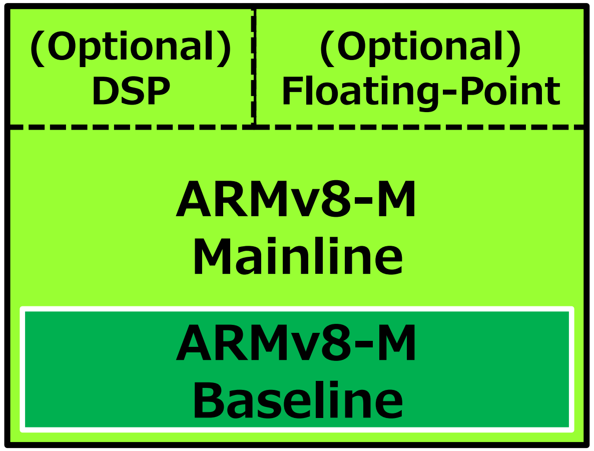
ARMv8-M TrustZone:
A New Security Feature for Embedded Systems

FFRI, Inc.
http://www.ffri.jp

ARMv8-M Architecture

- Architecture for embedded devices (Cortex-M Processor family) which was announced in Nov 2015.

- In order to comprehensively support for embedded systems that require the characteristics of the conventional ARMv6-M/ARMv7-M architecture, ARMv8-M has prepared 2 sub-profiles.
  - Baseline
    - For Ultra-low-power products
    - Similar to the ARMv6-M
  - Mainline
    - A full-featured, microcontroller products and high-performance embedded systems.
    - Similar to the ARMv7-M

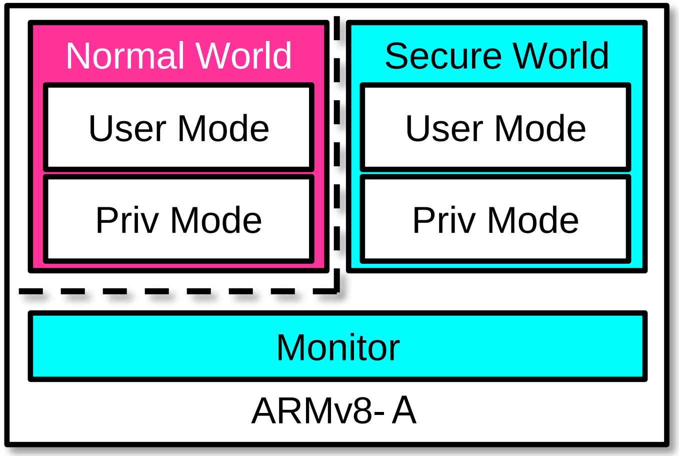


TrustZone

- Security features that ARM processor provides.
  - Cortex-A family or next-generation Cortex-M processors
- It is possible to separate/isolate the security level by adding the security state.
  - e.g. Normal World & Secure World
- ARMv8-M architecture has a different mechanism than TrustZone to provide traditional ARMv8-A architecture, which is optimized for embedded systems.

TrustZone (ARMv7, ARMv8-A, etc…)

- Add a monitor mode, it is separated into “Normal World” and “Secure World”.
  - To transition monitor mode, use SMC instruction.
  - A kind of virtualization feature using the OS monitor.



