How to Out-of-Tree a Trusted-Firmware-M (TF-M) Board

for the STM32U5A5x MCU

Gerson Fernando Budke
Leica Geosystems AG

Reference: github.com/zephyrproject-rtos/zephyr/pull/94875

Agenda

01 What is TF-M?

02 The nucleo_u5a5zj_q Board & STM32U5A5 Security

03 Why Out-of-Tree? The Problem Statement

04 The Zephyr-Hosted TF-M Board Model (PR #94875)

05 Project Structure & File Layout

06 Major details

07 Conclusion

About the Author

Gerson Fernando Budke (nandojve)

Embedded Software Engineer | Leica Geosystems AG

Zephyr Project Maintainer (Since 2019)

Microchip & Atmel platform maintainer

Extensive contributions: drivers, SoC bring-ups, board support, security fixes

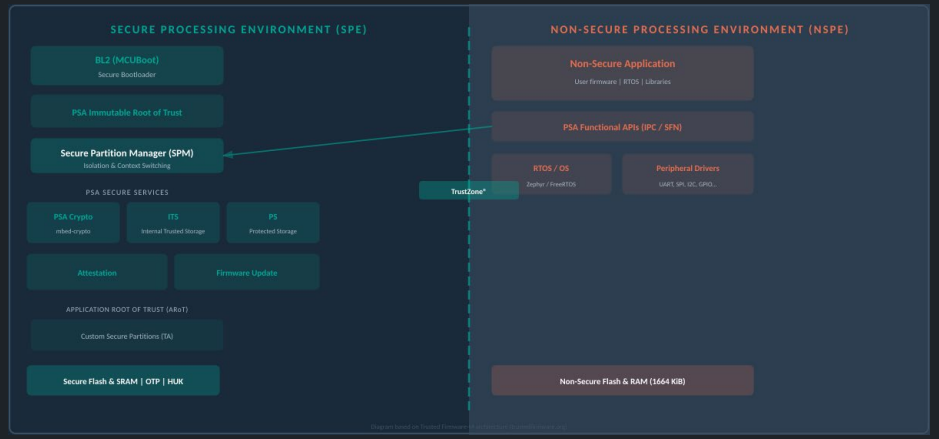
Strong expertise in device tree architecture, TrustZone integration, secure boot, and FOTA

Key Areas

TF-M/PSA • Secure boot chains • STM32 & Microchip microcontrollers

TF-M 101

Trusted Firmware-M is Arm's open-source reference implementation of the **Platform Security Architecture (PSA)** for Cortex-M processors.



- Implements SPE for Armv8-M / Armv8.1-M
- Isolates Secure & Non-Secure worlds via TrustZone®
- Provides PSA Crypto, ITS, PS, Attestation, FWU
- PSA Certified – Levels 1, 2 & 3

nucleo_u5a5zj_q & STM32U5A5 Security

STM32U5A5ZJ MCU

Arm Cortex-M33 @ 160 MHz | Armv8-M with TrustZone®

3 MiB Flash | 1664 KiB NS SRAM

ST-LINK/V3 debug | USB 2.0 HS

Arm TrustZone®

Hardware isolation between S and NS worlds enforced at the core.

GTZC Controller

Assigns security attributes to all peripherals and memory regions.

