Migrating from STM32MP15x to STM32MP25x MPUs

Introduction

The STM32MP251, STM32MP253, STM32MP255 and STM32MP257 devices are part of the STM32 Arm® Cortex® microprocessors (MPUs) subclass; they all feature a Cortex-M33, a Cortex-M0+, and depending on the part number, either a single-core or a dual-core Cortex-A35 and a graphics processing unit (GPU)/neural processing unit (NPU).

These devices are referred to as STM32MP25x in this document.

The Cortex-M33 inside the STM32MP25x devices is mostly backward compatible (for STM32Cube package) with the STM32MP15x devices. This compatibility allows the easy migration from an STM32MP15x device design to a similar STM32MP25x devices and to benefit from their significantly higher performances and security as well as advanced peripherals without adding any additional complexity. Similarly than to the STM32MP15x devices, the STM32MP25x high performance Cortex-A35 runs open operating systems like Linux®, which provides rich connectivity and the support of a software community. System configuration migration between different hardware using different device trees is supported with OpenSTLinux.

This application note provides information to facilitate the migration from an STM32MP15x design to an STM32MP25x design.

D3 domain (Cortex-M0+) has no equivalent on STM32MP15x devices, so not covered in this document.

New STM32MP25x features not present on STM32MP15x are not fully listed in this document as goal is migrating existing solution from previous product range. For whole coverage of STM32MP25x features, refer to STM32MP25x reference documents listed in Section 1.1: Reference documents.

Table 1. Applicable products

Reference	Product
STM32MP15x	STM32MP151A, STM32MP151C, STM32MP151D, STM32MP151F, STM32MP153A, STM32MP153C, STM32MP153D, STM32MP153F, STM32MP157A, STM32MP157C, STM32MP157D, STM32MP157F
STM32MP25x	STM32MP251A, STM32MP251C, STM32MP251D, STM32MP251F, STM32MP253A, STM32MP253C, STM32MP253D, STM32MP253F, STM32MP255A, STM32MP255C, STM32MP255D, STM32MP255F, STM32MP257A, STM32MP257C, STM32MP257D, STM32MP257F

1 General information

This document applies to the STM32MP15x and STM32MP25x Arm®-based microprocesor.

Note: Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

1.1 Reference documents

Table 2. Reference documents

Number	Reference	Title
STM32MP15x		
[1]	RM0436	STM32MP157 reference manual
[2]	RM0442	STM32MP153 reference manual
[3]	RM0441	STM32MP151 reference manual
[4]	DS12505	STM32MP157C/F datasheet
[5]	DS12504	STM32MP157A/D datasheet
[6]	DS12503	STM32MP153C/F datasheet
[7]	DS12502	STM32MP153A/D datasheet
[8]	DS12500	STM32MP151A/D datasheet
[9]	DS12501	STM32MP151C/F datasheet
[10]	AN5031	Getting started with STM32MP15x lines hardware development
STM32MP25x		
[11]	RM0457	STM32MP25x reference manual
[12]	DS14285	STM32MP25xA/D datasheet
[13]	DS14284	STM32MP25xC/F datasheet
[14]	AN5489	Getting started with STM32MP25x lines hardware development

1.2 Glossary

Table 3. **Glossary**

Term	Definition
ADC	Analogue to digital converter
CPU	Central processing unit
DAC	Digital to analog converter
DDR	Double density random access memory
GPU	Graphics processing unit
IC	Integrated circuit
MCU	Microcontroller unit
MPU	Microprocessor unit
NPU	Neural processing unit
OSTL	OpenSTLinux. Linux distribution based on the OpenEmbedded build framework
TSN	Time sensitive networking defined by IEEE 802.1

2 STM32MP15x lines overview

The STM32MP15x devices are part of the STM32MP1 series. Depending on the device part number, the system includes a Cortex-M4 and either a single-core or a dual-core Cortex-A7and a GPU.

The full featured system are listed in the table below and partitioned in:

- One MPU subsystem: dual Cortex-A7 with L2 cache
- One microprocessor (MCU) subsystem: Cortex-M4 with associated peripherals clocked according to central processor unit (CPU) activity

Table 4. Configuration of the lines of the STM32MP15x lines

Devices	Reference manual	Cortex-A7 configuration	Cortex-M4	GPU	DSI	FDCAN	Ethernet port
STM32MP151	[3]	Single-core	Yes	No	No	No	1
STM32MP153	[2]	Dual-core	Yes	No	No	Yes	1
STM32MP157	[1]	Dual-core	Yes	Yes	Yes	Yes	1

3 STM32MP25x lines overview

The STM32MP25x devices are part of the STM32MP2 series devices. Depending on the device part number, the system includes a Cortex-M33 and either a single-core or a dual-core Cortex-A35 and a GPU/NPU.

The full featured system is described in the table below, and is partitioned as listed below:

- One MPU subsystem: dual Cortex-A35 with L2 cache
- One MCU subsystem: Cortex-M35 with associated peripherals clocked according to CPU activity

Table 5. Configuration of the lines of the STM32MP25x lines

Lines	Reference manual	Cortex-A35 configuration	Cortex-M33	GPU/NPU(1)	DSI	FDCAN	Ethernet ports	LVDS
STM32MP251	[11]	Single-core	Yes	No	No	No	1	No
STM32MP253		Dual-core	Yes	No	No	Yes	2	No
STM32MP255		Dual-core	Yes	Yes	Yes	Yes	2	Yes
STM32MP257		Dual-core	Yes	Yes	Yes	Yes	3(2)	Yes

1. *This includes hardware video encode/decode*
2. *With a two port TSN compliant switch*

The STM32MP25x lines offer extra performance compared to the STM32MP15x devices.
The STM32MP25x devices include similar set of peripherals, some with advanced features and higher performances compared to the STM32MP15x devices such as:

Processing

- Single or dual core Arm Cortex-A35 1200 MHz 32/64-bit processor:
  - L1 and L2 caches
  - 1500 MHz frequency overdrive
- 3D graphic processing unit (GPU) running at 800 MHz
  - 900 MHz frequency overdrive
- AI neural processing unit (NPU) running at 800 MHz
  - 900 MHz frequency overdrive
- Hardware video encoder/decoder
- Arm Cortex-M33 400 MHz processor
  - L1 caches
- Arm Cortex-M0+ smart run domain
- HPDMA + LPDMA
- External LPDDR4/DDR4 16/32-bits 1200 MHz or DDR3L 16/32-bits 1066 MHz
- High-performance timers and low-power timers
- Digital temperature sensor

Connectivity

- USB2.0 high-speed host (up to 480 Mbits/s) with embedded PHY
- USB3.0 SuperSpeed dual role (up to 5 Gbits/s) with embedded PHYs
- USB Type-C power delivery
- PCI express (up to 5 Gbits/s) with embedded PHY
- 3 x SDMMC/SDIO
- 2 x OCTOSPI
- FMC for PSRAM or SLC NAND-flash
- USART, UART

- SPI/I²S
- I²C, I3C
- 3 x (TT)FDCAN 2.0
- 3 x Gigabit Ethernet with TSN, IEEE 1588 support
- CSI-2 at 2 x 2.5 Gbps with lite-ISP
- DSI at 4 x 2.5 Gbps
- LVDS dual link 4 x 1.1 Gbps

Audio

- 4 x SAI
- 8 x MDF
- 1 x ADF
- 4 x SPDIF-RX

Other

- Resource isolation framework
- 16- and 32-bit timers
- 3 x 12-bit ADC (5 MSPS) with voltage reference buffer (VREFBUF)
- Tj: -40°C to 125°C

This migration guide covers the migration from STM32MP15x devices towards STM32MP25x devices. This document only the features that are common to the STM32MP25x and the STM32MP15x devices. For more detailed information on the STM32MP25x devices, refer the appropriate reference manuals and datasheets.