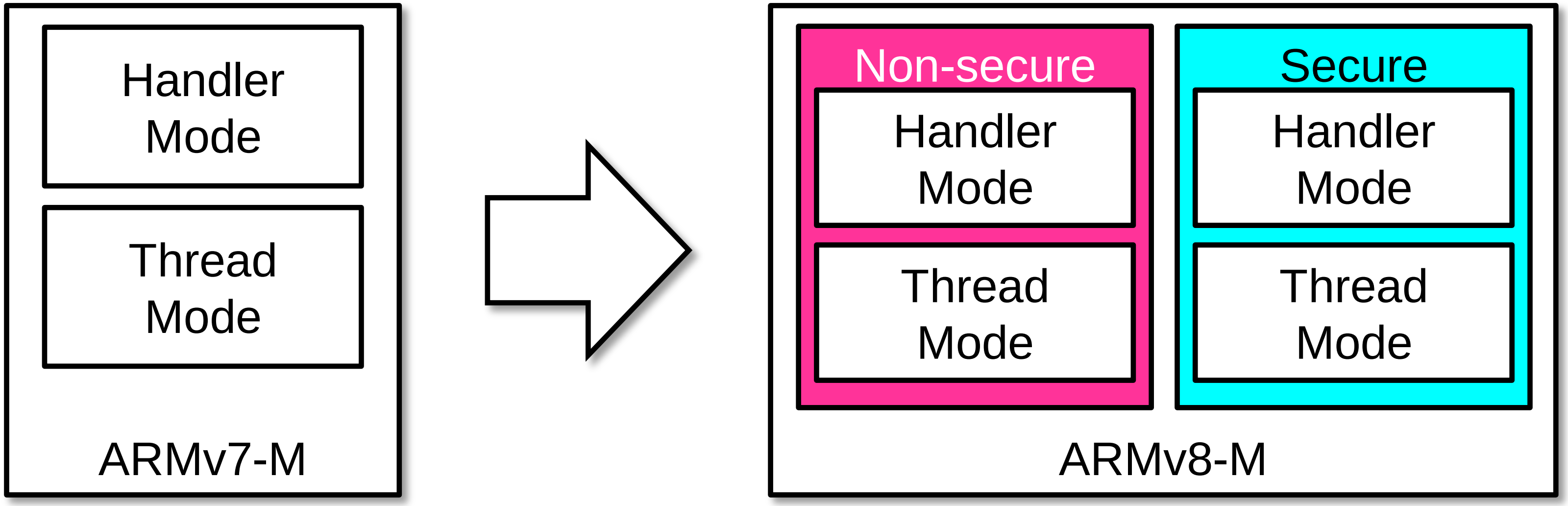
- iPhone of Secure Enclave are known to have been using the TrustZone.

- For more information, please refer to the our research paper, which was published in March 2013. (Japanese only)

TrustZone (ARMv8-M)

- Add a secure state, it is possible into Non-secure Handler/Thread mode and Secure Handler/Thread mode.
  - The state transition to use the branch instruction.
  - System rises by default in the “Secure” state.

- Throughout the reminder of this article describes ARMv8-M TrustZone.

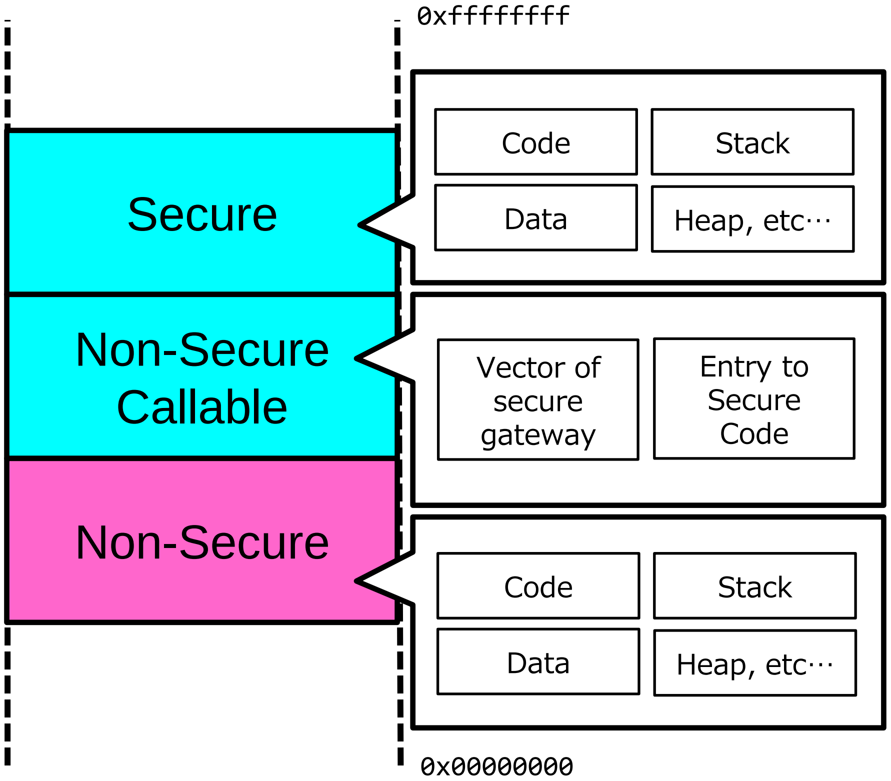


ARMv8-M TrustZone - Memory space separation

- In addition to the definition by the developer of microcontrollers and SoC, it can also be defined the software by utilizing the SAU and IDAU interfaces.

