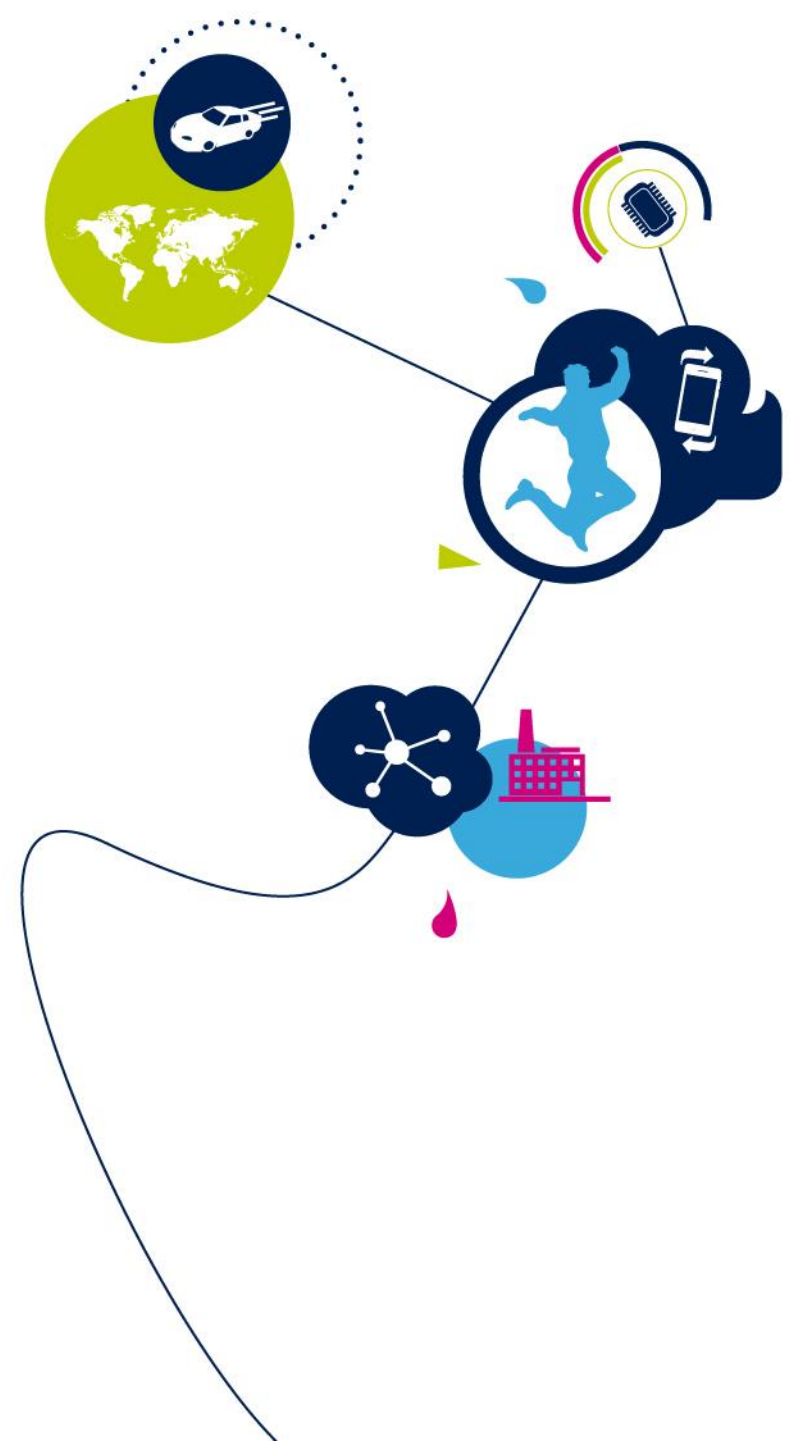


TrueSTUDIO for STM32

April 2018



- STM32전용 무료 개발환경 TrueSTUDIO v9.0.0 특징 및 기본구조
- STM32CubeMX를 활용한 TrueSTUDIO 프로젝트 생성/빌드
- TrueSTUDIO의 다양한 Build & Debugging 분석 기능

TrueSTUDIO for STM32 v9.0.0

- TrueSTUDIO for STM32 = Lite ver. + Pro ver.



TrueSTUDIO STM32
~~TrueSTUDIO Pro~~
~~TrueSTUDIO Lite~~

ALL FEATURES FREE FOR STM32 DEVELOPERS

The screenshot displays the TrueSTUDIO IDE interface for ARM® Pro. The main window shows a C++ code editor with the following code:

```
154 byte1++;  
157 byte2++;  
158  
159 i++;  
160 if (i > 10) {  
161     i++;  
162 }  
163  
164 if (j > 0x7fffffff) {  
165     j++;  
166 }
```

The SWV Console shows a table of SWV Trace Log data:

Index	Type	Data	Cycles	Time(s)	Extra info
22729	Data value	98	1426738175	89.1711359...	Timestamp...
22730	Data value	3209650226	1428337843	89.2711151...	
22731	Data value	98	1428337846	89.2711153...	
22732	Data value	98	142833816...	7	No timesta...
22733	Data value	99	1428338164	89.2711352...	Timestamp...

The Register window shows the SCB register set:

Register	Address	Value
SYST_CSR	0xe00e0100	0x10007
SYST_RVR	0xe00e0114	0x270ff
RELOAD	[0:24]	0x270ff
RESERVED	[24:8]	0x0
SYST_CVR	0xe00e0118	0x25479
CURRENT	[0:24]	0x25479
RESERVED	[24:8]	0x0
SYST_CALIB	0xe00e011c	0x0
TENMS	[0:24]	0x0
RESERVED	[24:6]	0x0
SKEW	[30:1]	0x0

The SWV Data Trace window shows a table of SWV Data Trace data:

Comp	Name	Value
0	signal1	1
1	signal2	99
2	signal3	-0.8100616

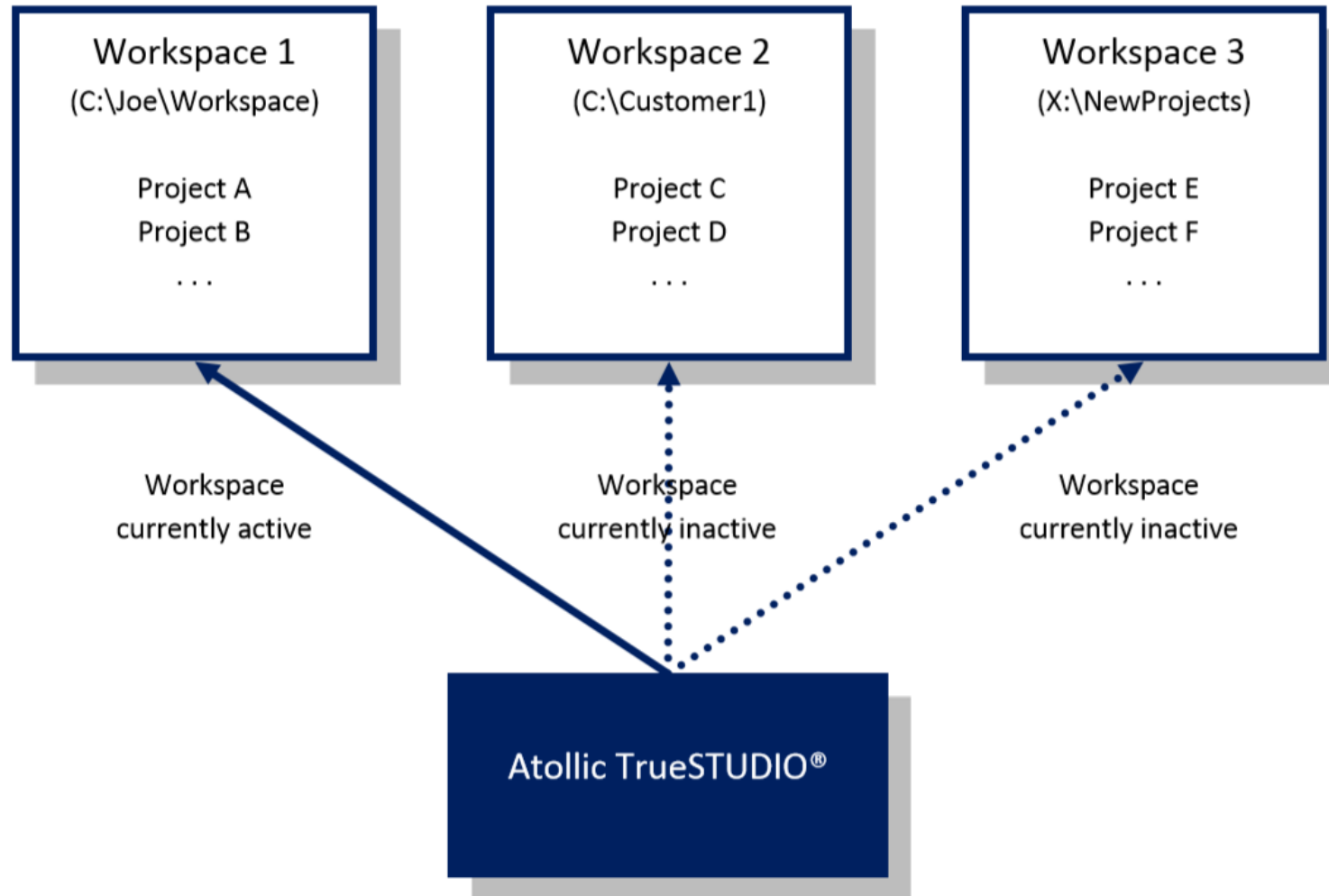
The SWV ITM Timeline Graph shows a graph of SWV Exception Timeline Graph with a red signal. The SWV Trace Timeline Graph shows a graph of SWV Trace Timeline Graph with a yellow signal.

