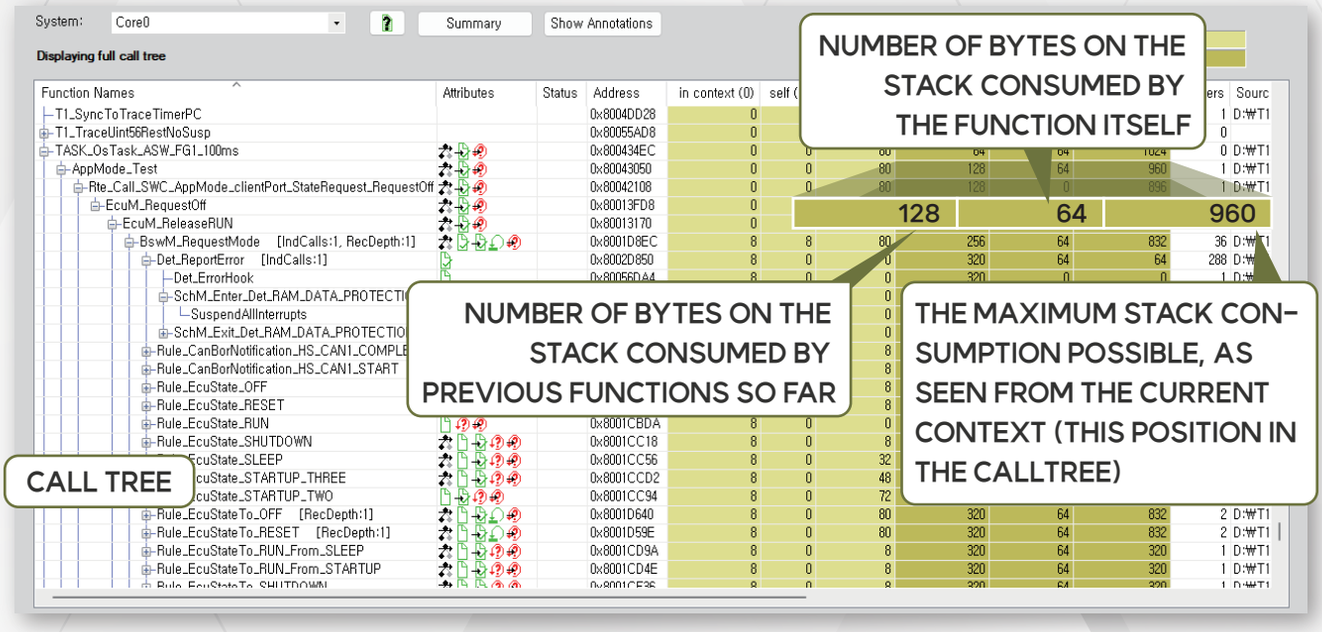
COST-EFFECTIVE, DETAILED AND ACCURATE STACK ANALYSIS

Introduction

Static stack analysis with T1.stack

T1.stack은 바이너리(ELF 파일) 기반 정적 코드 분석을 수행합니다.
: 1) 바이너리 disassemble, 2) 함수 호출 분석, 3) Call-tree 재구성, 4) 함수의 스택 사용량 결정



스택 사용량 분석에서의 간접 함수 호출(Indirect function calls)

정적 코드 분석을 통한 스택 사용량 분석에서 간접 함수 호출(indirect function calls) 문제를 해결해야 합니다.

간접 호출은 일반적으로 함수 포인터를 사용하기 때문에 런타임 중 호출할 수 있는 모든 호출 대상(함수)을 알아야 합니다.

T1.stack은 **주석(annotation)을 통해** 정적 분석의 약점을 보완할 수 있습니다.

T1.stack에서 지원하는 주석 기능은 1) 간접 호출 함수 추가, 2) T1.flex 측정을 통한 자동 함수 추가 기능 등이 있습니다.

T1.flex 기능을 통해 간단하게 호출 대상을 **측정**할 수 있는 것이 T1.stack만의 특별한 기능입니다.
이러한 측정 방식은 다른 프로젝트의 Call-tree를 교차 검증하는 데에도 사용할 수 있습니다.