SAU / MPU

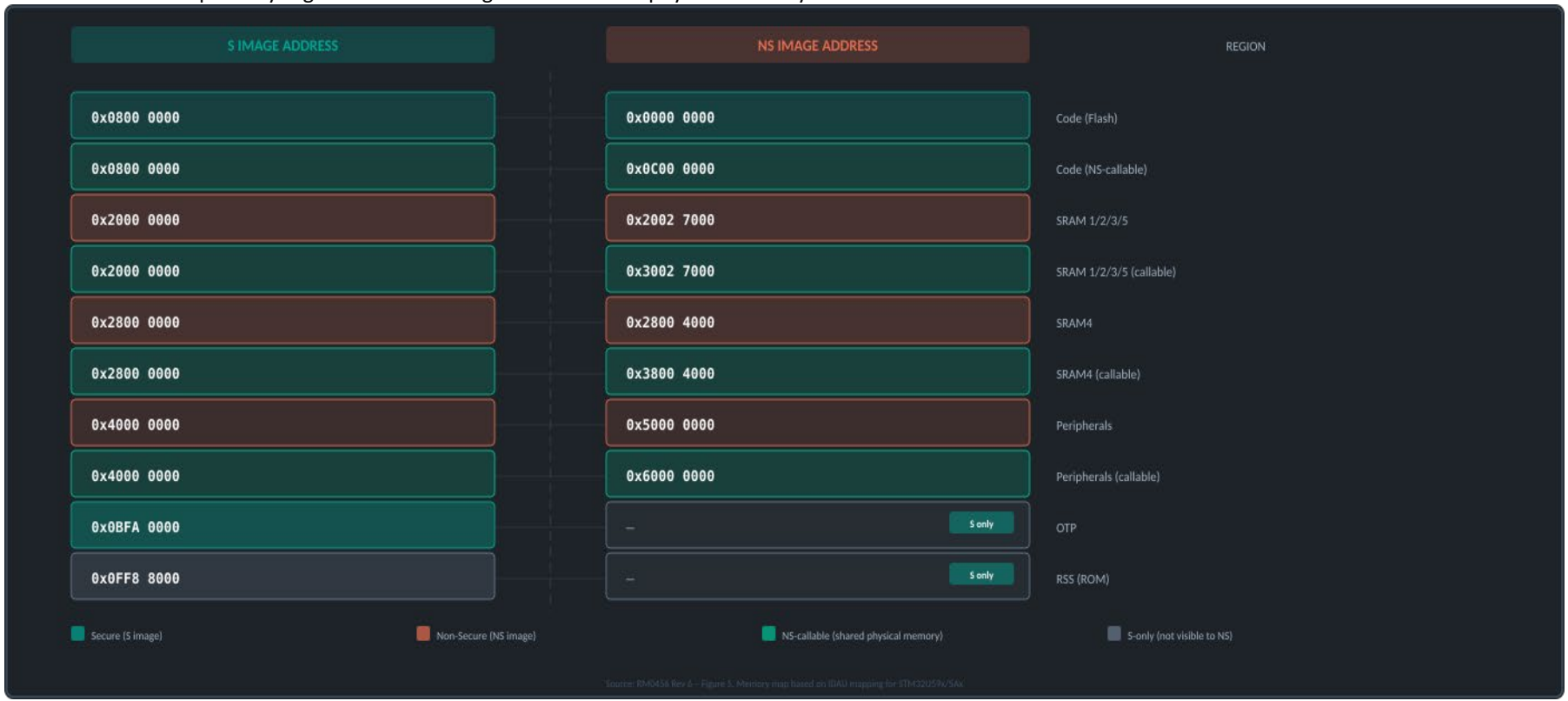
Security Attribution Unit + Memory Protection Unit enforce runtime boundaries.

RDP & OTP

Read Protection (levels 0–2) and OTP memory lock secrets after provisioning.

STM32U5A5 – S vs NS Address Space

The IDAU remaps every region. S and NS images see the same physical memory at different addresses.



Why Out-of-Tree? The Problem

Vendor boards only in TF-M mainline

Real products (e.g. Leica) will never appear upstream — only dev-kit references exist.

Memory layout differs per PCB

Custom boards have different flash/RAM partitioning. The upstream layout is only a starting point.

Zero out-of-tree examples existed

Our team had to reverse-engineer otp_provision.c and discovered critical TF-M bugs in the process.

Key Insight

TF-M + Zephyr require two in-sync boards.

Evolution of both projects causes recurring breakage when boards are maintained separately.

Solution: define the TF-M board inside the Zephyr board tree so both sides change atomically.

The Zephyr-Hosted TF-M Board Model

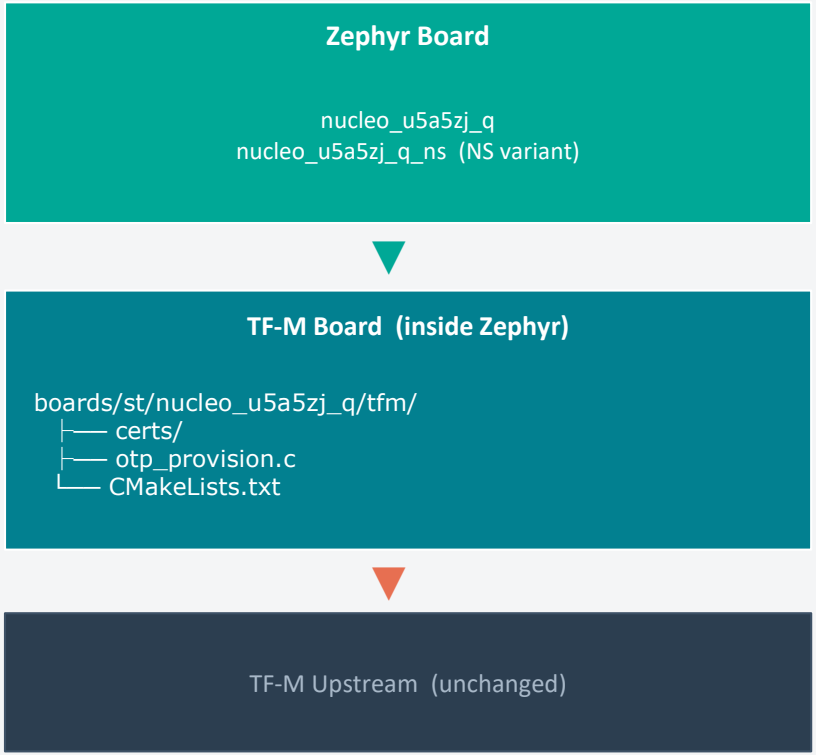
PR #94875 proposes:

The TF-M board definition lives inside **boards/st/nucleo_u5a5zj_q/tfm/** — not inside TF-M mainline.

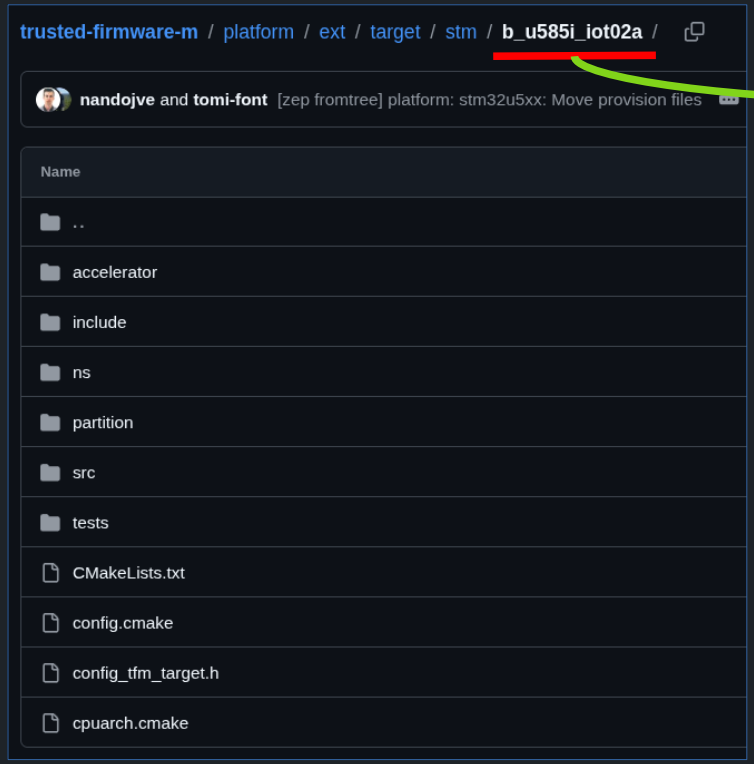
This means both the Zephyr board and its TF-M counterpart can be updated atomically in a single commit, keeping bisect valid.

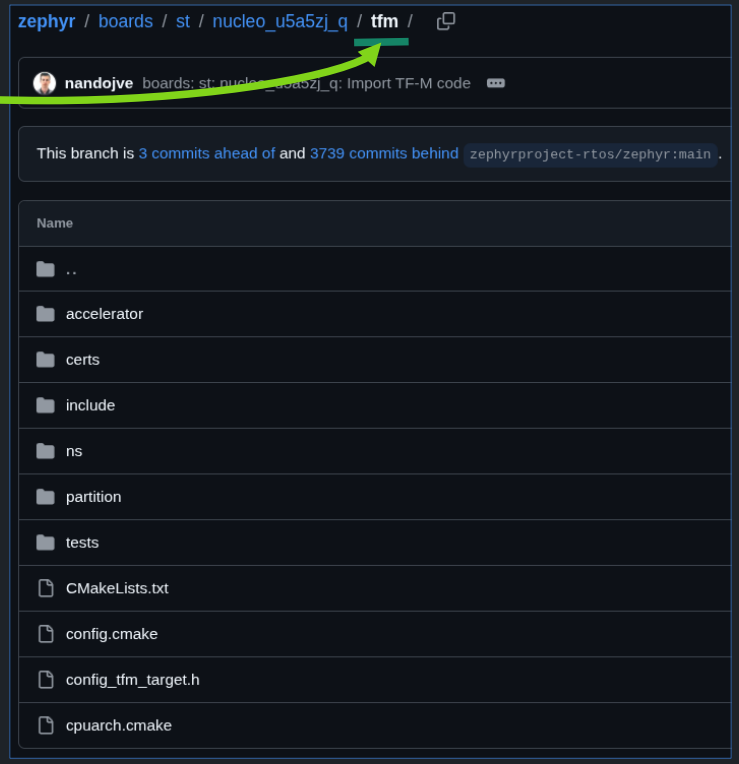
Benefits

- Atomic sync between Zephyr & TF-M boards
- Bisect stays valid across changes
- Custom otp_provision per board
- Certs kept alongside board — easy to swap for HSM