- Windows / Linux Installer 제공
- STM32 전용 개발환경
- Editor : 코드 탐색, 시각화, 커스터마이징 등 사용 편의제공
- Build / Memory Analyzer : Flash, SRAM, STACK 사용량 분석
- Hard Fault Analysis : Crash 발생 원인 및 위치 분석
- Trace & Profiling : SWV 기능을 이용한 실시간 Event / Data / Exception Trace
- Open Standards : Eclipse™ 기반

- Atollic TrueSTUDIO = Eclipse™ 기반 개발환경
- 비영리 재단, 오픈소스 소프트웨어
- C/C++, JAVA, PHP 등을 위한 개발환경으로 널리사용



- Workspaces & Projects 모델

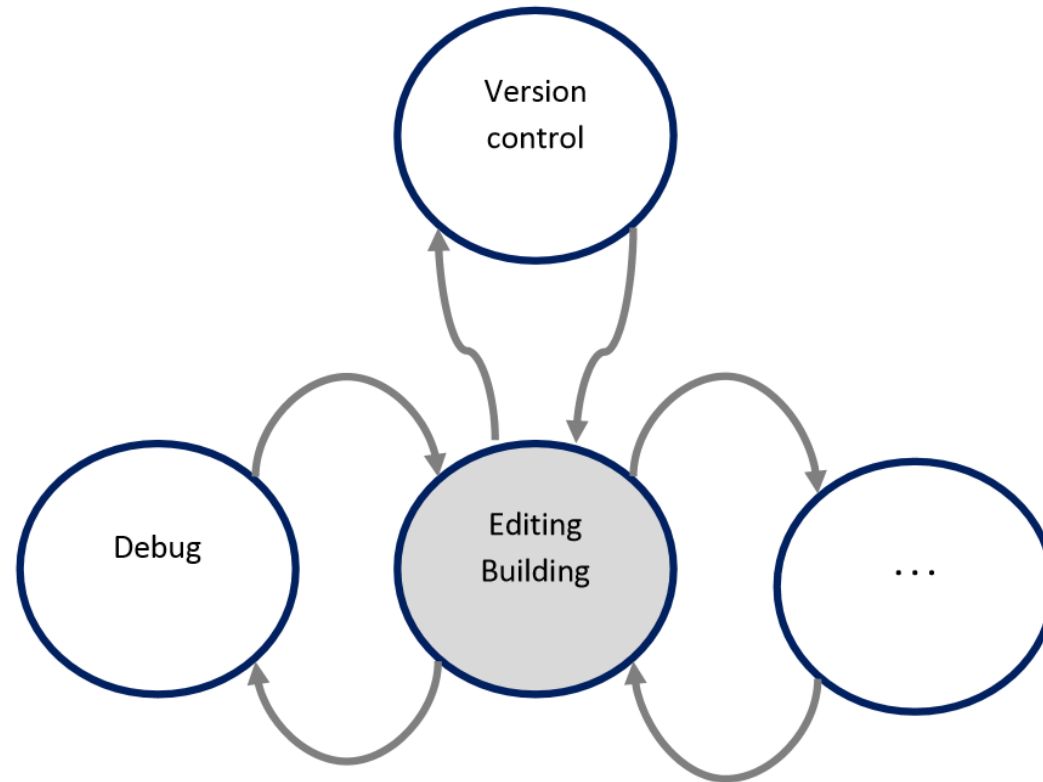


Eclipse™ 기본 구조(2/4)

- Workspaces & Projects 모델

The screenshot displays the Atollic TrueSTUDIO IDE interface. On the left, the Project Explorer shows a tree view of a project named 'myProject', with subfolders like 'Binaries', 'Includes', 'Libraries', 'Utilities', and 'src'. The 'src' folder is expanded, showing files such as 'conf.h', 'it.c', 'it.h', 'main.c', 'startup.s', 'syscalls.c', 'system.c', 'tiny_printf.c', 'Debug', 'unittest', 'linker_script.ld', and 'myProject.elf.launch'. A blue arrow points from the word 'Project' in the Project Explorer to the 'main.c' file. The main editor window shows the source code of 'main.c', which includes comments and function prototypes, followed by a 'main' function that initializes LEDs and enters an infinite loop. A blue arrow points from the text 'Workspace 변경 (재시작)' to the 'Switch Workspace' option in the File menu. The File menu is open, showing options like 'New', 'Open File...', 'Save', 'Print...', 'Switch Workspace', 'Restart', 'Import...', and 'Export...'. The 'Switch Workspace' option is highlighted, and a submenu is visible showing two workspace locations: 'C:\Users\john lee\Desktop\Cube\ts_workspace' and 'C:\Users\john lee\STM32Cube\Repository'. The word 'Workspace' is written in pink at the bottom left of the screenshot area. The Atollic logo and 'life.augmented' text are in the bottom left corner.

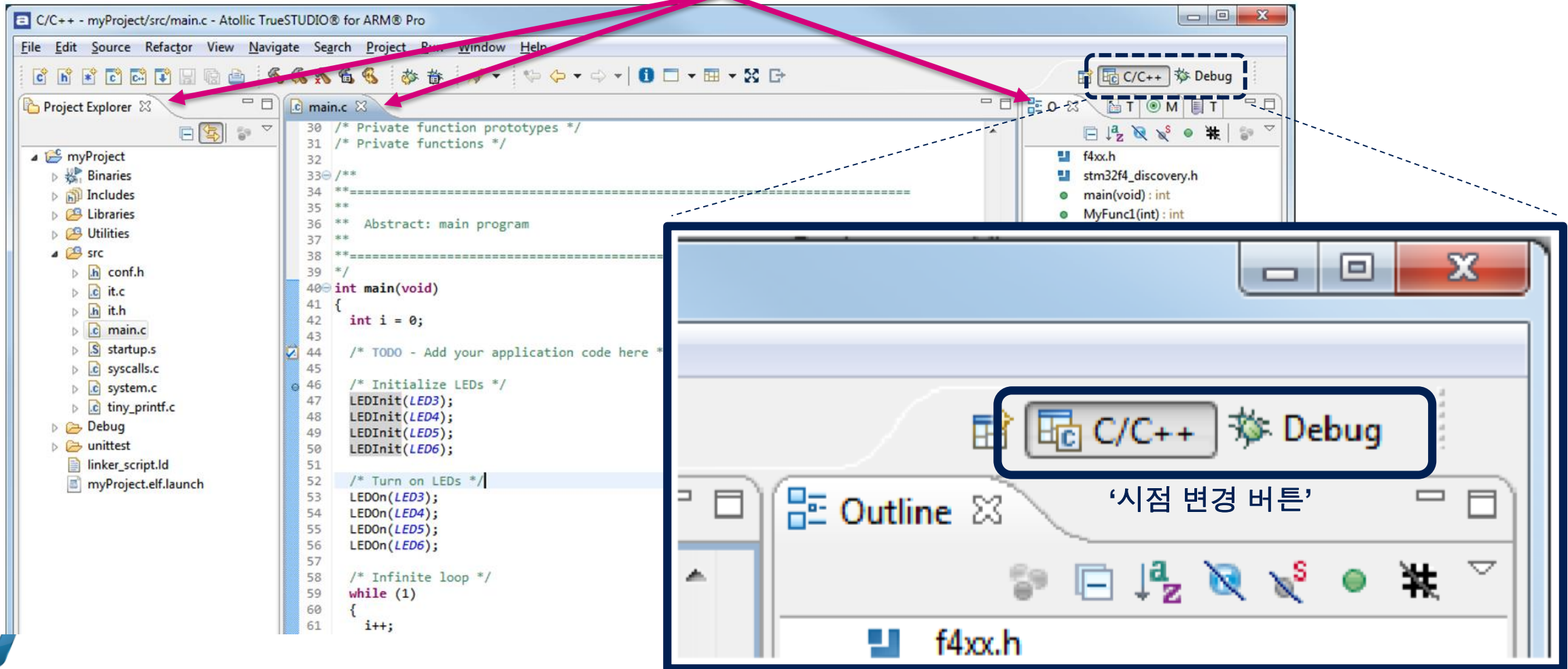
- Perspectives & Views



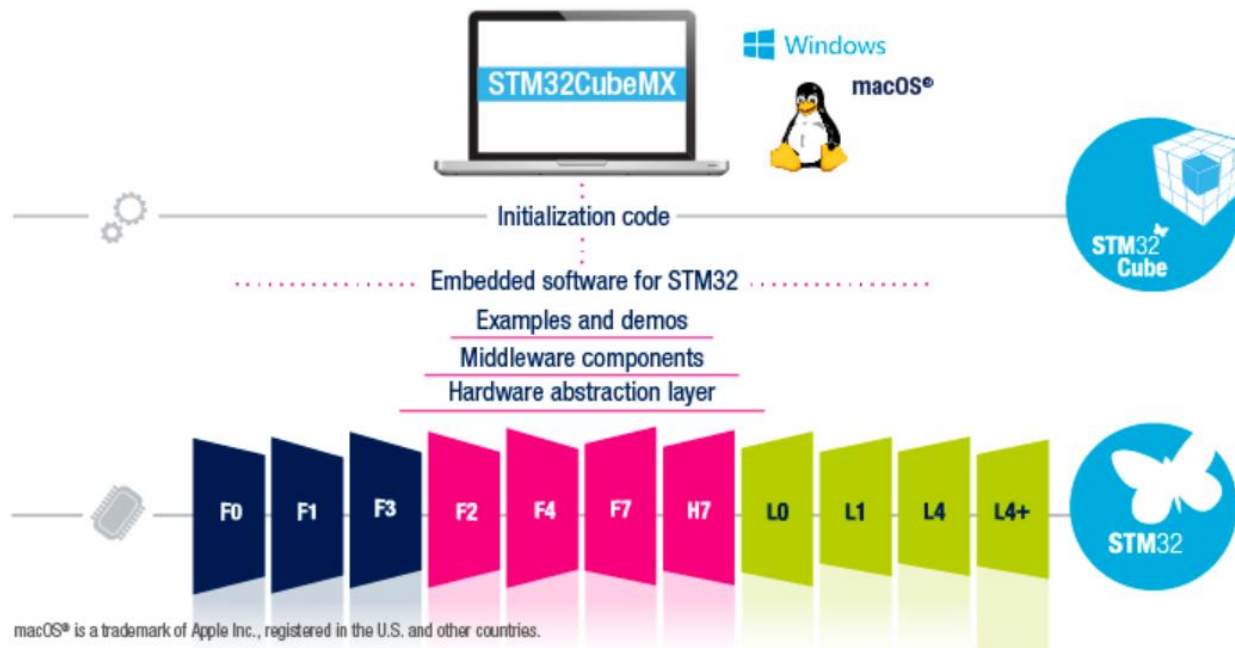
Eclipse™ 기본 구조(4/4)