There are some features available on STM32MP15x lines that are not present in the STM32MP25x devices. These features are listed below:

- HDMI-CEC
- MDIOS
- DAC

Security concept is different/enhanced, with different blocks names used.

4 Hardware migration

There is no compatible package between STM32MP15x devices and STM32MP25x devices, but a good candidate for migration towards an STM32MP15x device can be chosen by considering the following criteria:

- GPIO: equivalent number of available GPIOs. Precise count should be done for each application use case.
- Size: the package size (from 10×10 to 18×18)
- PCB: the PCB technology cost (TFBGA pitch 0.5 or LFBGA pitch 0.8)
- DDR: the DDR bus width (16- or 32-bit) which is linked to the maximum Cortex-A and GPU performances.

The table below presents a cross reference to assist the user to choose the closest migration candidate in number of GPIO, but also according to some different criteria priority.

Table 6. Cross-selector based on GPIO count, ball pitch and package size

Product	GPIO	Package	Size (mm)	Ball pitch (mm)	DDR bus width	Priority	Closest reference	GPIO	Package	Size (mm)	Ball pitch (mm)	DDR bus width
STM32MP15xxAA	176	LFBGA448	18x18	0.8	32	-	STM32MP25xxAI	172	VFBGA436	18x18	0.8	32
STM32MP15xxAB	98	LFBGA354	16x16	0.8	16	PCB	STM32MP25xxAI	172	VFBGA436	18x18	0.8	32
						DDR	STM32MP25xxAL	144	VFBGA361	10x10	0.5	16
STM32MP15xxAC	148	TFBGA361	12x12	0.5	32	PCB	STM32MP25xxAK	144	VFBGA424	14x14	0.5	32
						DDR	STM32MP25xxAL	144	VFBGA361	10x10	0.5	16
STM32MP15xxAD	98	TFBGA257	10x10	0.5	16	-	STM32MP25xxAL	144	VFBGA361	10x10	0.5	16

4.1 Power supply aspects

To ensure better performance and better power consumption, STM32MP25x lines provide separated supply voltage domain for Cortex-A35 as well as for GPU/NPU.

When the activity requires or permits, these supplies could be overdriven to increase performances, lowered to reduce leakage or even shutdown.

The STM32MP15x which recommends the use of the STPMIC1 power supply management IC, however the STM32MP25x requires to use the STPMIC2.

The power supply and power supply mode comparisons are given in Table 7 and Table 8.

Table 7. Power supply comparison between STM32MP15x and STM32MP25x lines

STM32MP15x lines		STM32MP25x lines		Comments
Supply Name	Typical	Supply Name	Typical	
V_{DD} and V_{DD_DSI}	1.8 / 3.3V	V_{DD}	1.8 / 3.3V	-
		V_{DDIO1}		Optional domain for SD-Card on SDMMC1
		V_{DDIO2}		Optional domain for eMMC on SDMMC2
		V_{DDIO3}		Optional domain for OCTOSPI port1
		V_{DDIO4}		Optional domain for OCTOSPI port2
V_{DDCORE}	1.2V(1)	V_{DDCORE}	0.82V(2)	Also VDDDSI/LVDS/CSI/COMBOPHY/PCIECLK
		V_{DDCPU}	0.8V(1)	For Cortex-A35. New in STM32MP25x lines
		V_{DDGPU}	0.8V(3)	For GPU/NPU. New in STM32MP25x lines
V_{DDQ_DDR}	-	V_{DDQDDR}	-	Different ranges to support available DDR
V_{DD_ANA}	1.8 / 3.3V	$V_{DDA18AON}$	1.8V	New in STM32MP25x lines
V_{DD_PLL}		$V_{DDA18PLL}$	1.8V	Also VDDA18DDR/USB/DSI/LVDS/CSI/COMBOPHY
V_{DDA1V1_REG}	1.1V(4)	-	-	No equivalence (different USB PHY)
V_{DD1V2_DSI}	1.2V(4)	-	-	No equivalence (different DSI PHY)
V_{DDA1V8_REG}	1.8V(4)	-	-	No equivalence (different USB PHY)
V_{DDA1V8_DSI}	1.8V	-	-	No equivalence (different DSI PHY)
V_{DD3V3_USB}	3.3V	$V_{DD33USB}$	3.3V	-
		$V_{DD33UCPD}$		
V_{DDA}	1.8 / 3.3V	$V_{DDA18ADC}$	1.8V	ADC is 1.8V only in STM32MP25x
V_{BAT}	3V	V_{BAT}	3V	-
-	-	V_{08CAP}	0.8V(4)	New in STM32MP25x lines

1. *Supports lowering and overdrive*
2. *Supports lowering*
3. *Supports overdrive*
4. *Internal LDO*

Table 8. System power modes comparison between STM32MP15x and STM32MP25x lines

STM32MP15x lines					STM32MP25x lines								
Mode	V_{DD}		V_{DDCORE}		Mode	V_{DD}		V_{DDCORE}		V_{DDCPU}		V_{DDGPU}	
	State	Control	State	Control		State	Control	State	Control	State	Control	State	Control
Run	ON	-	Overdrive	HW/SW(2)	Run1	ON	-	ON	-	Overdrive	HW/SW(2)	Overdrive or ON or OFF	HW/SW(2)
			ON	-						ON	-		
-	-	-	-	-	Run2(1)					OFF	PWR_CPU_ON		
Stop	ON	-	ON	-	Stop1	ON	-	ON	-	Overdrive	HW / SW(2)	Overdrive or ON or OFF	HW/SW(2)
										ON	-		
-	-	-	-	-	Stop2(1)					OFF	PWR_CPU_ON		
LP-Stop	ON	-	ON	-	LP-Stop1	ON	-	ON	-	Overdrive	HW / SW(2)	Overdrive or ON or OFF	HW/SW(2)
										ON	-		
-	-	-	-	-	LP-Stop2(1)					OFF	PWR_CPU_ON		
LPLV-Stop	ON	-	Lowered	PWR_ON or PWR_LP	LPLV-Stop1	ON	-	Lowered	PWR_ON or PWR_LP	Lowered	PWR_CPU_ON or PWR_LP	OFF or Lowered	PWR_CPU_ON or PWR_LP or HW/SW(2)
-	-	-	-	-	LPLV-Stop2(1)					OFF	PWR_CPU_ON		
Standby	ON	-	OFF	PWR_ON	Standby1	ON	-	OFF	PWR_ON	OFF	PWR_CPU_ON	OFF	PWR_CPU_ON
					Standby2								
V_{BAT}	OFF using HW or SW(2)				V_{BAT1}	OFF using HW or SW(2)							
					V_{BAT2}								

1. *STM32MP25x modes supported with STPMIC2 (new modes not available on STM32MP15x lines)*
2. *For example. GPIO or I^2C command to STPMIC*

The same power concept used on the STM32MP15x devices can be reused on STM32MP25x by having same control for V_{DDCPU}, V_{DDCORE}, and V_{DDGPU} regulator modules, only use PWR_ON (and PWR_LP), with potential overdrive handled by GPIO or direct I^2C commands to STPMIC2.