Project Structure & File Layout





Project Structure & File Layout

Key files introduced by PR #94875:

certs/rsa-2048-private-bl2.pem

BL2 bootloader signing key (RSA-2048)

certs/rsa-3072-private-s.pem

Secure image signing key (RSA-3072)

certs/rsa-3072-private-ns.pem

Non-secure image signing key (RSA-3072)

otp_provision.c

Board-specific OTP provisioning logic

CMakeLists.txt

Build integration for TF-M board

boards/st/nucleo_u5a5zj_q/

```
├── board.yml
├── nucleo_u5a5zj_q.dts
├── nucleo_u5a5zj_q_ns.dts
├── nucleo_u5a5zj_q.dtsi
├── nucleo_u5a5zj_q_ns_defconfig
├── tfm/
│   ├── certs/
│   │   ├── rsa-2048-private-bl2.pem
│   │   ├── rsa-3072-private-s.pem
│   │   └── rsa-3072-private-ns.pem
│   ├── otp_provision.c
│   └── CMakeLists.txt
└── doc/
```

Major Details

zephyr / boards / st / nucleo_u5a5zj_q / **Kconfig.defconfig**

nandojve boards: st: nucleo_u5a5zj_q: Add support to TF-M

Code | Blame | 24 lines (16 loc) · 488 Bytes

```
1  # Copyright (c) 2025
2  # SPDX-License-Identifier: Apache-2.0
3
4  if BOARD_NUCLEO_U5A5ZJ_Q
5
6  config TFM_BOARD
7          default "${ZEPHYR_BASE}/boards/st/nucleo_u5a5zj_q/tfm"
```

zephyr / boards / st / nucleo_u5a5zj_q / **nucleo_u5a5zj_q_stm32u5a5xx_ns_defconfig**

nandojve boards: st: nucleo_u5a5zj_q: Add support to TF-M

Code | Blame | 36 lines (29 loc) · 754 Bytes

```
1  #
2  # Copyright (c) 2025 Leica Geosystems AG
3  #
4  # SPDX-License-Identifier: Apache-2.0
5  #
6
7  # enable uart driver
8  CONFIG_SERIAL=y
9
10 # enable GPIO
11 CONFIG_GPIO=y
12
13 # console
14 CONFIG_CONSOLE=y
15 CONFIG_UART_CONSOLE=y
16
17 # Enable MPU
18 CONFIG_ARM_MPU=y
19
20 # Enable HW stack protection
21 CONFIG_HW_STACK_PROTECTION=y
22
23 # TF-M
24 CONFIG_ARM_TRUSTZONE_M=y
25 CONFIG_RUNTIME_NMI=y
26 CONFIG_TRUSTED_EXECUTION_NONSECURE=y
27
28 # Keys/Certificates
29 #
30 #       For development purposes only. These must be changed for a real product.
31 #
32 # The otp_provision.c will be picked from the same rsa-3072-private-s.pem folder.
33 #
34 CONFIG_TFM_MCUBOOT_SIGNATURE_TYPE="RSA-3072"
35 CONFIG_TFM_KEY_FILE_S="${BOARD_DIR}/tfm/certs/rsa-3072-private-s.pem"
36 CONFIG_TFM_KEY_FILE_NS="${BOARD_DIR}/tfm/certs/rsa-3072-private-ns.pem"
```