- Perspectives & Views

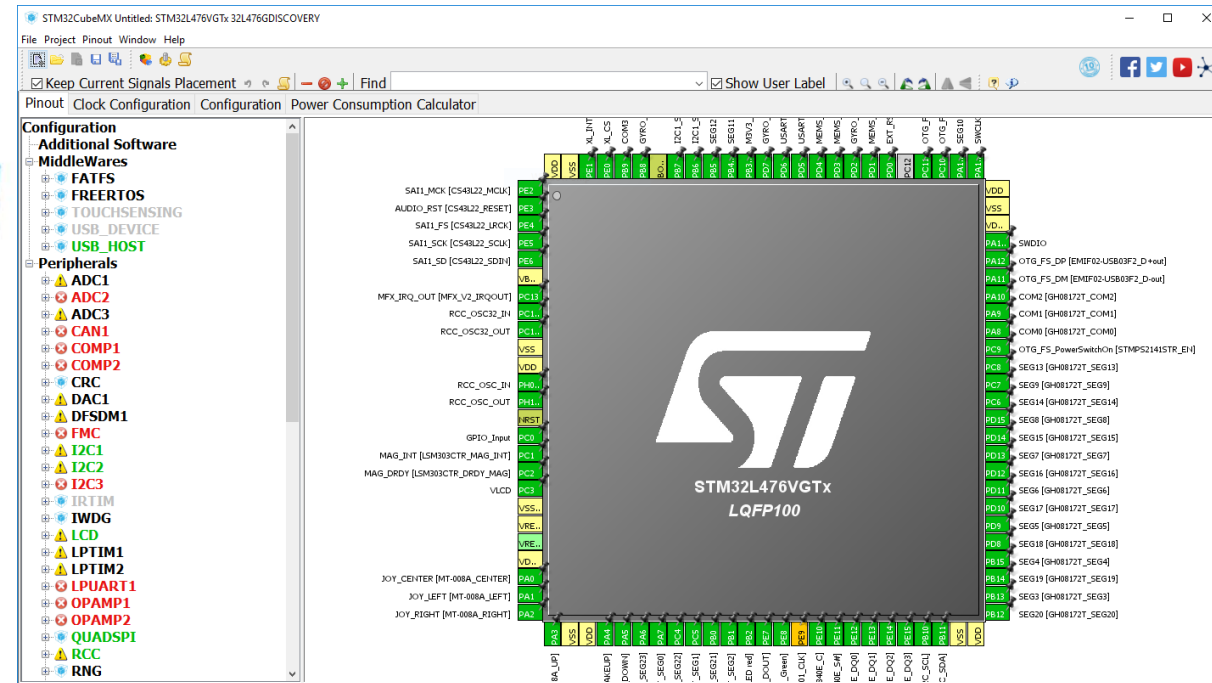
Docking views



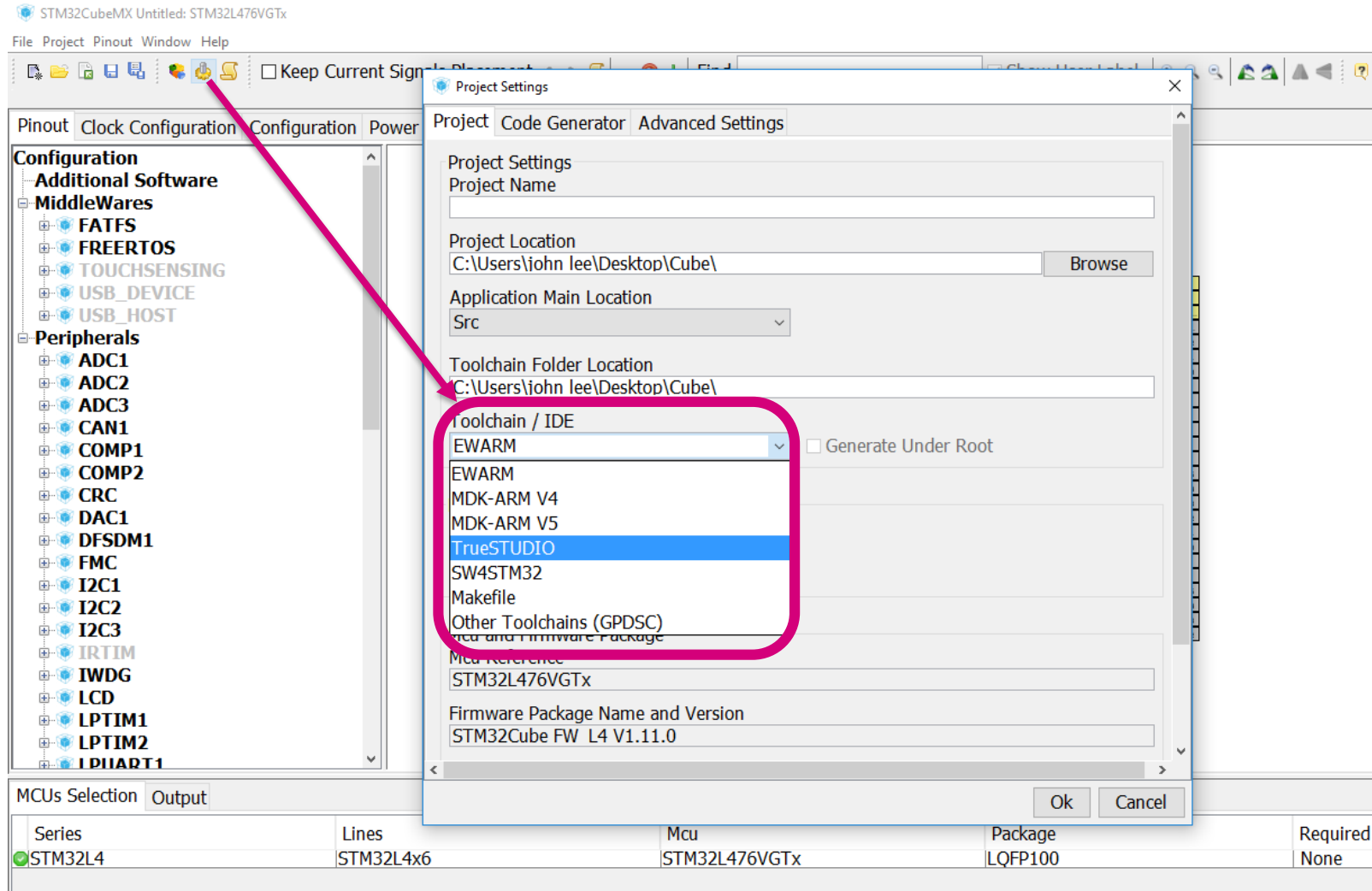
- STM32를 위한 초기화 코드 생성 툴
 - 그래픽 인터페이스로 편리한 설정 : Pinout, Clock, Peripherals...



macOS® is a trademark of Apple Inc., registered in the U.S. and other countries.

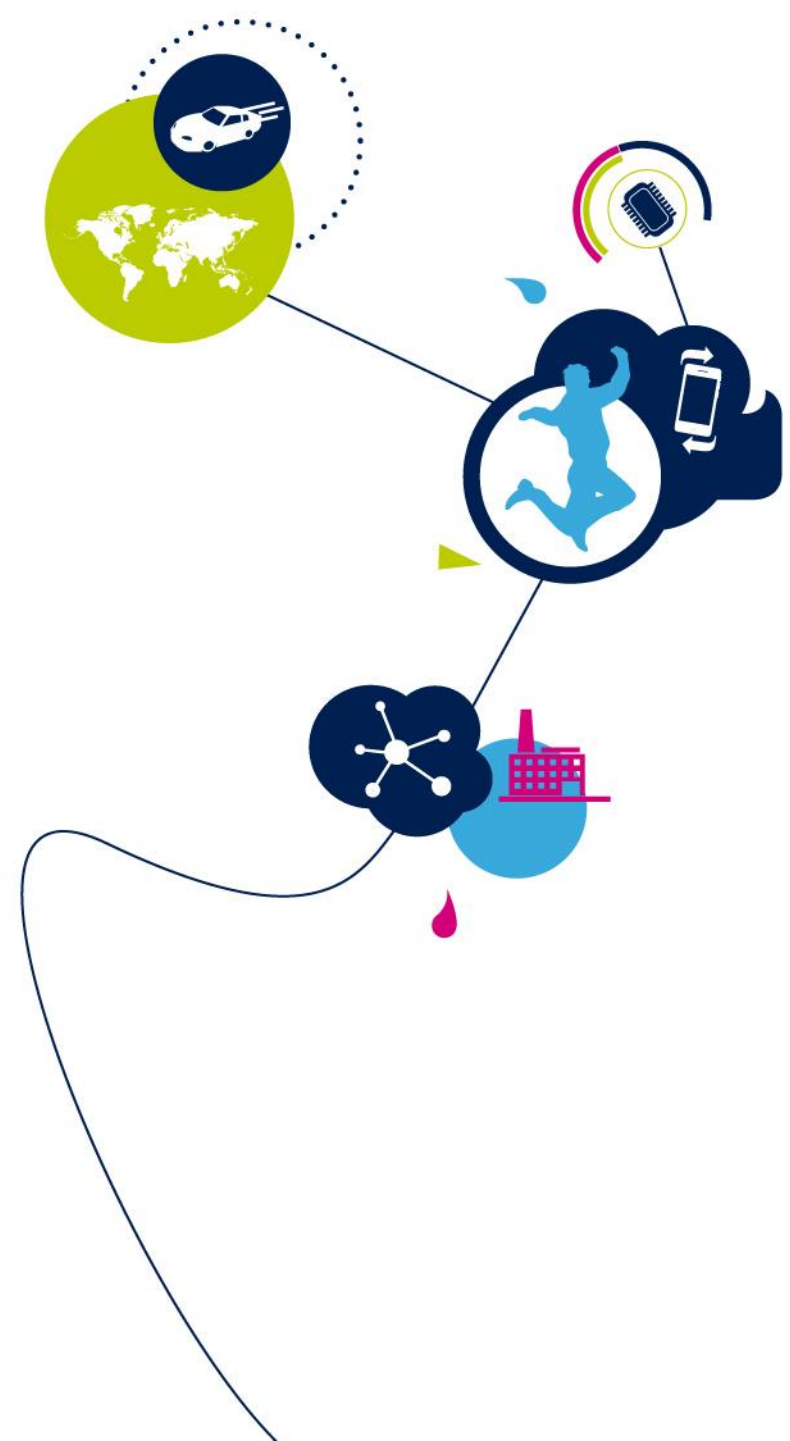


- 지원하는 개발환경 : Atollic(TS), IAR(EWARM), Keil(MDK-ARM) ...