With this simplified use, PWR_CPU_ON is not used, behavior versus low-power modes is similar to the STM32MP15x. In that case, the STM32MP25x power optimized modes such as Run2, Stop2, LP-Stop2 and LPLV-Stop2 are not available.

5 Boot modes selection

The STM32MP15x and STM32MP25x devices always start from internal BootROM. Internal BootROM starts based on boot pins and on internal OTP fuses.

- Boot from external Flash with default OTPs:
  - SD-Card (SDMMC1)
  - e•MMC (SDMMC2)
  - SLC-NAND (FMC)
  - Serial NOR-Flash or Serial NAND-Flash. Refer to Table 9

Table 9. Processor serial flash boot interface connection

STM32MP15x lines	STM32MP25x lines
QUADSPI	OCTOSPI

- Boot from UART or USB (device)
  - Used to access the device from STM32CubeProgrammer (STM32CubeProg) for example program the external Flash or internal OTP fuses. Refer to Table 10

Table 10. Processor USB boot interface connection

STM32MP15x lines	STM32MP25x lines
USB OTG on High-Speed PHY port #2	USB3DR on High-Speed PHY #2

Note: Default pins for Boot interface are different between STM32MP15x lines and STM32MP25x lines.
Refer to product datasheet and Application Note "Getting started with STM32MP25x lines hardware development."

Refer to following STM32 wiki articles:

- https://wiki.st.com/stm32mpu/index.php/Boot_chains_overview
- https://wiki.st.com/stm32mpu/index.php/STM32MP15_ROM_code_overview
- https://wiki.st.com/stm32mpu/index.php/STM32MP25_ROM_code_overview

6 Peripherals migration

This section presents a full view of the features and peripheral counts of STM32MP15x and STM32MP25x lines, an analysis of peripheral cross compatibility between STM32MP15x and STM32MP25x lines and a peripheral address mapping snapshot for the concerned products.

Table 11. STM32MP15xxx features and peripheral counts Vs STM32MP25xxx

Products	STM32MP157 line				STM32MP153 line				STM32MP151 line				STM32MP257 line			STM32MP255 line			STM32MP253 line			STM32MP251 line		
	STM32MP157xAD	STM32MP157xAB	STM32MP157xAC	STM32MP157xAA	STM32STM32MP153XAD	STM32MP153XAB	STM32MP153XAC	STM32MP153XAA	STM32MP151XAD	STM32MP151XAB	STM32MP151XAC	STM32MP151XAA	STM32MP257xAL	STM32MP257xAK	STM32MP257xAI	STM32MP255xAL	STM32MP255xAK	STM32MP255xAI	STM32MP253xAL	STM32MP253xAK	STM32MP253xAI	STM32MP251xAL	STM32MP251xAK	STM32MP251xAI
Package	TFBGA257	LFBGA354	TFBGA361	LFBGA448	TFBGA257	LFBGA354	TFBGA361	LFBGA448	LFBGA448	LFBGA354	TFBGA361	LFBGA448	VFBGA361	VFBGA424	VFBGA436	VFBGA361	VFBGA424	VFBGA436	VFBGA361	VFBGA424	VFBGA436	VFBGA361	VFBGA424	VFBGA436
Size (mm)	10x10	16x16	12X12	18X18	10x10	16x16	12X12	18X18	10x10	16x16	12X12	18X18	10X10	14X14	18X18	10X10	14X14	18X18	10X10	14X14	18X18	10X10	14X14	18X18
Ball pitch (mm)	0.5	0.8	0.5	0.8	0.5	0.8	0.5	0.8	0.5	0.8	0.5	0.8	0.5	0.5	0.8	0.5	0.5	0.8	0.5	0.5	0.8	0.5	0.5	0.8
GPIOs	98	98	148	176	98	98	148	176	98	98	148	176	144	144	172	144	144	172	144	144	172	144	144	172
CPU core	Cortex-A7 FPU Neon TrustZone												Cortex-A35 FPU Neon TrustZone											
Multicore	Dual-core								Single-core				Dual-core									Single-core		
Cache sizes	32-Kbyte L1 data cache for each core												32-Kbyte L1 data cache for each core											
	32-Kbyte L1 instruction cache for each core												32-Kbyte L1 instruction cache for each core											
	256-Kbyte level 2 unified coherent cache												512-Kbyte level 2 unified coherent cache											
Frequency	up to 650 MHz												up to 1200 MHz											
Overdrive mode	up to 800 MHz(1)												up to 1500 MHz(2)											

Products	STM32MP157 line				STM32MP153 line				STM32MP151 line				STM32MP257 line			STM32MP255 line			STM32MP253 line			STM32MP251 line		
	STM32MP157xAD	STM32MP157xAB	STM32MP157xAC	STM32MP157xAA	STM32STM32MP153XAD	STM32MP153XAB	STM32MP153XAC	STM32MP153XAA	STM32MP151XAD	STM32MP151XAB	STM32MP151XAC	STM32MP151XAA	STM32MP257xAL	STM32MP257xAK	STM32MP257xAI	STM32MP255xAL	STM32MP255xAK	STM32MP255xAI	STM32MP253xAL	STM32MP253xAK	STM32MP253xAI	STM32MP251xAL	STM32MP251xAK	STM32MP251xAI
GPU for 3D graphics	**Vivante - Open GL ES 2.0**												**Vivante - Open GL ES 3.2.8 - Vulkan 1.2**											
Frequency	up to 533 MHz				-								up to 800 MHz						-					
Overdrive mode	-												up to 900 MHz(2)						-					
MCU core	**Cortex-M4 FPU**												**Cortex-M33 FPU TrustZone**											
Cache size	-												16 Kbytes Data cache											
	-												16 Kbytes Instruction cache											
Frequency	up to 209 MHz												up to 400 MHz											
Embedded SRAM																								
CPU System	256 Kbytes												256 Kbytes (+128 Kbytes if video RAM is not used)						384 Kbytes					
Video RAM	-												128 Kbytes (free for CPU system if video accelerators are not used)						-					
MCU subsystem	384 Kbytes												256 Kbytes (128Kbytes are tamper protected)											
MCU retention	64 Kbytes												128 Kbytes											
Backup	4 Kbytes (tamper protected)												8 Kbytes (tamper protected)											
SDRAM																								
LPDDR 16-bits	Up to 1 Gbyte, single rank												Up to 4 Gbytes, dual rank											
LPDDR 32-bits	-	Up to 1 Gbyte, single rank			-	Up to 1 Gbyte, single rank			-	Up to 1 Gbyte, single rank			-	Up to 4 Gbytes, lockstep		-	Up to 4 Gbytes, lockstep		-	Up to 4 Gbytes, lockstep		-	Up to 4 Gbytes, lockstep	
LPDDR freq.	LPDDR2/3 up to 533 MHz												LPDDR4 up to 1200 MHz											