- Memory spaces can be divided into three.
 (See the figure on the right)

- State of the processor is dependent on definition of the memory space.



SAU: Software Attribution Unit
IDAU: Implementation Defined Attribution Unit

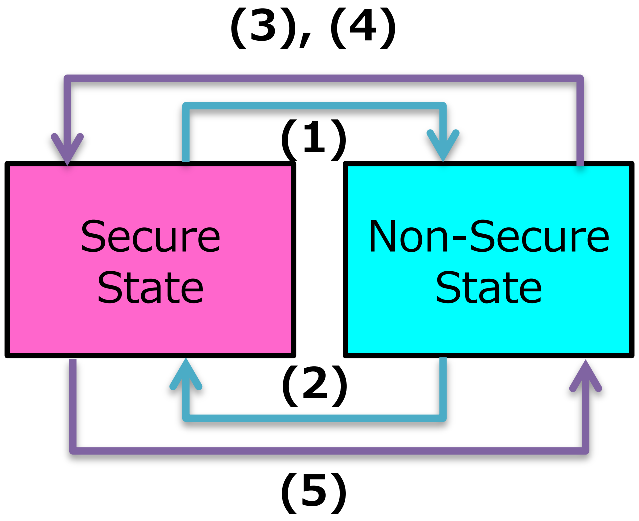
ARMv8-M TrustZone – Secure Gateway

- To call processing of the Secure region from the Non-Secure region, it is necessary to relay a secure gateway.
  - The first instruction of the function to be called from Non-Secure region MUST always SG (Secure Gateway) instruction.
  - SG instruction MUST be present in the NSC (Non-Secure Callable) region.

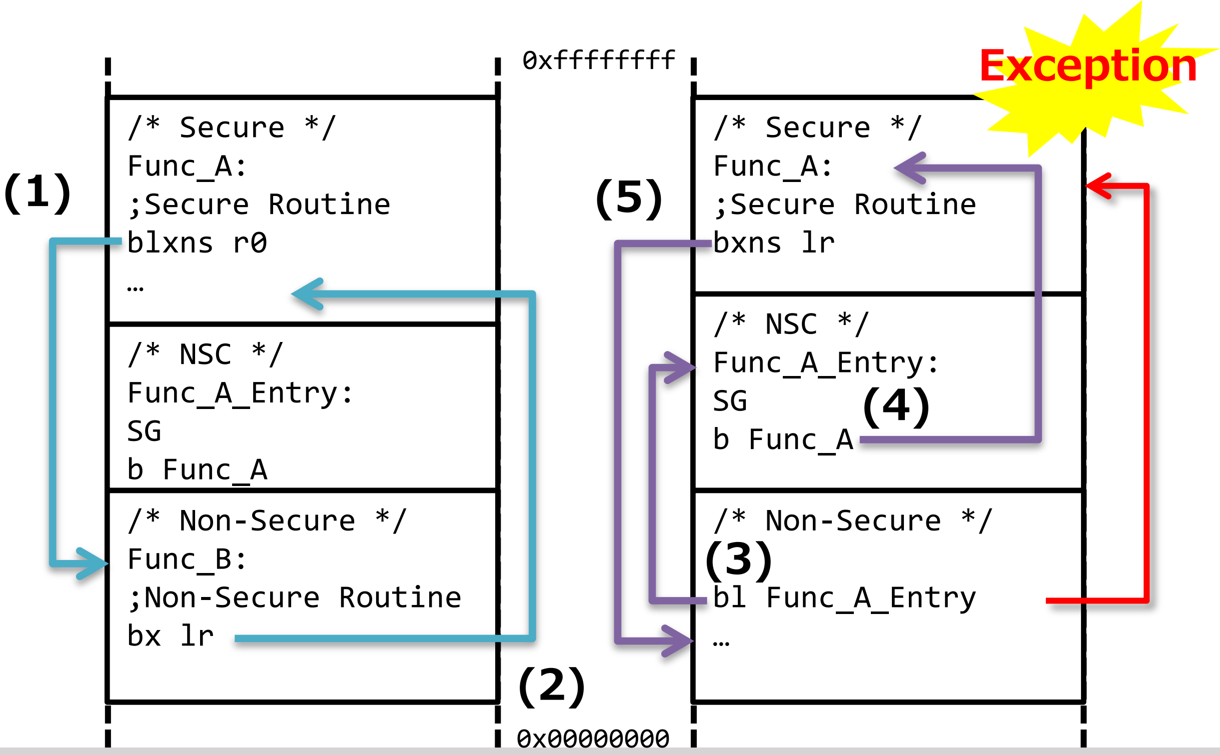
- In case of call processing of the Non-Secure region from the Secure region, push current state to stack and then branch to Non-Secure region.
  - When processing branch to the Non-Secure region, reserved value FNC_RETURN is set to Link Register. (LR)
  - When returning to Secure region branches to this Link Register. (FNC_RETURN)