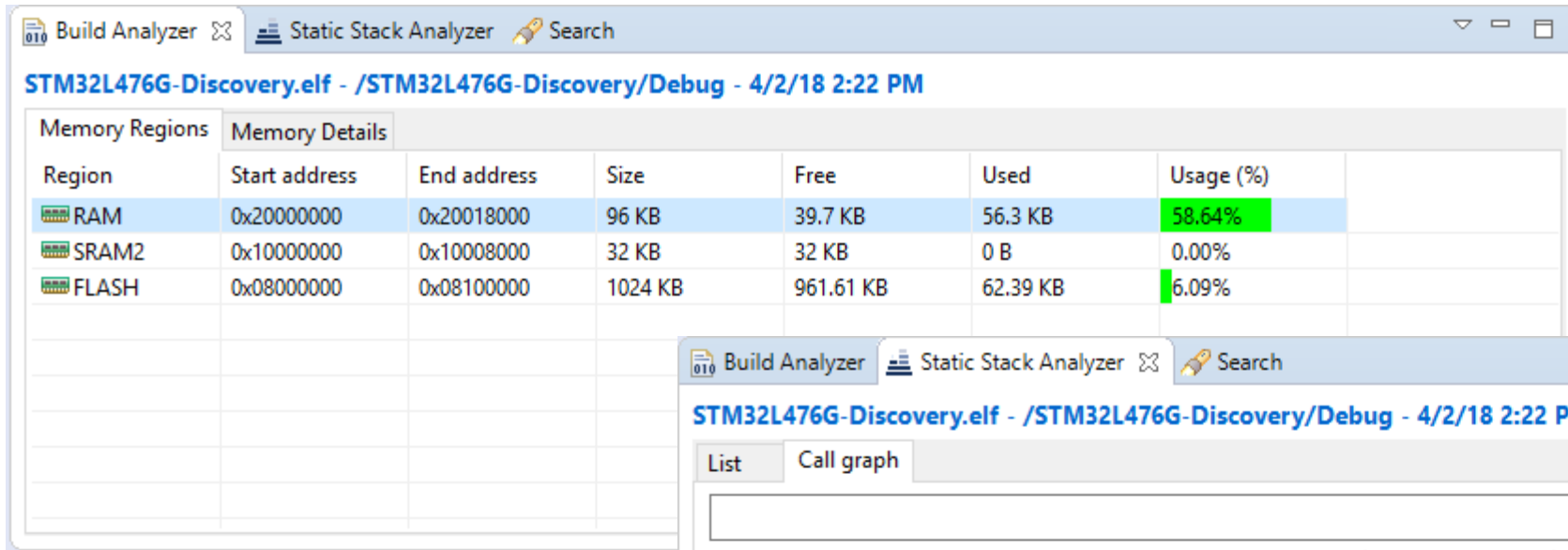


- CubeMX UI를 이용한 프로젝트 생성 시연
- TrueSTUDIO 가져오기 / 빌드 시연

Build & Debug features



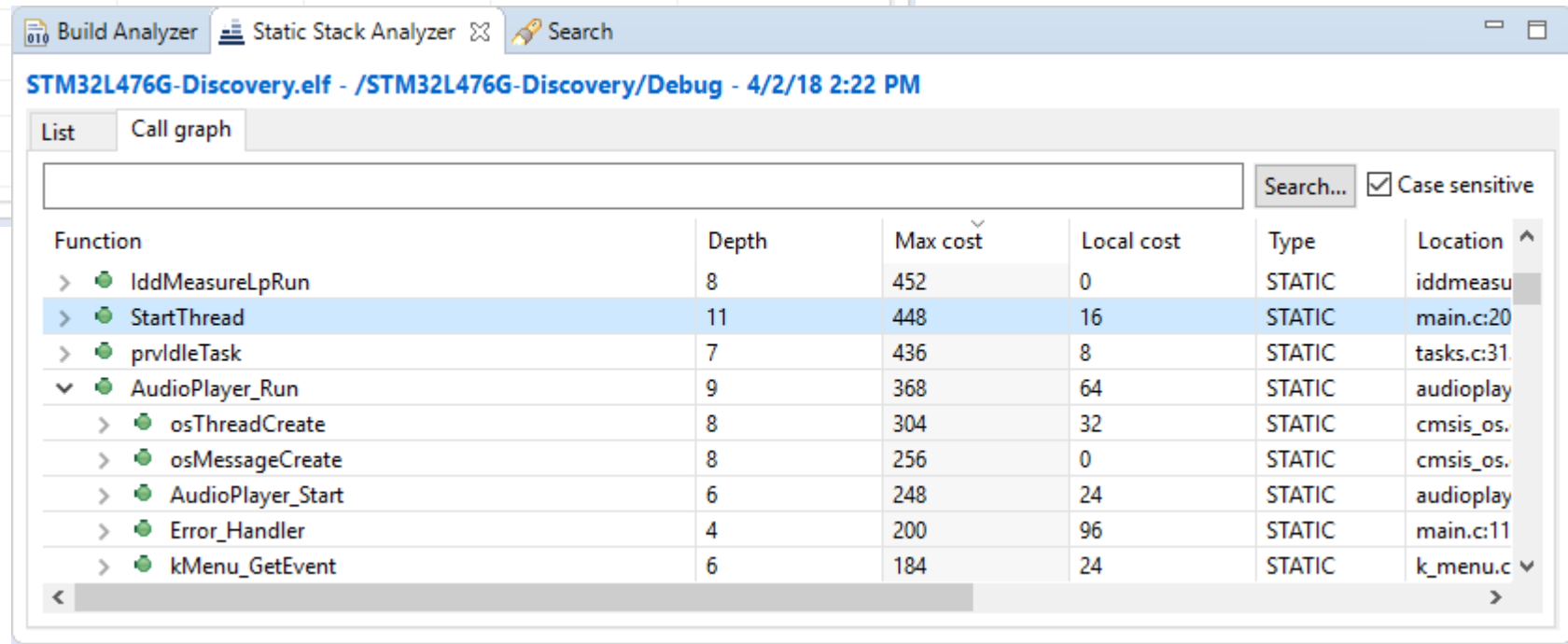
- Memory 및 STACK 사용량 분석



Build Analyzer Static Stack Analyzer Search

STM32L476G-Discovery.elf - /STM32L476G-Discovery/Debug - 4/2/18 2:22 PM

Region	Start address	End address	Size	Free	Used	Usage (%)
RAM	0x20000000	0x20018000	96 KB	39.7 KB	56.3 KB	58.64%
SRAM2	0x10000000	0x10008000	32 KB	32 KB	0 B	0.00%
FLASH	0x08000000	0x08100000	1024 KB	961.61 KB	62.39 KB	6.09%



Build Analyzer Static Stack Analyzer Search

STM32L476G-Discovery.elf - /STM32L476G-Discovery/Debug - 4/2/18 2:22 PM

List Call graph

Search... Case sensitive

Function	Depth	Max cost	Local cost	Type	Location
> IddMeasureLpRun	8	452	0	STATIC	iddmeasu
> StartThread	11	448	16	STATIC	main.c:20
> prvIdleTask	7	436	8	STATIC	tasks.c:31
▼ AudioPlayer_Run	9	368	64	STATIC	audioplay
> osThreadCreate	8	304	32	STATIC	cmsis_os.
> osMessageCreate	8	256	0	STATIC	cmsis_os.
> AudioPlayer_Start	6	248	24	STATIC	audioplay
> Error_Handler	4	200	96	STATIC	main.c:11
> kMenu_GetEvent	6	184	24	STATIC	k_menu.c

CPU fault 분석기

```
...
// Average reading.
accel[zAxis] = (accel[zAxis]*AVG+s8buff[zAxis]) / (AVG + 1);
accel[yAxis] = (accel[yAxis]*AVG+s8buff[yAxis]) / (AVG + 1);
accel[xAxis] = (accel[xAxis]*AVG+s8buff[xAxis]) / (AVG + 1);
// Update the LED every 45ms.
if ( ++i == 3 ) {
    TIM_SetDuty( TIM4, ORG_LED, 0 );
} else {
    if ( accel[xAxis] > 0 ) {
        TIM_SetDuty( TIM4, ORG_LED, ptopwm( accel[xAxis] * MEMS_MULT) );
    } else {
        TIM_SetDuty( TIM4, BLU_LED, ptopwm( accel[xAxis] * -MEMS_MULT) );
        TIM_SetDuty( TIM4, ORG_LED, 0 );
    }
}

if ( accel[yAxis] == 0 ) {
    TIM_SetDuty( TIM4, GRN_LED, RED_LED );
} else {
    if ( accel[yAxis] > 0 ) {
        TIM_SetDuty( TIM4, GRN_LED, ptopwm( accel[yAxis] * MEMS_MULT) );
        TIM_SetDuty( TIM4, RED_LED, 0 );
    } else {
        TIM_SetDuty( TIM4, RED_LED, ptopwm( accel[yAxis] * -MEMS_MULT) );
        TIM_SetDuty( TIM4, GRN_LED, 0 );
    }
}

// Send debug string every 180ms.
if ( fMems_Debug && (++j) >= 24 ) { //A] change &&
    j = 0;
    memsXaxis = accel[xAxis];
    memsYaxis = accel[yAxis];
    memsZaxis = accel[zAxis];

    printf( "Accel X=%i Y=%i Z=%i\n", accel[xAxis], accel[yAxis], accel[zAxis] );
}
}
```



Fault Analyzer

Hard Fault Detected

Hard Fault Details

- Triggered by bus, memory management or usage fault (FORCED)
- Indicate hard fault is caused by failed vector fetch (VECTBL)
- Indicate hard fault is triggered by debug event (DEBUGEVT)

Bus Fault Details

Usage Fault Details

- Attempt to execute an undefined instruction (UNDEFINSTR)
- Attempt to switch to invalid state (INVSTATE)
- Attempt to do exception with bad value in EXEC_RETURN number (INVPC)
- Attempt to execute a coprocessor instruction (NOCP)
- Indicates that an unaligned access fault has taken place (UNALIGNED)
- Indicates a divide by zero has taken place (DIVBYZERO)

Memory Management Fault Details

Register Content During Fault Exception

Name	Value
00000000 sp (MSP)	0x2001ffc0
00000000 r0	0x20000000
00000000 r1	0x1
00000000 r2	0x0
00000000 r3	0x20000000
00000000 r12	0x0
00000000 lr	0x8000509
00000000 pc	0x80004ce
00000000 xpsr	0x61000000

The value of the stack pointer when the fault occurred. Please verify that this value points to a valid stack memory region.

MSP = Main Stack Pointer
PSP = Process Stack Pointer

```
LIS302DL_Read( (uint8_t*)s8buff, LIS302DL_OUT_X_ADDR, 5 );

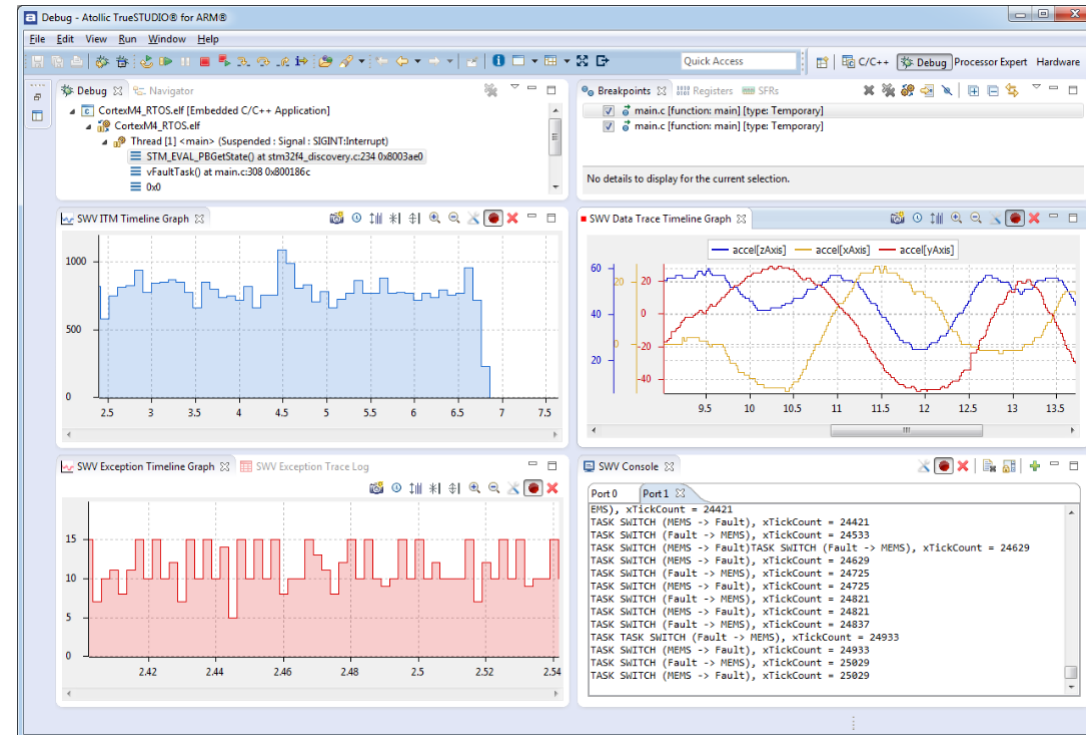
// Average reading.
accel[zAxis] = (accel[zAxis]*AVG+s8buff[zAxis]) / (AVG + 1);
accel[yAxis] = (accel[yAxis]*AVG+s8buff[yAxis]) / (AVG + 1);
accel[xAxis] = (accel[xAxis]*AVG+s8buff[xAxis]) / (AVG + 1);
// Update the LED every 45ms.

Disassembly
Enter location here
0800158a: adds r2, r3, r2
0800158c: movw r3, #6968 ; 0x1b38
08001590: movt r3, #8192 ; 0x2000
08001594: ldrb r3, [r3, #4]
08001596: sxtb r3, r3
08001598: adds r2, r2, r3
0800159a: mov.w r3, #0
0800159e: sdiv r3, r2, r3
080015a2: uxtb r2, r3
080015a4: movw r3, #6960 ; 0x1b30
080015a8: movt r3, #8192 ; 0x2000
```

