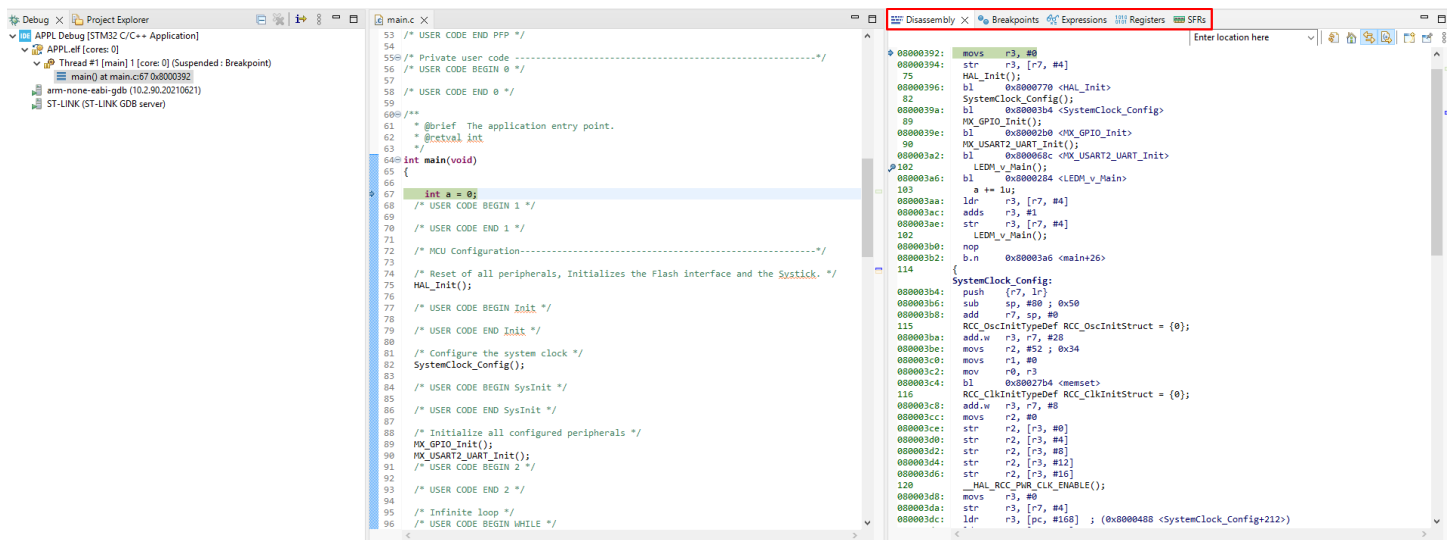
Debug Guide

Setup

In order to debug the code flashed on a device in real time Debug Configuration must be created. The process is EXACTLY the same as the one described in “Flash Guide.pdf” with one single difference:



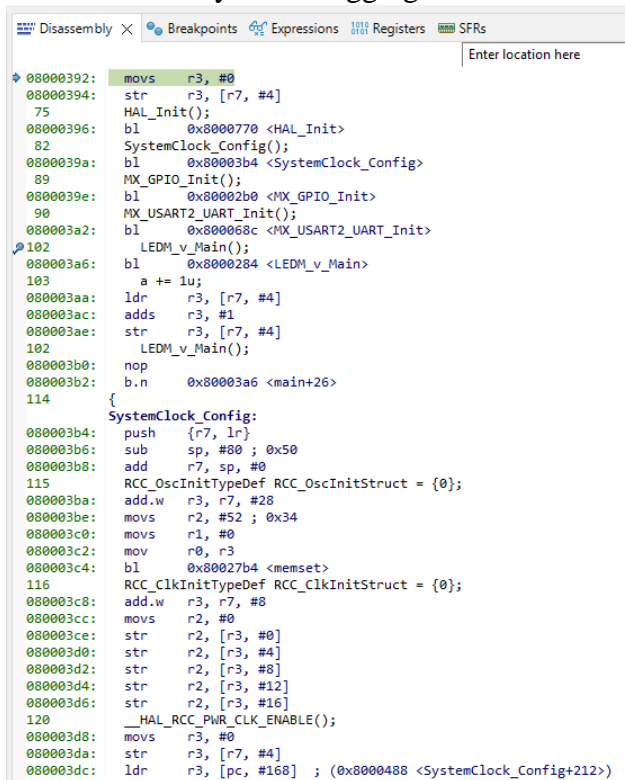
After performing necessary steps described in “Flash Guide.pdf”, new window will prompt asking to change perspective, click on “Switch option”. After this, new arrangement of windows will present itself:



The program will execute and stop at the entry of main function. At this point, the device is halted and ready for debugging. On the picture above, marked with red rectangle, we can see several important views. These can be added simply by selecting them from “View -> Show View”.

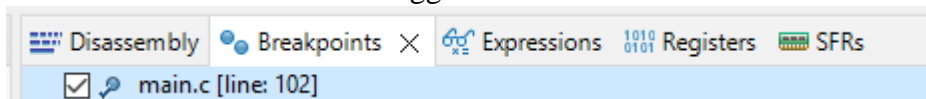
Views

1. Disassembly – this view shows assembly instructions which the device will execute or have executed. It also shows the instruction which will execute next or rather where the device is currently halted. This view is necessary for debugging as it shows how the code is being executed.