ARMv8-M TrustZone - Secure Gateway

- If Non-Secure region program accessed directly to address of the Secure region occurs following exception.
  - In Mainline SecureFault(7), in Baseline HardFault(3) is an exception occurs in Secure State.



State transition of the processor



```
/* Secure */
Func_A:
;Secure Routine
blxns r0
…

/* NSC */
Func_A_Entry:
SG
b Func_A

/* Non-Secure */
Func_B:
;Non-Secure Routine
bx lr
```

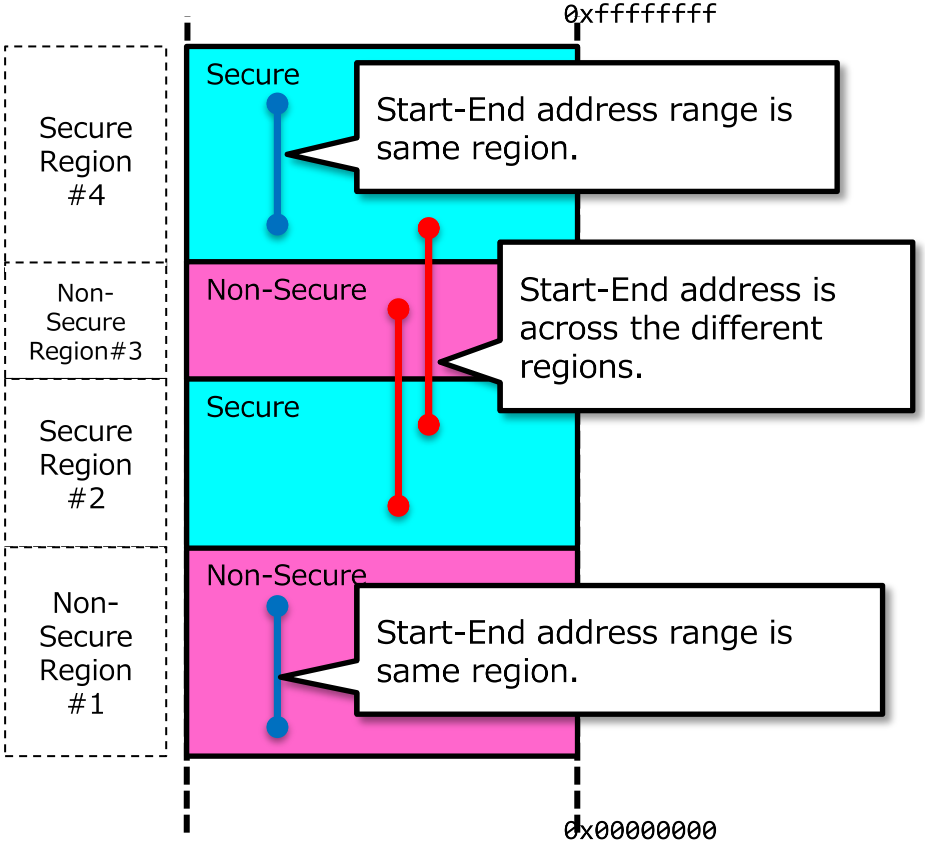
```
/* Secure */
Func_A:
;Secure Routine
bxns lr

/* NSC */
Func_A_Entry:
SG
b Func_A

/* Non-Secure */
bl Func_A_Entry
…
```

ARMv8-M TrustZone - Test Target

- Region number is assigned in the memory space defined by the aforementioned SAU and IDAU.
  - Possible to know whether it has the security attribute target is continuous by the region number.
- New TestTarget (TT) instruction to return security attributes and region number from the address.
  - By using TT instruction, it is possible to know address range of the array or structure is belong to the same region.