- Build & Memory 분석기능 + CPU fault 분석기능 시연

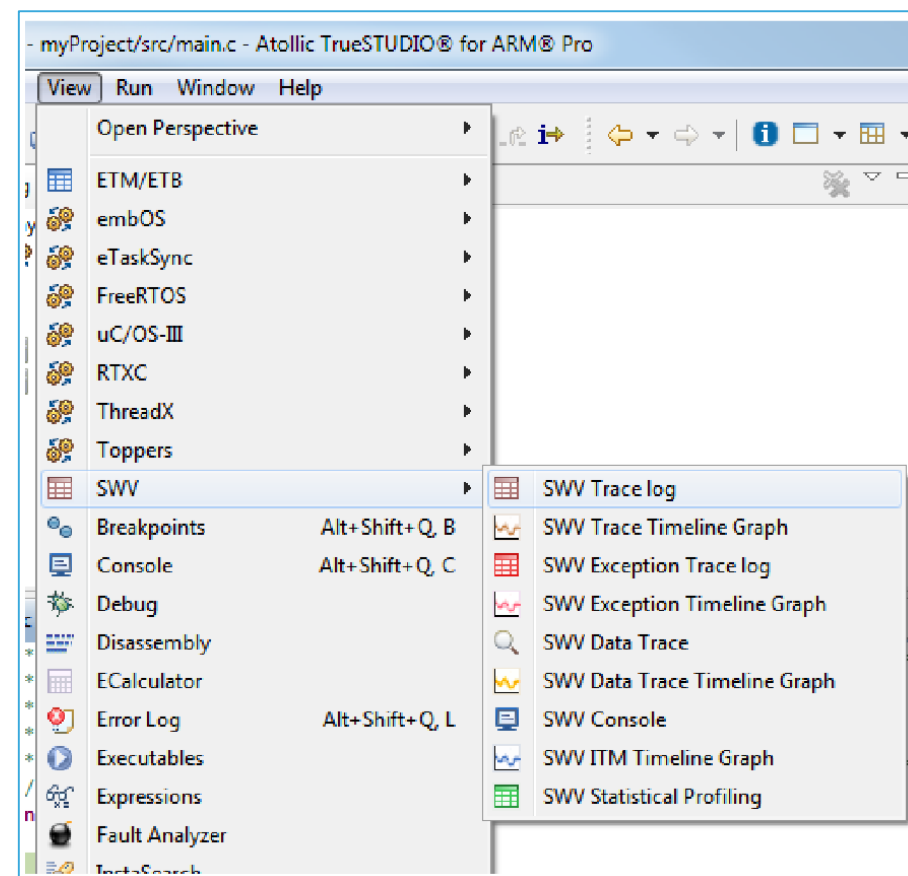
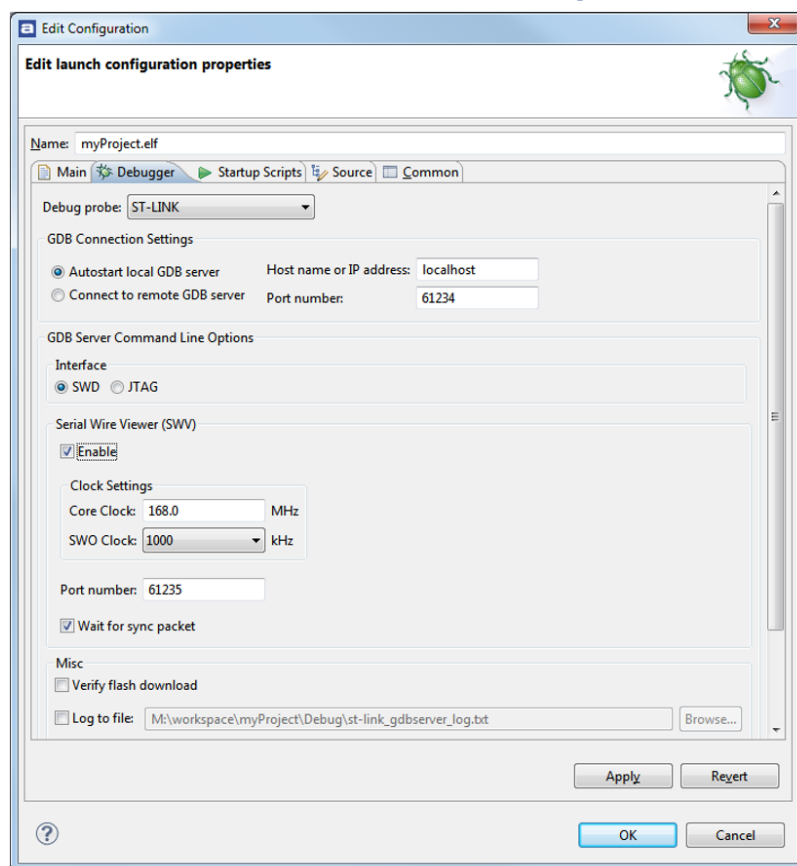
SWV(Serial Wire Viewer)

- SWV : SWD+SWO를 이용하는 Real-time Tracing 기술
 - SWD : 2-pin, JTAG보다 적은 핀으로 같은 디버깅 기능 제공(run, stop, break, step)
 - SWO : 1-pin, Real-time 데이터 출력
- SWV가 제공하는 정보
 - Program counter 샘플링(CPU 점유율 분석)
 - Memory read/write 분석
 - Interrupt(exception) entry/exit 분석
 - Timestamp and CPU cycle 정보



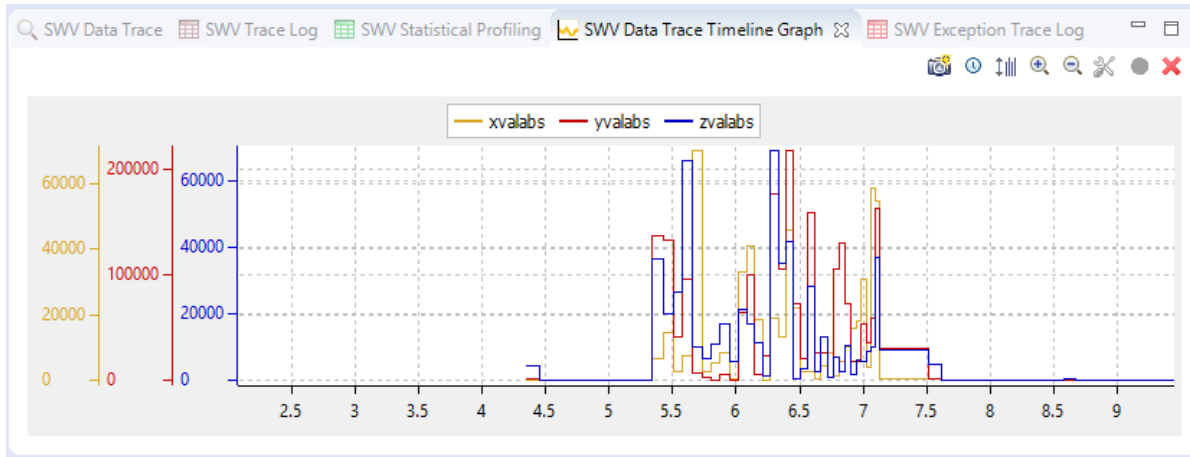
SWV(Serial Wire Viewer)

- SWV 설정 / 뷰어열기
 - 설정 : 툴바에서 'Configure Debug' 버튼 클릭
 - 뷰어 : View → SWV → SWV Trace log



SWV(Serial Wire Viewer)

- SWV viewers



Trace시작

The figure shows the 'SWV Statistical Profiling' window. It contains a table with the following columns: Function, % in use, Samples, Start address, and Size. The table lists various functions and their usage percentages. A red arrow points from the 'Trace시작' text to the 'Start' button in the window's toolbar. Another red arrow points from the 'SWV설정' text to the 'Settings' icon in the toolbar.

Function	% in use	Samples	Start address	Size
HAL_GetTick()	48.90%	49138	0x80032e1	0xc
HAL_Delay()	47.46%	47696	0x80032ed	0x1c
HAL_LCD_UpdateDisplay...	3.58%	3600	0x8004541	0x48
HAL_LCD_Write()	0.01%	15	0x80044c1	0x80
WriteChar.isra.3()	0.01%	15	0x800209d	0x530
SPIx_WriteRead()	0.01%	9	0x8000b5d	0x3c
L3GD20_ReadXYZAngRat...	0.01%	7	0x80006d9	0xd8
BSP_LCD_GLASS_Display...	0.00%	5	0x8002849	0x2c
Gyro_demo()	0.00%	4	0x8006d49	0x10c
BSP_LCD_GLASS_Scrolls...	0.00%	2	0x8002a2d	0xd0

Overflow packets: 497 PC Samples: 100492

SWV설정

The figure shows the 'SWV Exception Trace Log' window. It contains a table with the following columns: Index, Type, Name, Peripheral, Function, Cycles, Time(s), and Extra info. The table lists various exception events and their details. The 'Extra info' column contains the text 'Timestamp delayed. Packet delayed.' for several entries.

Index	Type	Name	Peripheral	Function	Cycles	Time(s)	Extra info
2849	Exception exit	SYSTICK (EXC 15)	SysTick	SysTick_Handler()	68876568	956.619 ms	
2850	Exception return	N/A (EXC 0)			68879113	956.654347222222 ms	Timestamp delayed. Packet delayed.
2851	Exception entry	SYSTICK (EXC 15)	SysTick	SysTick_Handler()	68948284	957.615055555556 ms	
2852	Exception exit	SYSTICK (EXC 15)	SysTick	SysTick_Handler()	68948565	957.618958333334 ms	
2853	Exception return	N/A (EXC 0)			68951110	957.654305555555 ms	Timestamp delayed. Packet delayed.
2854	Exception entry	SYSTICK (EXC 15)	SysTick	SysTick_Handler()	69020281	958.615013888889 ms	
2855	Exception exit	SYSTICK (EXC 15)	SysTick	SysTick_Handler()	69020562	958.618916666667 ms	
2856	Exception return	N/A (EXC 0)			69023107	958.654263888889 ms	Timestamp delayed. Packet delayed.
2857	Exception entry	SYSTICK (EXC 15)	SysTick	SysTick_Handler()	69092278	959.614972222223 ms	
2858	Exception exit	SYSTICK (EXC 15)	SysTick	SysTick_Handler()	69092559	959.618875 ms	
2859	Exception return	N/A (EXC 0)			69095104	959.654222222222 ms	Timestamp delayed. Packet delayed.