Unresolved indirect function call:

Core0_1msRunnable0		0x800035A2	136	0	8	256	64	192
callIndirects [IndCalls:2]		0x8000510E	136	0	8	256	64	0
funC		0x80005106	144	8	8	192	0	0
TA_Button_Read		0x8000400C	136	0	0	256	0	0
T1_DelayRoutinePC		0x80008F8E	136	0	0	320	64	128
Engine_RPMTick		0x80004952	136	0	0	256	0	0

Resolved indirect function call by dynamic T1.flex measurement:

Core0_1msRunnable0		0x800035A2	136	0	8	256	64	192
callIndirects [IndCalls:2]		0x8000510E	136	0	8	256	64	64
funC		0x80005106	144	8	8	192	0	0
funB		0x800050F4	136	0	0	256	0	0
funA		0x800050E2	136	0	0	256	0	0
TA_Button_Read		0x8000400C	136	0	0	256	0	0
T1_DelayRoutinePC		0x80008F8E	136	0	0	320	64	128
Engine_RPMTick		0x80004952	136	0	0	256	0	0

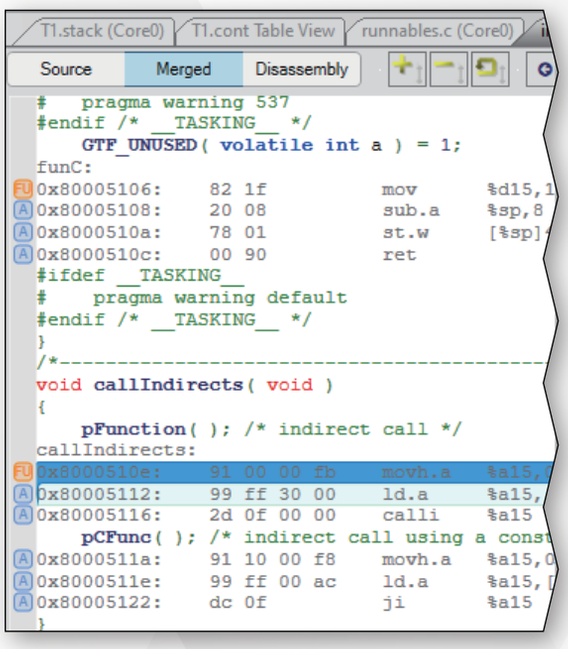
T1. stack을 활용하시면 스택 사용량뿐만 아니라 스택이 어떻게, 왜 사용되는지를 자세하게 확인할 수 있습니다.

스택 사용 현황을 파악하여 스택 사용을 최적화하고, 의도하지 않았거나 목적이 없는 스택 사용을 파악할 수 있습니다.

스택을 적게 사용한다면 런타임 성능 또한 향상시킬 수 있습니다. C언어와 같은 고급 언어를 사용하는 경우, C 코드에 대한 지식만으로는 스택 사용량을 예측할 수 없습니다. 또한, 자동으로 생성되는 코드의 경우 문제가 더욱 그렇습니다.

T1.stack을 사용하면 스택 소비를 지속적으로 추적할 수 있으므로 코딩 및 컴파일러의 영향을 모니터링하고 이해할 수 있습니다.

T1.stack의 정확한 분석을 통해 스택의 크기를 안정적으로 분석할 수 있기 때문에 불필요한 메모리 할당을 피할 수 있으며, 스택 오버 플로우를 방지할 수 있습니다.



Key benefits include:

- **바이너리 파일** 기반 정적 분석
- 간접 함수 호출(함수 포인터)의 **측정 지원**
- **최강의 분석 속도** (예: 엔진 제어기의 150MB ELF 파일을 일반 PC에서 2분 이내 분석 가능)
- **Call-tree를 통한** 소프트웨어 구조 파악 가능
- **소스코드 & 디스어셈블리 뷰어 제공**

Technical data

Supported CPU architectures:

- Infineon TriCore (*)
- ARM, ARM Thumb
- PowerPC, PowerPC VLE (*)
- RH850
- x86

(*) enhanced static analysis engine

경기도 용인시 기흥로 58-1
기흥ICT밸리 A동 1205, 1207호

itivai.com

+82-31-5174-9890
support@itivai.com