Products	STM32MP157 line				STM32MP153 line				STM32MP151 line				STM32MP257 line			STM32MP255 line			STM32MP253 line			STM32MP251 line		
	STM32MP157xAD	STM32MP157xAB	STM32MP157xAC	STM32MP157xAA	STM32STM32MP153XAD	STM32MP153XAB	STM32MP153XAC	STM32MP153XAA	STM32MP151XAD	STM32MP151XAB	STM32MP151XAC	STM32MP151XAA	STM32MP257xAL	STM32MP257xAK	STM32MP257xAI	STM32MP255xAL	STM32MP255xAK	STM32MP255xAI	STM32MP253xAL	STM32MP253xAK	STM32MP253xAI	STM32MP251xAL	STM32MP251xAK	STM32MP251xAI
DDR 16-bits	up to 1 Gbyte, single rank												Up to 2 Gbytes dual rank for DDR3L, Up to 4 Gbytes single rank for DDR4											
DDR 32-bits	-		Up to 1 Gbyte, single rank		-		Up to 1 Gbyte, single rank		-		Up to 1 Gbyte, single rank		-	Up to 1 Gbyte, single rank		-	Up to 1 Gbyte, single rank		-	Up to 1 Gbyte, single rank		-	Up to 1 Gbyte, single rank	
DDR freq.	DDR3/3L up to 533 MHz												DDR3L up to 1066 MHz. DDR4 up to 1200 MHz											
On-the-fly encrypt/ decrypt	-												Yes(3)											
Backup registers	128 bytes (32x32-bit, tamper protected)												512 Bytes (128x32-bits, tamper protected)											
Timers																								
General purpose 16 bits	8 x 16-bits + 2 x 32-bits												8 x 16-bits + 4 x 32-bits											
Advanced control	2												3											
Basic	2												2											
Low power	2												5											
Watchdog	3 (independent, independent secure, window)												7 (5 x Independent, 2 x Window)											
Communication interfaces																								
SPI / I^2S	6 / 3 (full-duplex)												8 / 3 (full-duplex)											
I^2C	6												8											
USART and UART	4 + 4												4 + 5											
SAI	4												4											
USB host	USB 2.0 host (USBH), 2 ports, embedded hi-speed PHY												USB 2.0 host (USBH), 1 port, embedded hi-speed PHY											

Products	STM32MP157 line				STM32MP153 line				STM32MP151 line				STM32MP257 line			STM32MP255 line			STM32MP253 line			STM32MP251 line		
	STM32MP157xAD	STM32MP157xAB	STM32MP157xAC	STM32MP157xAA	STM32STM32MP153XAD	STM32MP153XAB	STM32MP153XAC	STM32MP153XAA	STM32MP151XAD	STM32MP151XAB	STM32MP151XAC	STM32MP151XAA	STM32MP257xAL	STM32MP257xAK	STM32MP257xAI	STM32MP255xAL	STM32MP255xAK	STM32MP255xAI	STM32MP253xAL	STM32MP253xAK	STM32MP253xAI	STM32MP251xAL	STM32MP251xAK	STM32MP251xAI
USB OTG	USB 2.0 Host/Device (OTG), 1 port, Embedded Hi-Speed PHY (shared with USBH port 2)												USB 2.0/3.0 dual role (USB3DR), embedded hi-speed and SuperSpeed 5 Gbps PHY											
SPDIFRX	4 inputs												4 inputs											
FDCAN	2 (FDCAN)								-				3 (1 x TT-FDCAN), 10 Kbytes shared buffer									-		
HDMI-CEC	1												-											
SDMMC (SD, SDIO, e•MMC)	3 (8 + 8 + 4 bits)												3 (8 + 8 + 4 bits)											
Serial-flash																								
Interface	2 x QUADSPI												2 x OCTOSPI											
On-the-fly decryption	-												Yes[3]											
FMC memory controller																								
PSRAM	4 × CS, up to 4 × 64 Mbytes												4 x CS, up to 4 x 64 MBytes											
NAND	1 × CS, SLC, BCH4/8												4 x CS, SLC, BCH4/8											
Other interfaces																								
Ethernet	1 port, MII, RMII, GMII, RGMII (GMII, RGMII only on TFBGA361 and LFBGA448 packages)												3 ports, R(G)MII, MII			2 ports, R(G)MII, MII			1 port, R(G)MII, MII					
													TSN Switch with two external ports			-								
LCD-TFT parallel interface	Up to 24-bit data (up to 1366×768 60 fps)												up to 24-bits 150 MHz pixel clock (up to 1080p60)											
Display Serial Interface (DSI)	2 × data lanes 1 GHz each (up to 1366×768 60 fps)				-				-				4 x data lanes 2.5 Gbps each (up to 1536p60)						-					
LVDS display	-												Dual-link of 4 x data lanes 1.1 Gbps each (up to 1536p60)[4]						-					

Products	STM32MP157 line				STM32MP153 line				STM32MP151 line				STM32MP257 line			STM32MP255 line			STM32MP253 line			STM32MP251 line		
	STM32MP157xAD	STM32MP157xAB	STM32MP157xAC	STM32MP157xAA	STM32STM32MP153XAD	STM32MP153XAB	STM32MP153XAC	STM32MP153XAA	STM32MP151XAD	STM32MP151XAB	STM32MP151XAC	STM32MP151XAA	STM32MP257xAL	STM32MP257xAK	STM32MP257xAI	STM32MP255xAL	STM32MP255xAK	STM32MP255xAI	STM32MP253xAL	STM32MP253xAK	STM32MP253xAI	STM32MP251xAL	STM32MP251xAK	STM32MP251xAI
DMA	3 instances, 48 physical channels in total												3 instances, 48 physical channels in total											
Cryptography	DES/TDES, AES-256[5]												DES/TDES, AES-256[3]											
Hash	SHA-256, MD5, HMAC												SHA-1, SHA-2 and SHA-3 (up to 512), MD5, HMAC											
True random number generator	Yes												Yes											
Fuses (one-time programmable)	3072 effective bits												12288 effective bits											
Camera interface																								
Parallel	Up to 14-bit, up to 80 MHz												Up to 16-bits, up to 120 MHz. Path shared with CSI											
CSI-2	-												2 x 4 data lanes 2.5 Gbps each. Path shared with DCMIPP											
ISP	-												Yes, basic ISP											
Analog interfaces																								
DFSFM	8 input channels with 6 filters												8 input channels with 8 filters											
ADC	2 x up to 16-bit ADC												3 x 12-bit ADC											
Number of ADC channels	17		22		17		22		17		22		22	24		22	24		22	24		22	24	
DAC	Yes (12-bits)												-											
Number of DAC channels	2												-											
Operating voltage	1.71 to 3.6 V												1.71 to 1.89 V or 3 to 3.6 V											