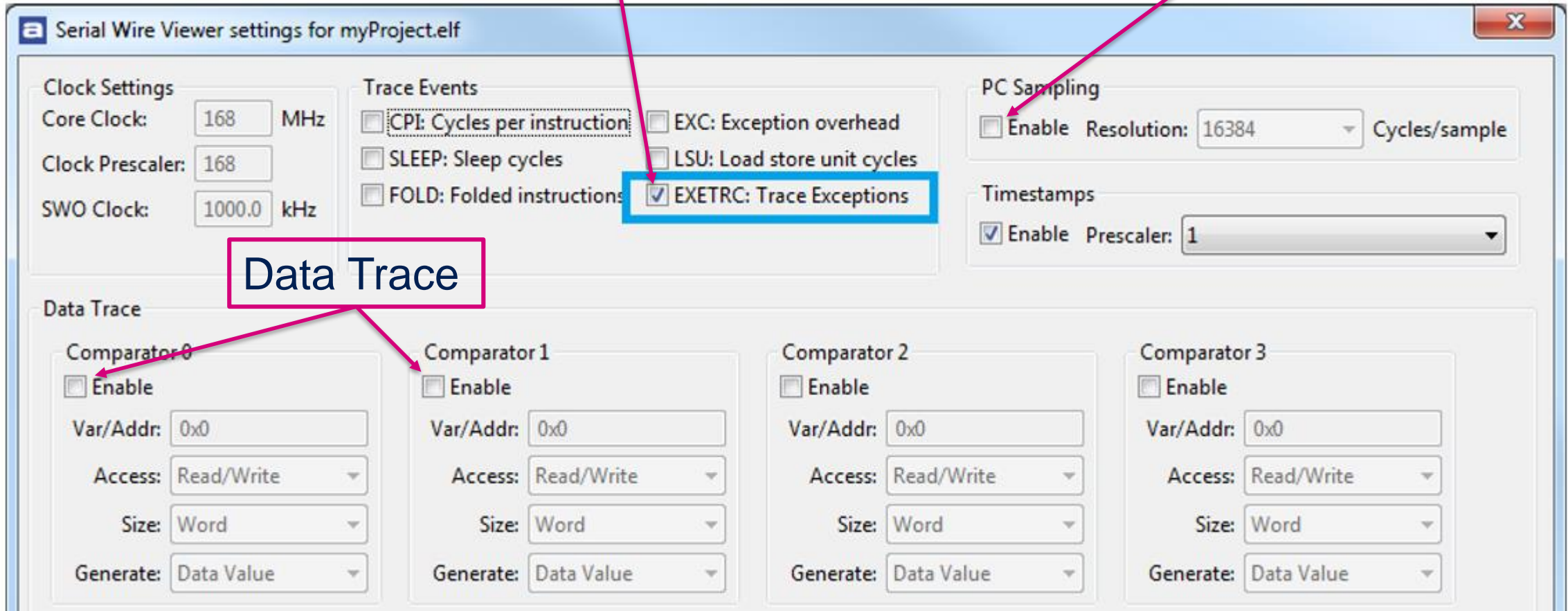
Overflow packets: 20

SWV(Serial Wire Viewer)

- SWV 설정

인터럽트&익셉션 Trace

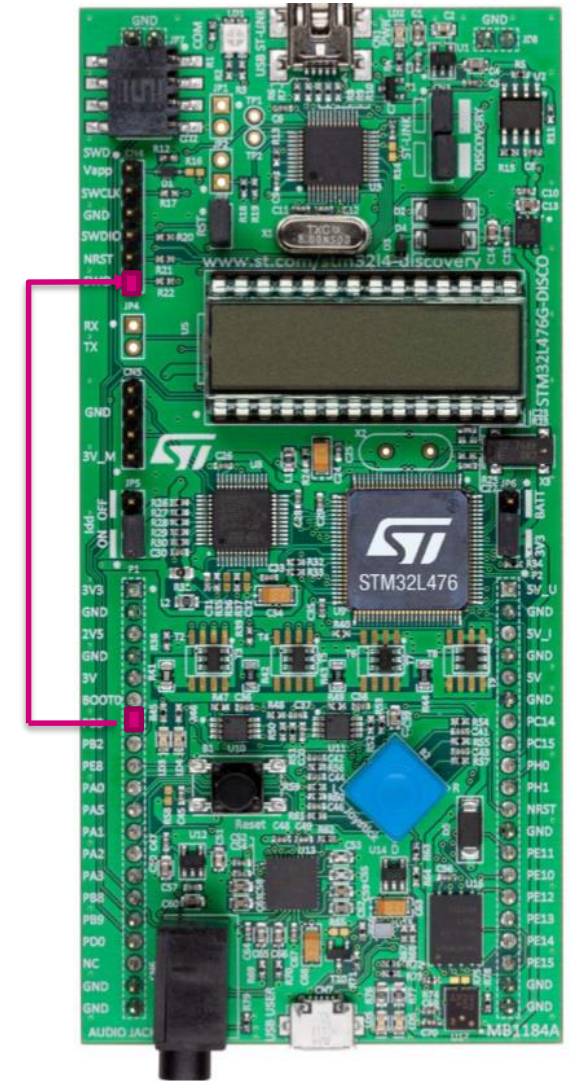
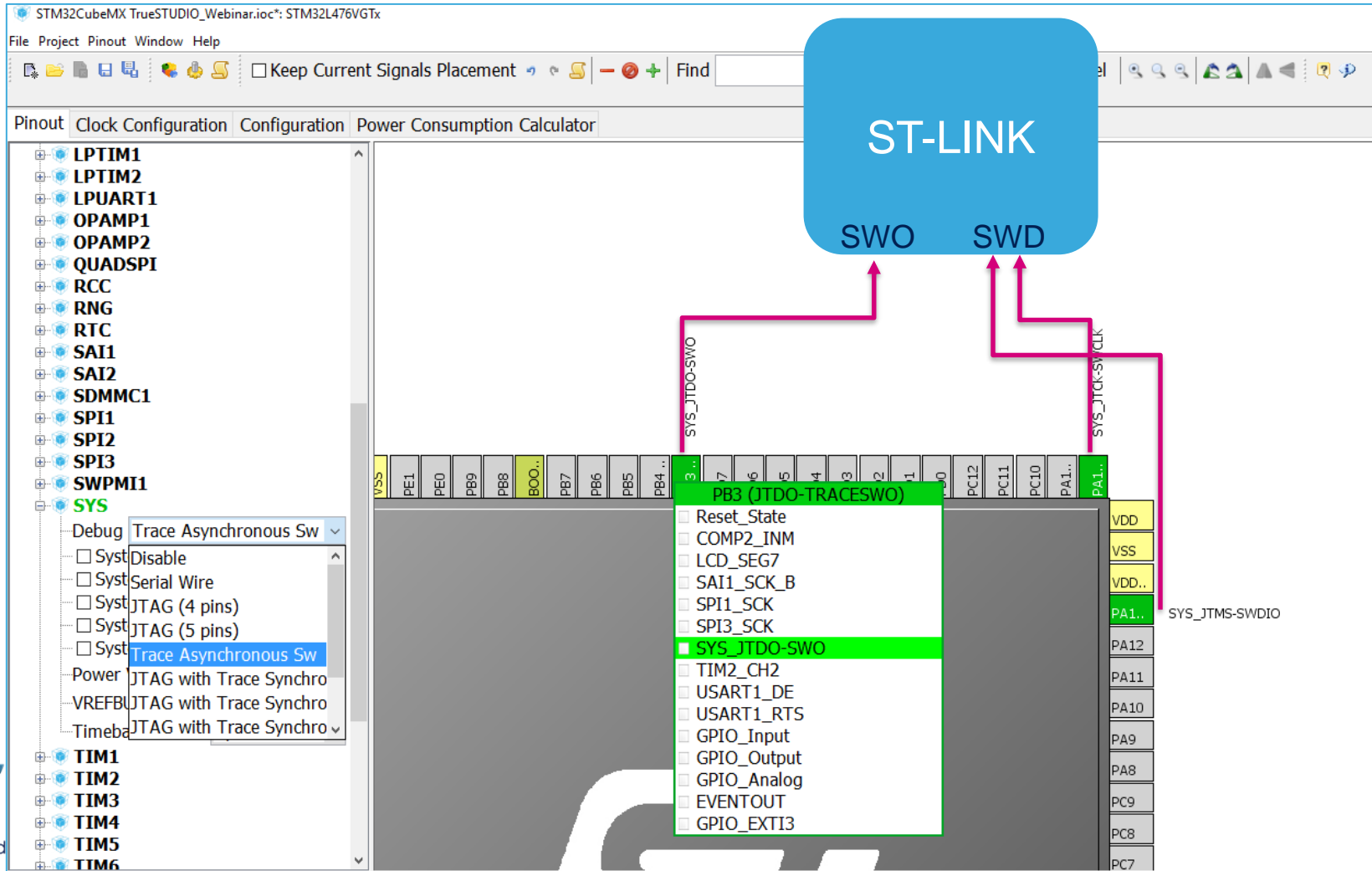
CPU점유율 분석



Data Trace

SWV(Serial Wire Viewer)

- 하드웨어 연결 (3 wire)