Major Details

zephyr / boards / st / nucleo_u5a5zj_q / **nucleo_u5a5zj_q_stm32u5a5xx_ns.dts**

nandojve boards: st: nucleo_u5a5zj_q: Add support to TF-M

Code | Blame | 98 lines (83 loc) · 2.05 KB

```
/*
 * Copyright (c) 2025 Leica Geosystems AG
 *
 * SPDX-License-Identifier: Apache-2.0
 */

/dts-v1/;
#include "nucleo_u5a5zj_q-common.dtsi"

/ {
        model = "STMicroelectronics STM32U5A5ZJ-NUCLEO-Q board";
        compatible = "st,stm32u5a5zj-nucleo-q";

        #address-cells = <1>;
        #size-cells = <1>;

        chosen {
                zephyr,console = &usart1;
                zephyr,shell-uart = &usart1;
                zephyr,sram = &sram35;
                zephyr,flash = &flash0;
                zephyr,code-partition = &slot0_ns_partition;
                zephyr,entropy = &psa_rng;
        };

        aliases {
                led0 = &blue_led_1;
                sw0 = &user_button;
        };

        /delete-node/ memory@20000000;
        /* SRAM3 + SRAM5 (832 kiB + 832kiB)*/
        sram35: memory@200d0000 {
                compatible = "mmio-sram";
                reg = <0x200d0000 DT_SIZE_K(1664)>;
        };

        psa_rng: psa-rng {
                compatible = "zephyr,psa-crypto-rng";
                status = "okay";
        };
};
```

Major Details

zephyr / boards / st / nucleo_u5a5zj_q / tfm / **CMakeLists.txt**

nandojve boards: st: nucleo_u5a5zj_q: Import TF-M code

Code | Blame | 80 lines (67 loc) · 2.33 KB

```
#-------------------------------------------------------------------------------
# Copyright (c) 2020, Arm Limited. All rights reserved.
#
# SPDX-License-Identifier: BSD-3-Clause
#
#-------------------------------------------------------------------------------

set(NUCLEO_U5A5ZJ_Q_DIR ${CMAKE_CURRENT_LIST_DIR})
set(STM_COMMON_DIR ${PLATFORM_DIR}/ext/target/stm/common)

include(${STM_COMMON_DIR}/stm32u5xx/CMakeLists.txt)

#======================== Platform defs ================================#

# Specify the location of platform specific build dependencies.
target_sources(tfm_s
    PRIVATE
        ${STM_COMMON_DIR}/stm32u5xx/Device/Source/startup_stm32u5xx_s.c
)
```

zephyr / boards / st / nucleo_u5a5zj_q / tfm / **config.cmake**

nandojve boards: st: nucleo_u5a5zj_q: Import TF-M code

Code | Blame | 48 lines (44 loc) · 4.41 KB

```
#-------------------------------------------------------------------------------
# Copyright (c) 2020-2023, Arm Limited. All rights reserved.
# Copyright (c) 2021 STMicroelectronics. All rights reserved.
# Copyright (c) 2022 Cypress Semiconductor Corporation (an Infineon c
# or an affiliate of Cypress Semiconductor Corporation. All rights re
#
# SPDX-License-Identifier: BSD-3-Clause
#
#-------------------------------------------------------------------------------

################################## BL2 ##########################################
set(MCUBOOT_IMAGE_NUMBER                2           CACHE STRING
set(BL2_TRAILER_SIZE                    0x9000      CACHE STRING
set(MCUBOOT_ALIGN_VAL                   16          CACHE STRING
set(MCUBOOT_UPGRADE_STRATEGY      "SWAP_USING_SCRATCH" CACHE STRING
set(TFM_PARTITION_PLATFORM              ON          CACHE BOOL
set(MCUBOOT_CONFIRM_IMAGE               ON          CACHE BOOL
set(MCUBOOT_BOOTSTRAP                   ON          CACHE BOOL
set(MCUBOOT_ENC_IMAGES                  ON          CACHE BOOL
set(MCUBOOT_ENCRYPT_RSA                 ON          CACHE BOOL
set(MCUBOOT_DATA_SHARING                ON          CACHE BOOL
cmake_path(NORMAL_PATH MCUBOOT_KEY_S)
cmake_path(NORMAL_PATH MCUBOOT_KEY_NS)
cmake_path(GET MCUBOOT_KEY_S PARENT_PATH MCUBOOT_KEY_PATH)
```

Recap

- TF-M application with Zephyr requires 2 distinct boards
  - In mainline we only can find development boards – not products
  - It is a challenge keep board from TF-M and Zephyr in sync
  - There is no real life examples in mainline

- Adding TF-M board direct inside Zephyr seems to solve this major issue.
  - Easy to sync with atomic operations
  - Easy to develop
  - Clear Kconfigs to be exposed to a CI/CD infrastructure

Thank You

References

1. github.com/zephyrproject-rtos/zephyr/pull/94875 — The base PR of this discussion
2. trustedfirmware-m.readthedocs.io — TF-M Official Documentation
3. trustedfirmware.org — TF-M Technical Overview
4. st.com — UM2851: Getting Started with STM32CubeU5 TF-M
5. review.trustedfirmware.org — NS SRAM 1664 KiB fix (#45951)

Gerson Fernando Budke | Leica Geosystems AG
gerson.budke@leica-geosystems.com
nandojve@gmail.com