Products	STM32MP157 line				STM32MP153 line				STM32MP151 line				STM32MP257 line			STM32MP255 line			STM32MP253 line			STM32MP251 line		
	STM32MP157xAD	STM32MP157xAB	STM32MP157xAC	STM32MP157xAA	STM32STM32MP153XAD	STM32MP153XAB	STM32MP153XAC	STM32MP153XAA	STM32MP151XAD	STM32MP151XAB	STM32MP151XAC	STM32MP151XAA	STM32MP257xAL	STM32MP257xAK	STM32MP257xAI	STM32MP255xAL	STM32MP255xAK	STM32MP255xAI	STM32MP253xAL	STM32MP253xAK	STM32MP253xAI	STM32MP251xAL	STM32MP251xAK	STM32MP251xAI
Junction temperature	−40 to 125 °C (15xA or 15xC) or −40 to 105 °C (15xD or 15xF)												−40 to 125 °C											

1. *Only for STM32MP15xD or STM32MP15xF*
2. *Only for STM32MP25xD or STM32MP25xF*
3. *Only for STM32MP25xC or STM32MP25xF*
4. *Except with VFBGA361 package: Single-link of 4 x data lanes 1.1 Gbps each (up to 1050p60)*
5. *Only for STM32MP15xC or STM32MP15xF*

6.1 STM32 peripherals cross-compatibility

The STM32MP15x devices embed a set of peripherals which can be classified in three groups of compatibility:

- Full: Full backward compatibility is ensured by the STM32CubeMP2 package or Linux driver, although some minor changes are possible in either use code or external hardware.
- Partial: Compatibility could be ensured with some modification in the user code or external hardware due to major peripheral version changes.
- None: The feature does not exist in the product or no compatibility is possible.

Table 12. STM32MP15x existing peripheral compatibility with STM32MP25x

Peripheral	Compatibility	Comments
ARM_CortexM4	Partial	Cortex-M33 offer some compatibility with Cortex-M4. Refer to ARMv8-M Architecture Reference Manual and ARMv7-M Architecture Reference Manual from Arm
ARM_CortexA7	Partial	Cortex-A35 offer some compatibility with Cortex-A7. Refer to ARMv8-A Architecture Reference Manual and ARMv7-A Architecture Reference Manual from Arm
TZC	None	Major upgrade in security concepts. Equivalent functionality now in RIF
ADC	Partial	Different ADC, similar features. 12-bits 1.8V only, 3 instances
BSEC	Partial	Major upgrade in security concepts
DCMI	Full	Major version increase
FDCAM	Full	-
CRC	Full	-
CRYP	Full	SAES key sharing added in STM32MP25x devices
DAC	None	No DAC support on STM32MP25x devices
DDRCTRL	None	Major upgrade. Support of LPDDR4 and DDR4
DDRPERFM	None	Major upgrade. Support of LPDDR4 and DDR4
DFSDM	Partial	Replaced by MDF
DMAMUX	Partial	Functionality embedded in HPDMA
DMA	None	New HPDMA and LPDMA IPs
MDMA	None	New HPDMA and LPDMA IPs
DSI	Partial	Major upgrade
ETH	Partial	Major upgrade (TSN support)
MDIOS	None	No MDIOS support on STM32MP25x devices
FMC	Full	Major version increase
TIM	Full	Major version increase
GPU	Partial	Different GPU, but similar API (Open GL ES 3.1 instead of Open GL ES 2.0)
HASH	Full	-
HDMI-CEC	None	No HDMI-CEC support on STM32MP25x devices
HSEM	Full	-
I^2C	Full	Major version increase
LTDC	Full	-
LPTIM	Full	-
IPCC	Full	-
QUADSPI	Partial	Replaced by OCTOSPI
RNG	Full	-
RTC	Full	-

Peripheral	Compatibility	Comments
SAI	Full	-
SDMMC	Full	-
SPDIFRX	Full	-
SPI/I2S	Full	Major version increase
ETZPC	None	Major upgrade in security concepts. Equivalent functionality now in RIF
UART/USART	Full	Major version increase. Increased FIFO size.
OTG	None	Replaced by USB3DR
USBH	Partial	Only one port in STM32MP25x devices. Still EHCI/OHCI compliant.
USBPHYC	None	Different IP. Similar functionalities
IWDG	Full	Major version increase
WWDG	Full	-

6.2 Memory mapping

The peripheral address mapping has been changed in ST32MP25x lines compared to STM32MP15x lines.

The peripheral address mapping is abstracted when using STM32CubeMP2 package for Cortex-M33. So, compatibility in insured.

The internal Cortex-M33 memory address mapping has been changed in STM32MP25x lines, mostly because of Cortex-M33 security concept.

Thanks to instruction and data caches, the Cortex-M33 can efficiently execute or access data from external memories (such as Serial-Flash, HyperFlash, DDR).

Figure 1 and Figure 2 illustrate the STM32MP15x and STM32MP25x line memory maps respectively.