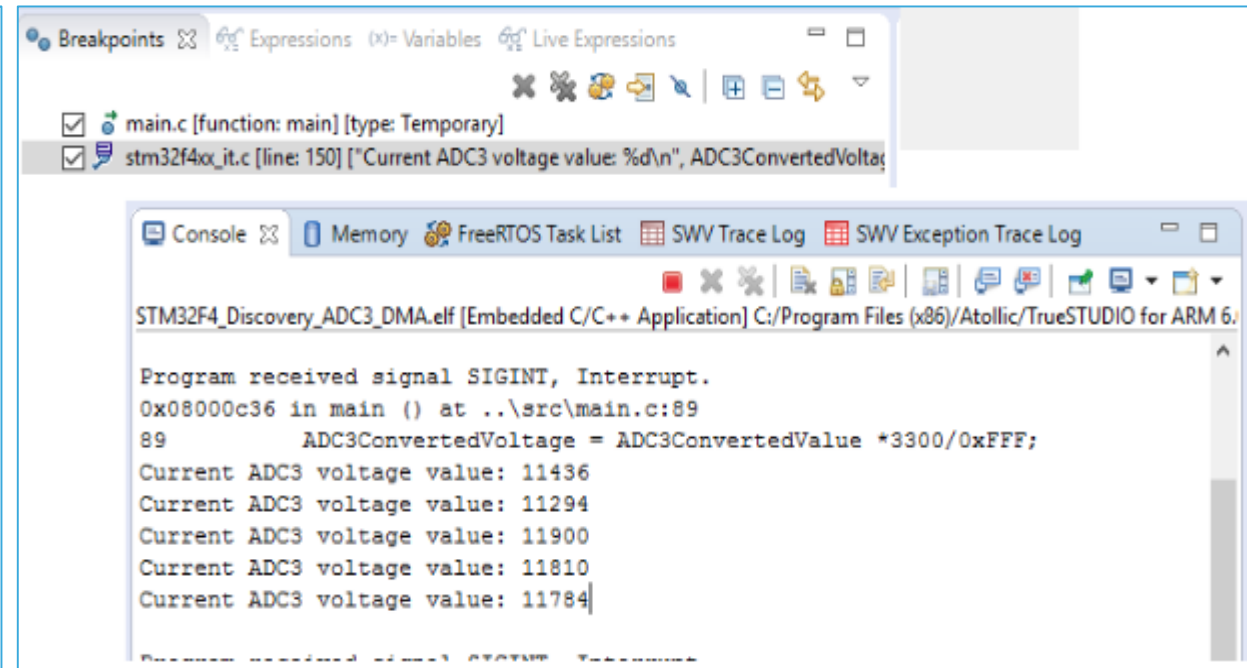
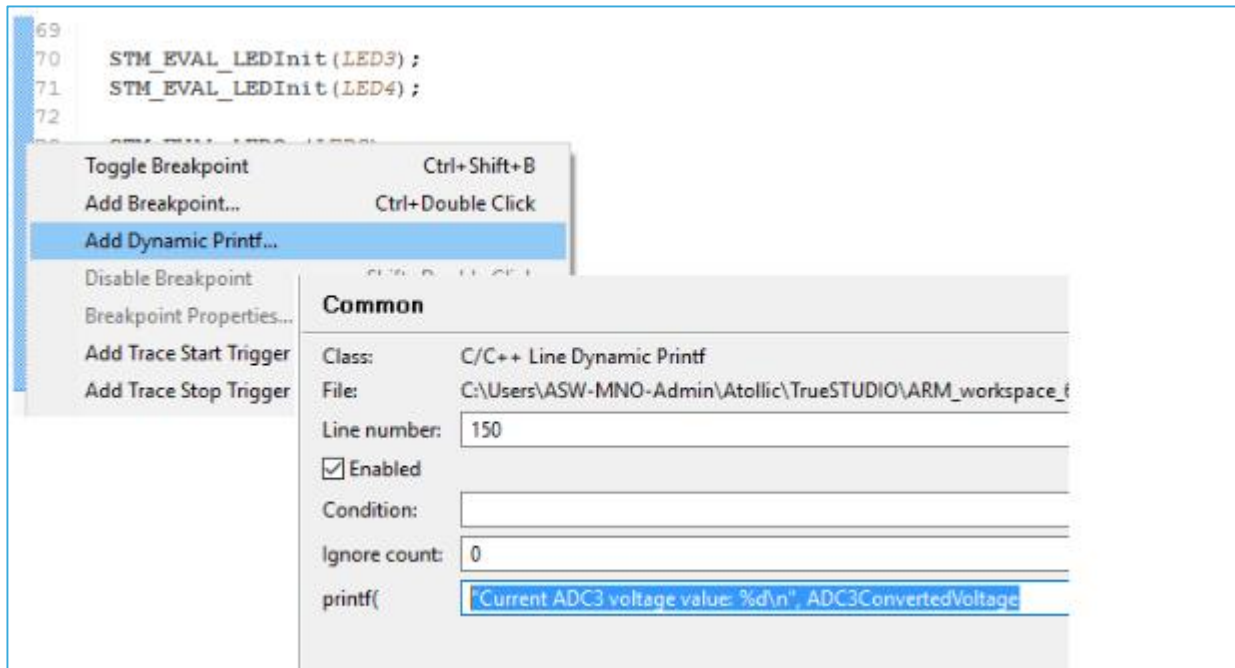
SWV(Serial Wire Viewer)

22

- Exception & Data Trace 시연
 - Data Trace, CPU usage %

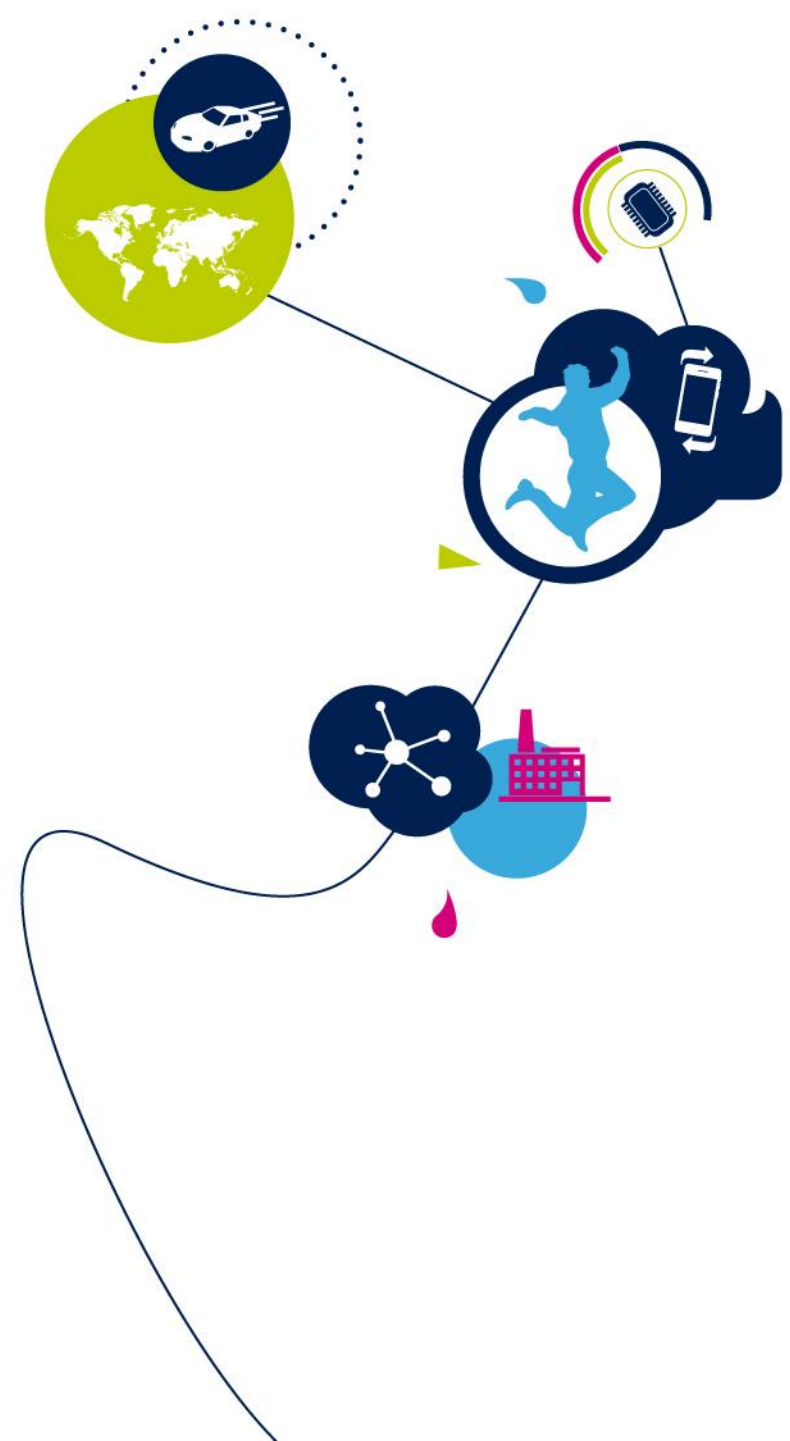
Dynamic printf()

- SWD 인터페이스 활용한 Log 출력 : printf 코드 삽입 / 빌드 불필요
 - Debug 모드에서 원하는 코드라인 왼쪽 가장자리에서 우측클릭 -> Add Dynamic Printf...

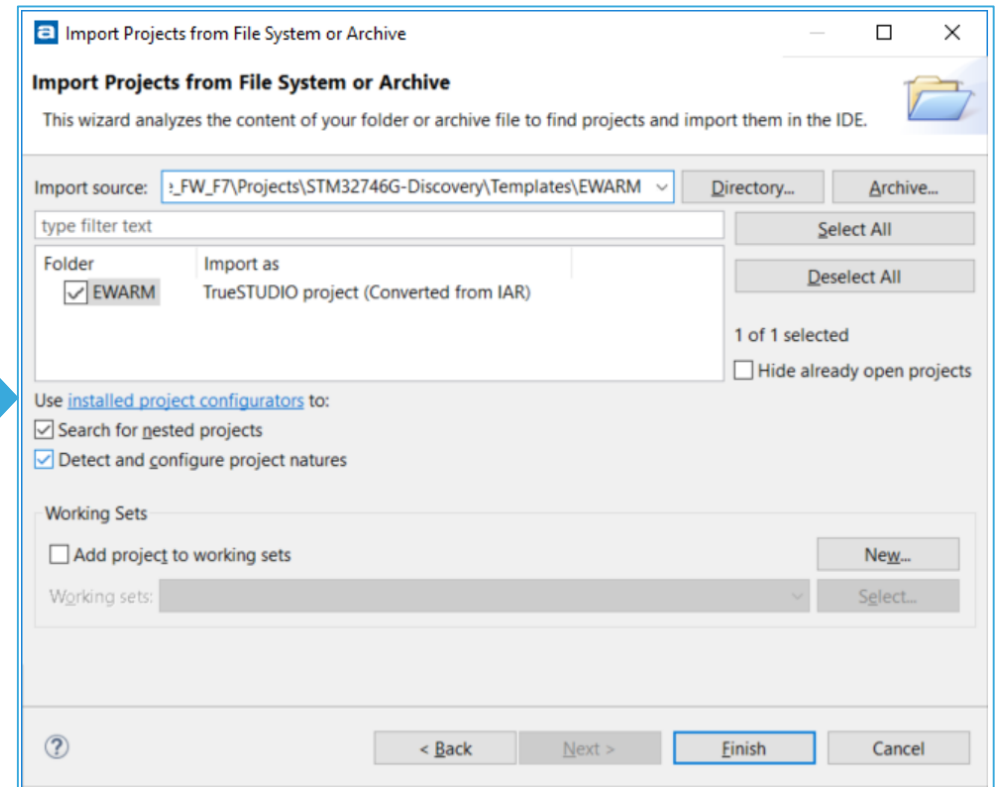
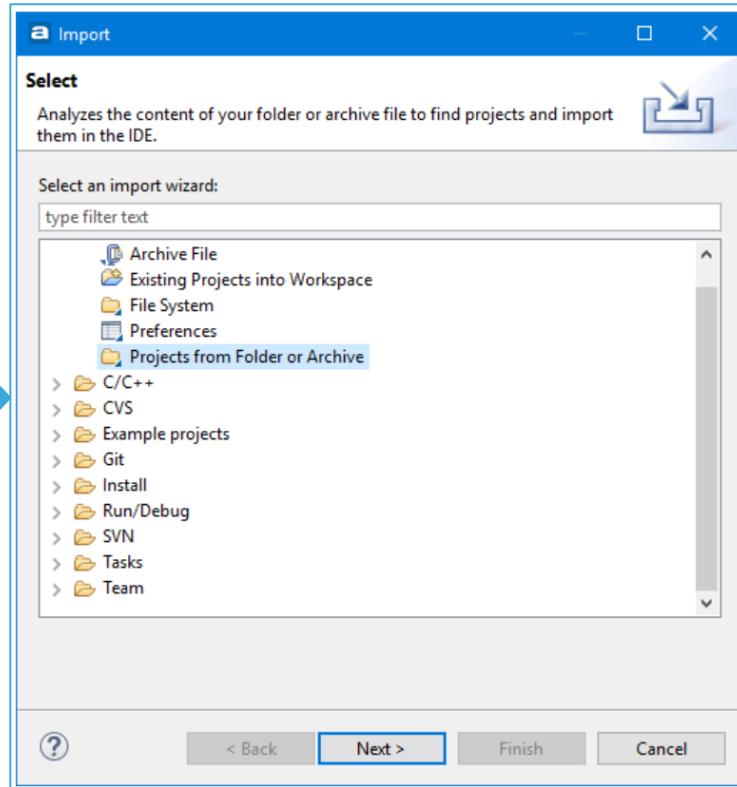
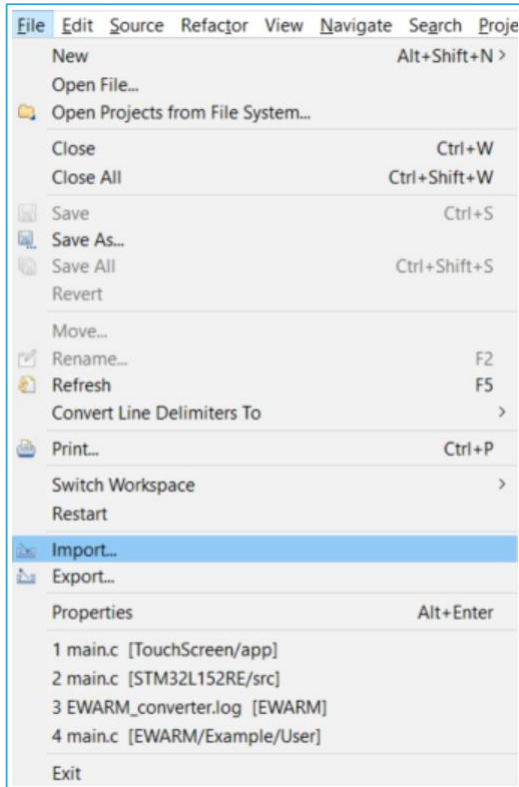