Start-End address range is same region.

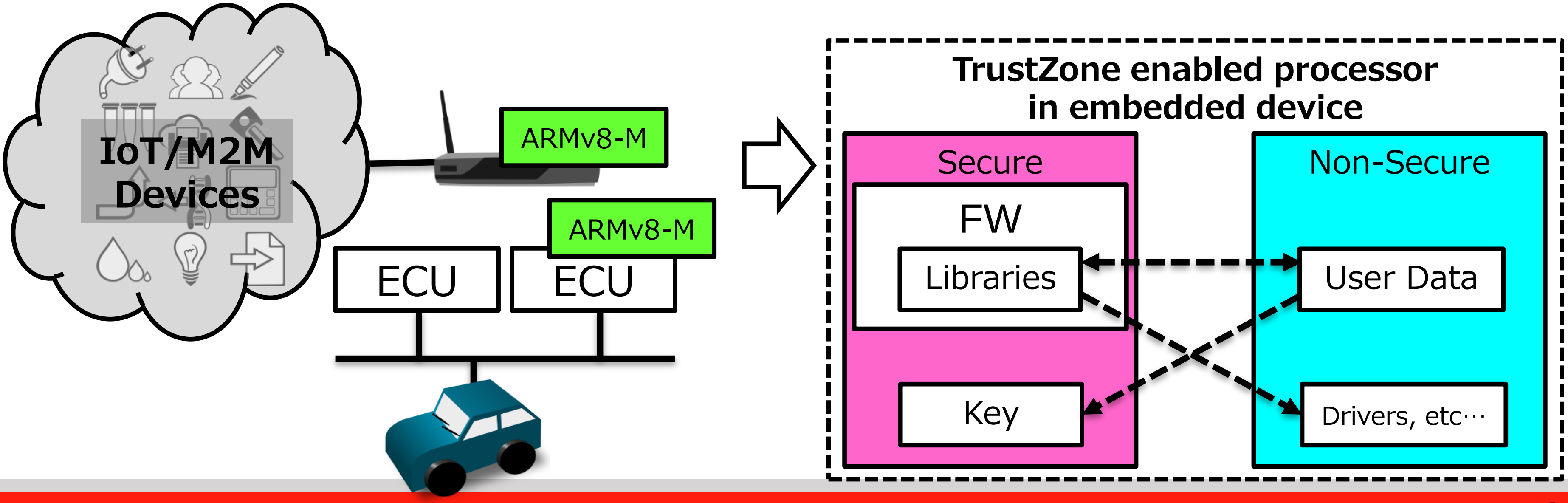
Start-End address is across the different regions.

Start-End address range is same region.

ARMv8-M TrustZone - example usage for embedded systems

- Even for embedded device architecture that supported the TrustZone, protection of data it is realistic also due to this technology for a variety of IoT and in-vehicle devices.

- For example, IoT device vendors by storing in advance the firmware in the Secure region, it can be expected that the reverse engineering measures.



Summary

- In this paper, we introduce the TrustZone of information that has published at this time in relation to ARMv8-M.
  - There is a specification change possibility in the future because some document is still Beta.
- In Febrary 2016, the processor and evaluation board of ARMv8-M architecture has not been confirmed in the market.
  - For even compiler, GCC and Clang is currently working.
- For automotive, already HSM (Hardware Security Module) is present as a standard.
  - Therefore, semiconductor manufactures are mainly shipped microcontroller products that conform to this standard as automotive.
  - With the advent of the ARMv8-M, the future there is a possibility that products utilizing the TrustZone is announced.

References

- Whitepaper – ARMv8-M Architecture Technical Overview
  - https://community.arm.com/docs/DOC-10896
- ARM® コンパイラ ソフトウェア開発ガイド バージョン6.3
  - http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.dui0773dj/pge1446115999905_00009.html
- (動画) ARMv8-M architecture: what's new for developers
  - https://youtu.be/V5zr5mPjAvU
- FFRI Monthly Research – セキュアハードウェアの登場とその分析
  - http://www.ffri.jp/assets/files/monthly_research/MR201303_TrustZone.pdf