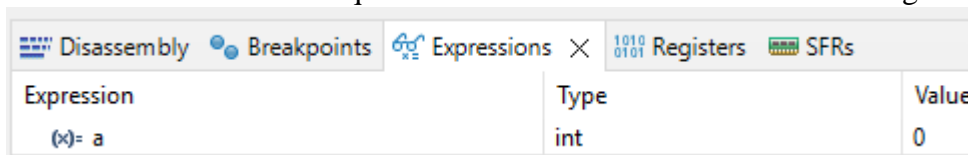


```
08000392: movs r3, #0
08000394: str r3, [r7, #4]
75: HAL_Init();
08000396: bl 0x8000770 <HAL_Init>
82: SystemClock_Config();
0800039a: bl 0x80003b4 <SystemClock_Config>
89: MX_GPIO_Init();
0800039e: bl 0x80002b0 <MX_GPIO_Init>
90: MX_USART2_UART_Init();
080003a2: bl 0x800068c <MX_USART2_UART_Init>
102: LEDM_v_Main();
080003a6: bl 0x8000284 <LEDM_v_Main>
103: a += 1u;
080003aa: ldr r3, [r7, #4]
080003ac: adds r3, #1
080003ae: str r3, [r7, #4]
102: LEDM_v_Main();
080003b0: nop
080003b2: b.n 0x80003a6 <main+26>
114: {
SystemClock_Config:
080003b4: push {r7, lr}
080003b6: sub sp, #80 ; 0x50
080003b8: add r7, sp, #0
115: RCC_OscInitTypeDef RCC_OscInitStruct = {0};
080003ba: add.w r3, r7, #28
080003bc: movs r2, #52 ; 0x34
080003be: movs r1, #0
080003c0: mov r0, r3
080003c2: bl 0x80027b4 <memset>
116: RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
080003c8: add.w r3, r7, #8
080003cc: movs r2, #0
080003ce: str r2, [r3, #0]
080003d0: str r2, [r3, #4]
080003d2: str r2, [r3, #8]
080003d4: str r2, [r3, #12]
080003d6: str r2, [r3, #16]
120: HAL_RCC_PWR_CLK_ENABLE();
080003da: movs r3, #0
080003dc: str r3, [r7, #4]
080003de: ldr r3, [pc, #168] ; (0x8000488 <SystemClock_Config+212>)
```

2. Breakpoints – this view shows breakpoint list: a list of instructions where the device should halt in case they are meant to be executed. They are useful for debugging in order to stop and slowly execute a part of code which needs to be debugged.

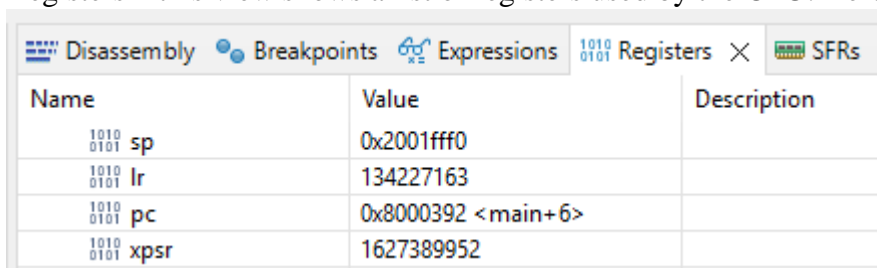


3. Expressions – this view shows a list of variables chosen by the developer and their values in real time. Useful to have listed for a quick look of the variables value for a debug.



Expression	Type	Value
(x)= a	int	0

4. Registers – this view shows a list of registers used by the CPU. Here can be seen PC, SP and so on.



Name	Value	Description
sp	0x2001fff0	
lr	134227163	
pc	0x8000392 <main+6>	
xpsr	1627389952	

5. SFRs – this view shows a list of Special Function Registers which are related to the specific chip. Here can be found all sort of register groups like MPU, SysTick and so on.

Actions

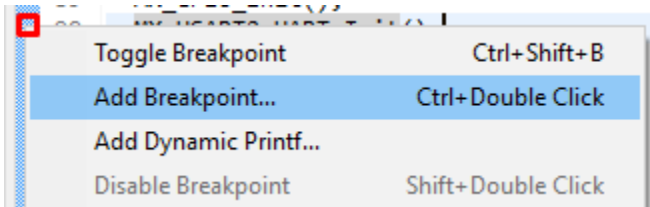
All necessary actions can be found here:



1. Reset – resets the device
2. Disable Breakpoint – disables all breakpoints
3. Terminate and Relaunch – terminates this debug session and relaunches it
4. Resume – breaks a device from a halt
5. Suspend – stops a device whilst running
6. Terminate – terminates this debug session
7. Disconnect – disconnects the device and terminates this debug session
8. Step into – performs the following instruction, group of instructions or function call
9. Step over – performs the following instruction, group of instructions or function call and moves to the next
10. Step return – not possible
11. Instruction stepping mode – switches the mode from a group of instructions to a single instruction and vice versa. This changes the step “size”.

Creating a breakpoint

On the left side of a line number in a source code pressing the right click and choosing “Add a Breakpoint...” option prompts a window for breakpoint options. After configuring press “Apply and Close”. New breakpoint can be found in Breakpoints view.



Adding a variable to Expressions view

Clicking on a variable in source code with a right click and selecting “Add Watch Expression...” will add a variable to Expressions view.