- Dynamic printf 사용 데모영상

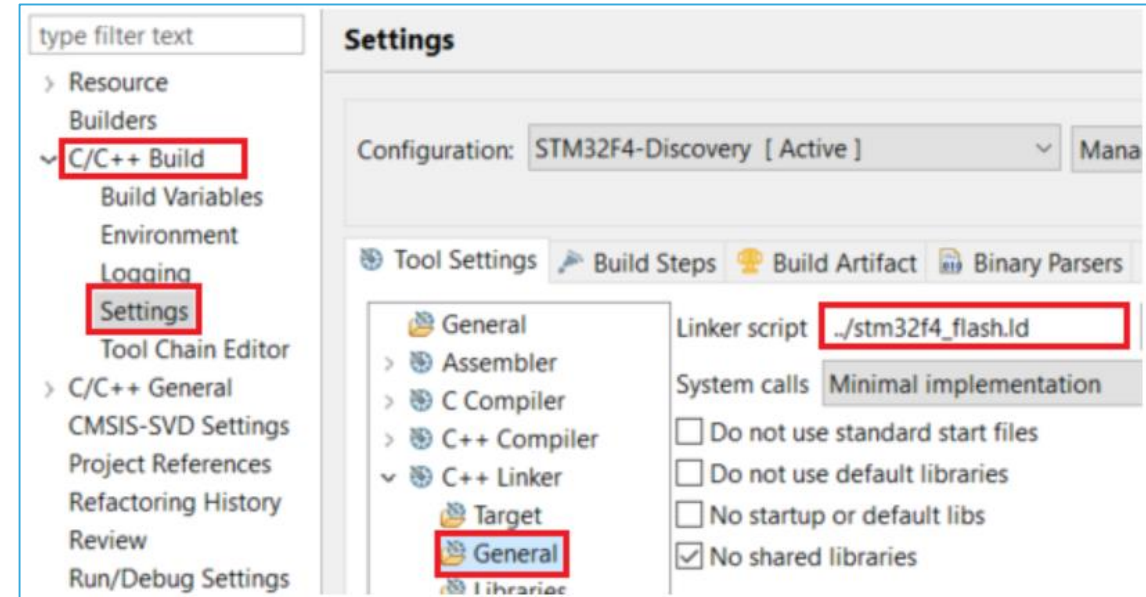
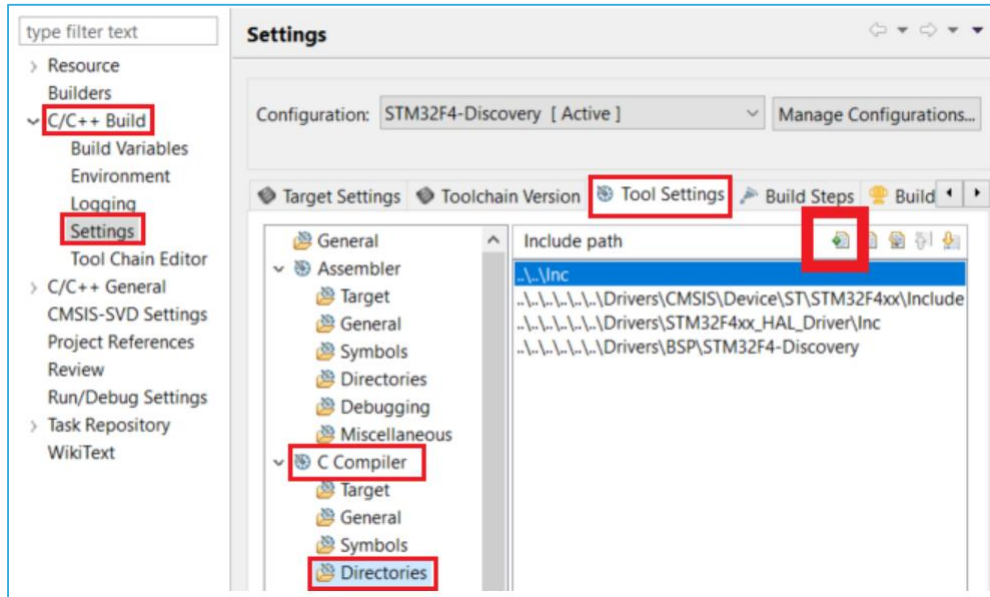
IAR 프로젝트 불러오기



- Import : Projects from Folder or Archive
 - 프로젝트 소스구조, 프로젝트 설정(Symbols, Include PATH) 자동으로 반영



- 추가로 필요한 작업
 - CMSIS 소스파일 참조 경로 추가
 - Startup assembler 소스파일 교체
 - TrueSTUDIO용 Linker script 파일 추가



- 데모시연

- YouTube 채널 : Atollic tools 구독

Atollic tools
577 subscribers

HOME VIDEOS PLAYLISTS CHANNELS DISCUSSION ABOUT

Uploads **PLAY ALL**

Video Title	Views	Upload Date	Duration
Printf redirect	827 views	2 months ago	2:14
2016 02 17 Advanced Debugging of ARM Cortex	465 views	2 months ago	50:43
Getting Started with TrueSTUDIO	4.9K views	2 months ago	1:03:35
CubeMX and TrueSTUDIO joint webinar with ST	2.5K views	2 months ago	57:30
Advanced Debugging of Cortex M Systems	297 views	2 months ago	47:23

- TrueSTUDIO의 최신 소식과 유용한 팁 다수

HOME TRUESTUDIO STM32 TARGET SUPPORT BUY RESOURCES BLOG ABOUT US

TruePERSPECTIVES Blog

Search this site on Google

STM32CubeProgrammer + Atollic TrueSTUDIO for STM32

Posted by Mattias Norlander on Feb 20, 2018 2:29:27 PM

[Tweet](#) [Share](#) [Like 16](#) [Share](#) [G+](#)

ST recently released [STM32CubeProgrammer](#), a software tool allowing users to flash STM32 devices through the SWD/JTAG debug interface and via the UAR/USB bootloader interface. STM32CubeProgrammer can be integrated with Atollic TrueSTUDIO. This is *how* and *why*!



[Read More](#)

11 Comments [Click here to read/write comments](#)

Topics: [STM32](#), [STM32Cube](#), [flash load](#)

TrueSTUDIO for STM32 9.0.0 released

Posted by Mattias Norlander on Jan 19, 2018 9:32:35 AM

[Tweet](#) [Share](#) [Like 163](#) [Share](#) [G+](#)

We made a promise in December 2017 to release a TrueSTUDIO for STM32 in early 2018.

Subscribe

Email*

Notification Frequency*

- Instant
- Daily
- Weekly
- Monthly

Recent Posts

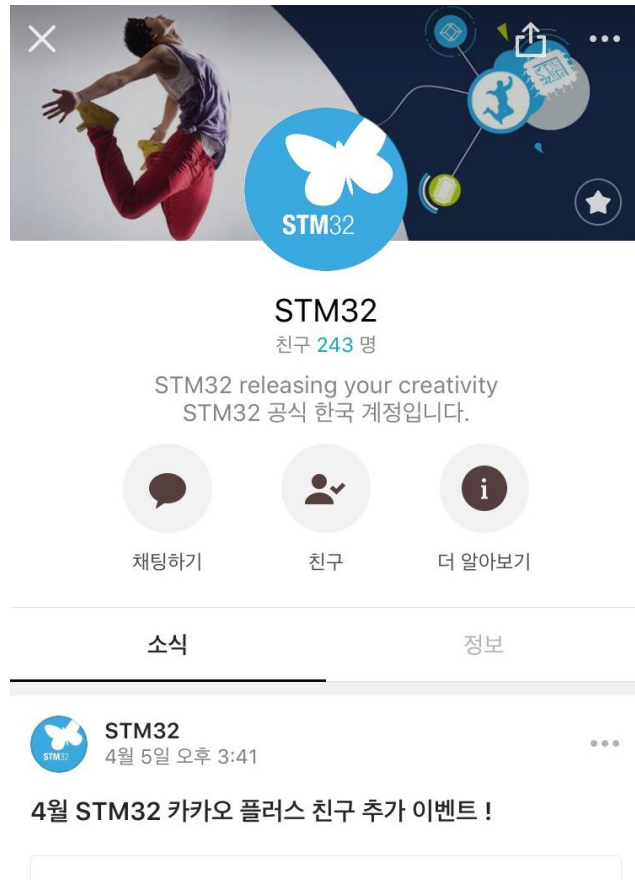
STM32CubeProgrammer + Atollic TrueSTUDIO for STM32
posted Feb 20, 2018

TrueSTUDIO for STM32 9.0.0 released
posted Jan 19, 2018

TrueSTUDIO Developers using STM32

STM32 카카오톡 플러스친구 등록

- STM32와 관련된 최신 소식을 카카오톡으로 편리하게 받아보세요.
 - 검색에서 'STM32' 입력 후 친구등록 또는 QR코드 촬영



Releasing your creativity with the STM32

32



▶ Famous video [here](#)

 [/STM32](#)

 [@ST_World](#)

 [community.st.com](#)

www.st.com/stm32