Figure 1. STM32MP157 line memory map

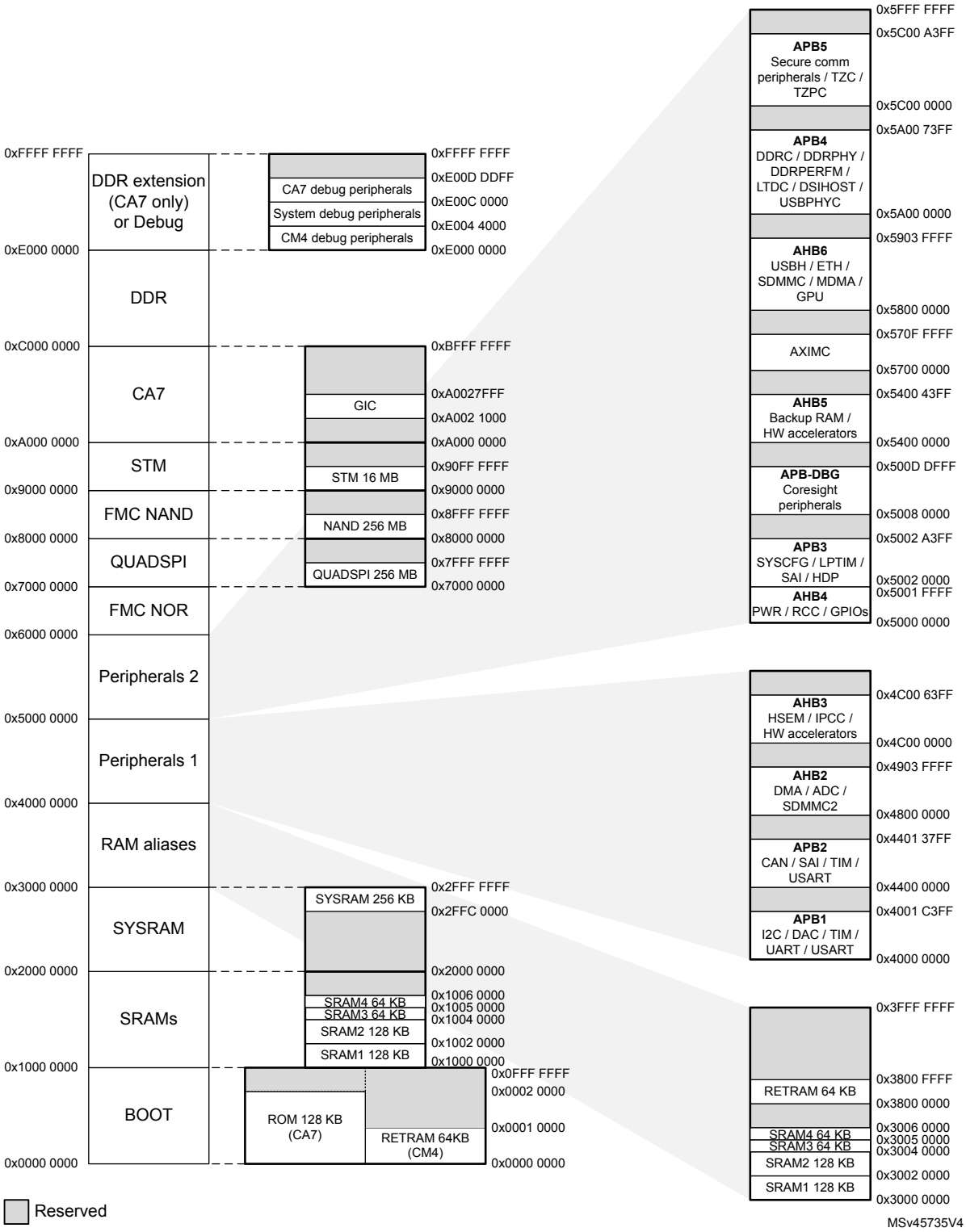
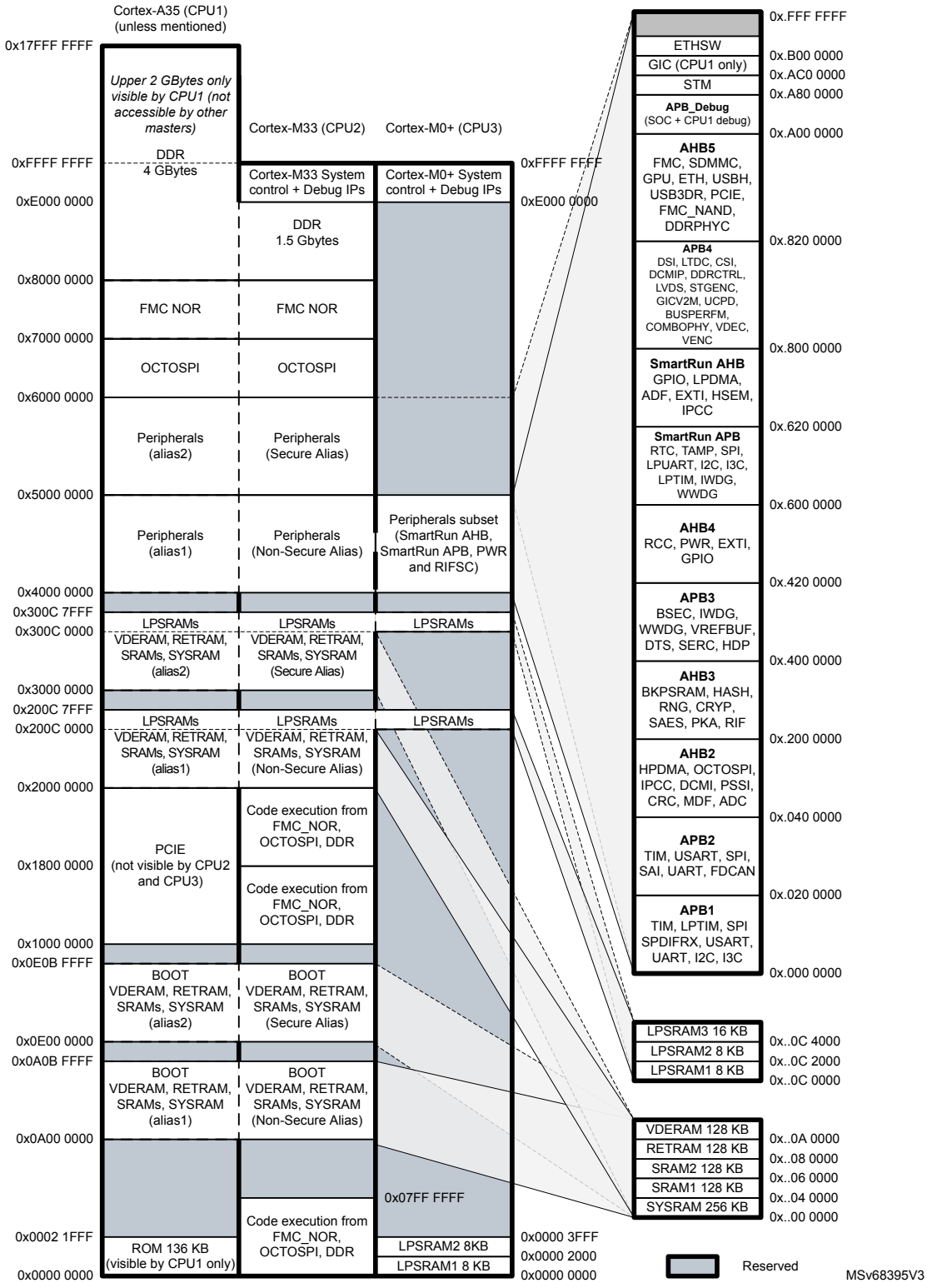


Figure 2. STM32MP257 memory map



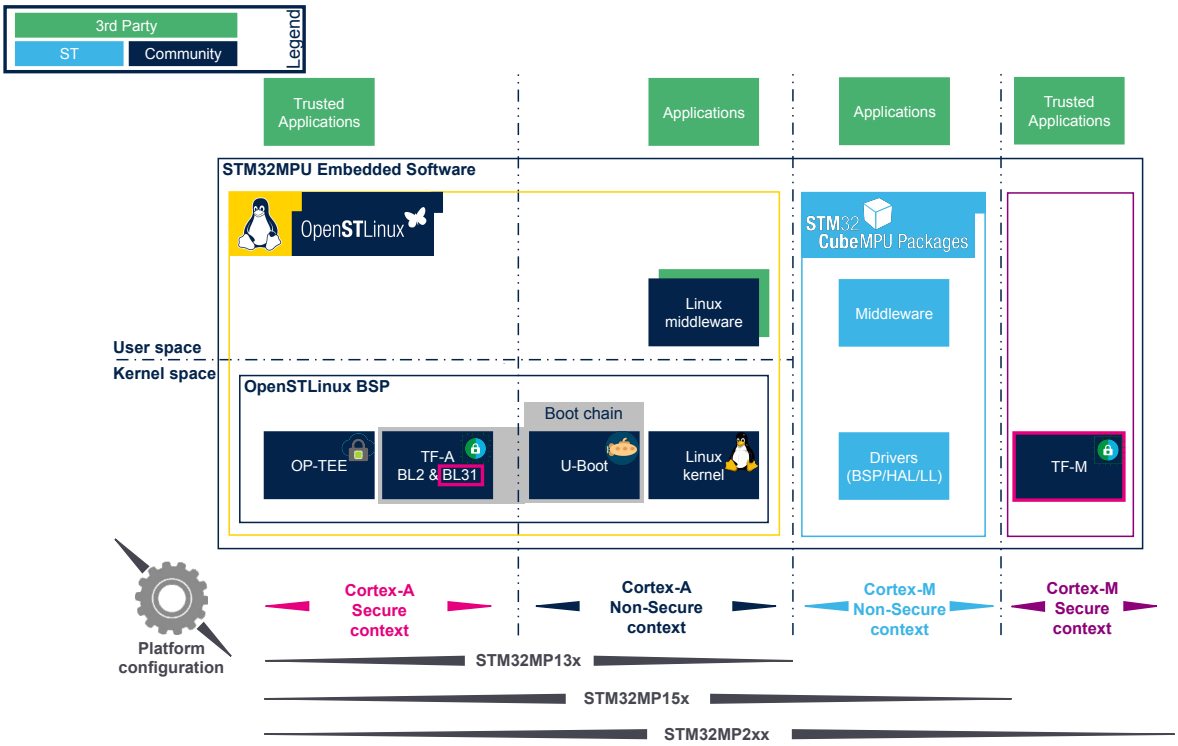
7 Application migration strategy

7.1 STM32MPU embedded software

STM32MP15x and STM32MP25x platforms are supported by the same OpenSTLinux delivery, which automatically adapts the compilation chain and the selected software components according to the machine selection.

The following figure shows all the software components part of STM32MPU embedded software delivery and their applicability per STM32MPU device (STM32MP15x, STM32MP13x and STM32MP25x). New dedicated STM32MP2 series software components are highlighted in pink.

Figure 3. STM32MPU embedded software

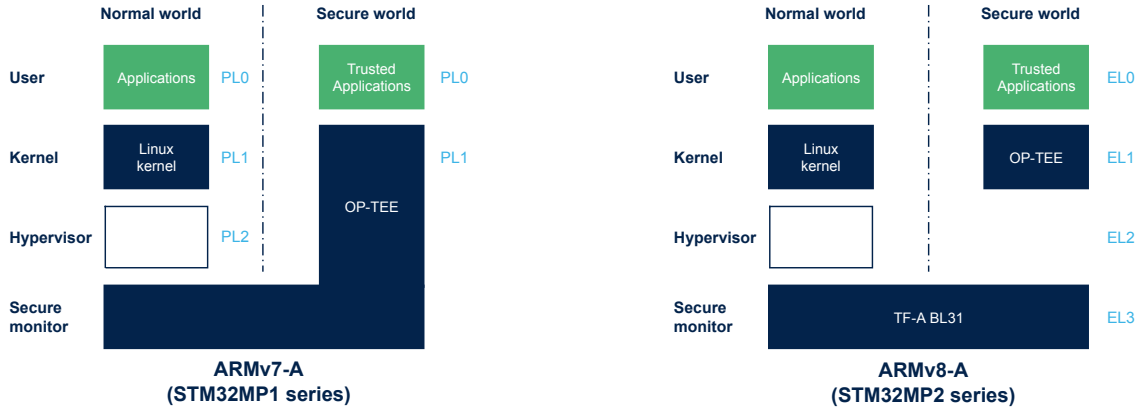


One of the main differences between STM32MP1 series and STM32MP2 series is the Arm Cortex-A architecture are listed below:

- STM32MP1 embeds a Cortex-A7 cluster based on Armv7-A 32-bit architecture
- STM32MP2 embeds a Cortex-A35 cluster based on Armv8-A 64-bit or 32-bit architecture.

Armv8-A architecture introduces the notion of execution levels (EL) which is different from Armv7-A privilege levels (PL).

Figure 4. Armv7-A Vs Armv8-A execution level and associated software



On Armv7-A secure monitor functionalities are managed at secure PL1 level. This is also used for secure OS functionality.

On Armv8-A, the new architecture introduces the EL3 execution level which is dedicated to the secure monitor execution, in charge of switches between secure and non-secure contexts and cluster management (low power modes, CPU plug/unplug and so on). The secure monitor and secure OS are executed in separate environments.

OpenSTLinux supports both Armv7-A and Armv8-A architectures.

- On STM32MP1 series, secure monitor functionality is provided by OP-TEE secure OS. There is one software component providing two set of features:
  - Secure monitor
  - Secure OS
- On STM32MP2 series, Trusted Firmware Cortex-A (TF-A) BL31 provides a secure monitor and OP-TEE provides a secure OS.

Another main difference between STM32MP1 series and STM32MP2 series is the architecture evolution of the associated Cortex-M processor.

- STM32MP15x devices embed a Cortex-M4 processor based on Armv7-M architecture. The processor owns only one non-secure execution context which can run STM32MPU Cube firmware.
- STM32MP25x devices embed a Cortex-M33 processor based on Armv8-M architecture. It owns secure (TrustZone®) and non-secure execution contexts.
  - Cortex-M33 non-secure context can run STM32MPU Cube firmware
  - Cortex-M33 secure context (optional) can run Trusted Firmware Cortex-M (TF-M) secure OS

7.2 Linux application migration

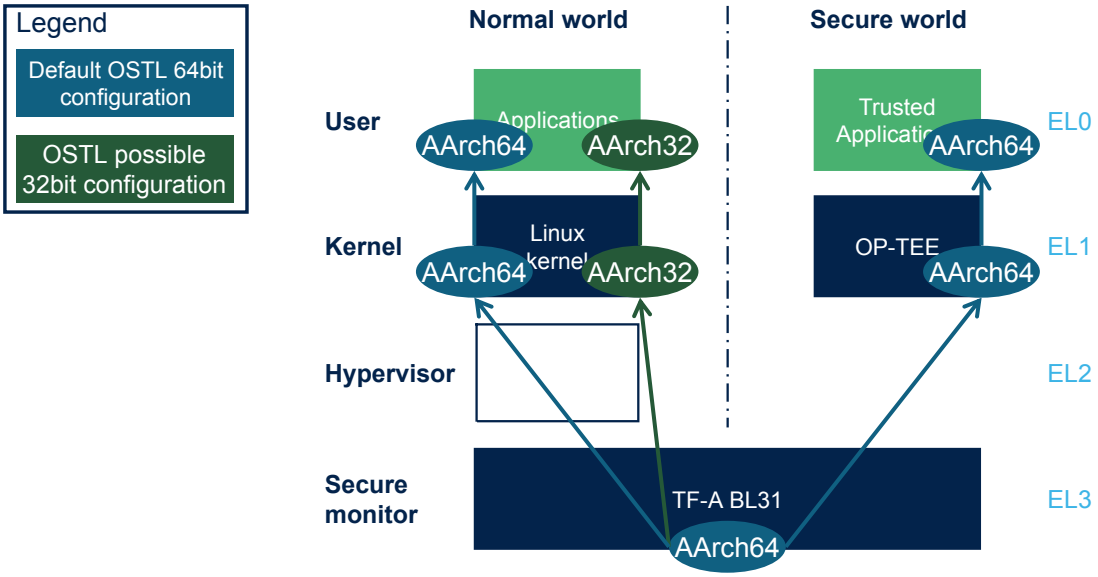
On STM32MP1 series, Linux applications are compiled in Armv7-A 32-bit architecture compliant with Cortex-A7 processor.

On STM32MP2 series, Cortex-A35 processor native instruction set is Armv8-A Aarch64 (64bit). By default, all OSTL software components including Linux applications which are compiled in Aarch64 to be aligned with Cortex-A35 native instructions set and offer up to 4 GB DDR support.

Nevertheless, Cortex-A35 offers the possibility to select at runtime the instruction set (native Aarch64 or Aarch32) for each execution level with the following rules:

- EL3 is always Aarch64
- Transition from Aarch64 ELx to Aarch32 ELy with x > y is possible
- Transition from Aarch32 ELx to Aarch64 ELy with x > y is impossible

Figure 5. **Aarch32 and Aarch64 bits split detail**



STMicroelectronics recommends using the same instruction set for Linux kernel and associated user space applications. Indeed, mixing 32-bit and 64-bit user space applications is possible using "CONFIG_COMPAT" option of the Linux kernel allowing a 32-bit application to run with a 64-bit kernel. But this complexifies the build system (need to generate both 32-bit and 64-bit libraries, toolchain, debug tools and so on) and increases the rootfs size.

STMicroelectronics proposes two options to port an existing application running on STM32MP1 series to STM32MP2 series.

7.2.1 Port applications to 64-bit architecture

Use the OSTL delivery to easily compile an application compilation in a 64-bit architecture. Two different methods are listed below:

- By relying on OSTL build system. STM32MP2 machine automatically selects the Aarch64 compiler and associated compilation options.
- By using OSTL STM32MP25 SDK which provides Aarch64 compiler and associated libraries and compilation options.

Refer to https://wiki.st.com/stm32mpu/wiki/Main_Page for more information on "How to integrate an external software package".

In addition to the 64-bit compilation step, customers must take care of a few items to guarantee the 64-bit compliance. Indeed, the size of many fundamental types have changed and even if well-written C code should not have many dependencies on the size of individual types. It is inevitable that customers will come across some.

The following table sums up the Armv8-A supported data models:

Table 13. Armv8-A supported data models

Type	ILP32(1)	LP64(2)	LLP64(3)	ILP64(1)
char	8	8	8	8
short	16	16	16	16
int	32	32	32	32
long	32	64	32	64
long long	64	64	64	64
size_t	32	64	64	64
pointer	32	64	64	64

1. *ILP means Int, Long and Pointers*
2. *LP means Long and Pointers*
3. *LLP means Long Long and Pointers*

A 32-bit Armv7-A Linux implementation uses a data model equivalent to ILP32 and 64-bit Armv8-A Linux implementations use LP64. The differences are highlighted in the table above.

A best practice is to enable all warnings and errors when re-compiling and to be notified of any warnings and have all issues reported by compiler. Particular attention must be given to type casts in the code which are often the source of errors.

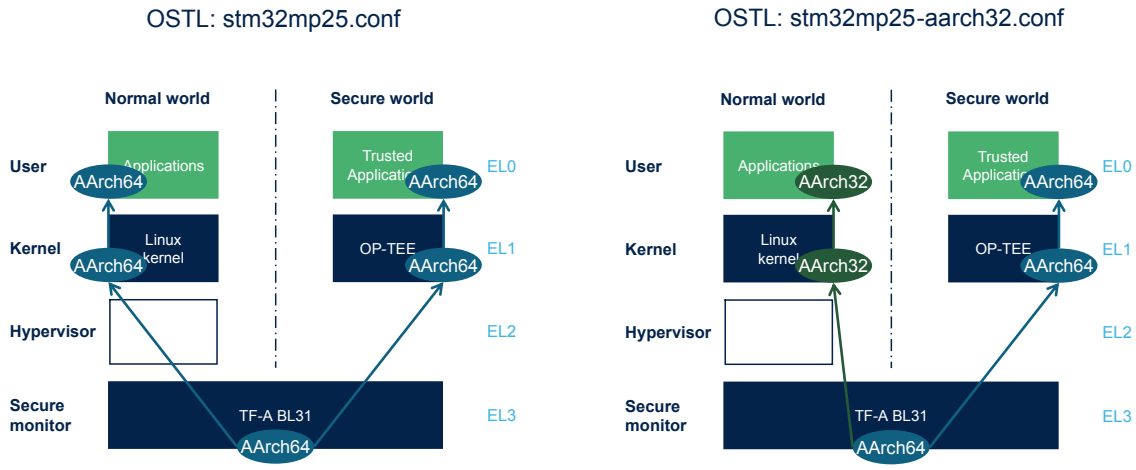
7.2.2 Configure Linux kernel and associated applications in 32-bit configuration

Another solution consists in compiling the Linux kernel and associated applications in 32-bit configuration. To do that, OSTL provides a dedicated build machine named `stm32mp25-aarch32.conf` allowing the Linux kernel and associated RootFS in Aarch32 and firmware (TF-A BL2 & BL31, OP-TEE and U-Boot) to be generated in Aarch64.

Note: stm32mp25-aarch32 OSTL machine may not yet be available at the time of publication of this application note.

The following figure shows the differences between the default stm32mp25 OSTL machine (in Aarch64) and the stm32mp25-aarch32 machine.

Figure 6. Aarch32 versus Aarch64 machine differences



This 32bit machine is not recommended in following cases:

- Products that embed more than 2 GB. Even if large physical address extension (LPAE) allows for addresses with more than 4 GB address space, Cortex-A35 in AArch32 is a 32-bit processor limited to 4 GB of virtual memory. To access the whole DDR, the Linux kernel relies on highmem to temporally virtually map physical memory to access. Use of Linux kernel highmem feature introduces some penalties on global system performances.

- Products that embed between 756 MB and 2 GB of DDR. In such configuration, the whole DDR can be physically addressed by the Cortex-A35 in Aarch32 mode, but its virtual memory is still limited to 4 GB. Use of Linux kernel highmem feature is required, adding penalties on global system performances.

Note: No OSTL developer package is available for this OSTL aarch32 machine.
For further information, refer to https://wiki.st.com/stm32mpu/wiki/Main_Page

7.3 MCU application migration

In the same way as the Cortex-M4 on STM32MP15x devices, the Cortex-M33 on STM32MP25x device acts as a coprocessor from Cortex-A point of view. The Linux kernel (and U-boot) offers the same method for loading the firmware, starting and stopping the Cortex-M33 via the remoteproc framework.

The main difference is that Cortex-M33 owns two execution contexts:

- Cortex-M33 secure context which is dynamically activated or deactivated via Linux remoteproc configuration. This context is used to execute trusted firmware Cortex-M (TF-M) secure OS.
- Cortex-M33 nonsecure context which is always present and equivalent to the Cortex- M4 execution context. This context is used to execute STM32MPU Cube package.

A customer can easily port its STM32MP15x Cortex-M4 application on STM32MP25x Cortex-M33 non secure context using the STM32CubeMX and STM32CubeIDE tools.

Indeed, STM32CubeMX offers a project option to select Cortex-M33 TrustZone® (secure context) activation or deactivation, and then to generate associated Cube project for STM32CubeIDE.

Revision history

Table 14. **Document revision history**

Date	Version	Changes
20-Jun-2024	1	Initial release.

Contents

List of tables

List of figures