STM32MP15 ecosystem errata sheet

STM32MP15 ecosystem errata sheet

Stable: 25.02.2020 - 08:56 / Revision: 25.02.2020 - 08:37

The STM32MP15xx device errata sheet[1] lists the different device limitations and explains the corresponding workarounds (software and/or hardware) if any.

The objective of this article is to explain which of the workarounds described in this errata sheet are implemented in STM32MP15 ecosystem releases.

Here is the legend used in the columns "Status STM32MP15xx Rev.B" and "Status STM32MP15xx Rev.Z" of the table below. It refers to the availability and status of a given errata workaround for the described erratum:

- A = workaround available
- P = partial workaround available
- - = no workaround needed because the limitation is absent in the given revision

Here is the legend used in the column "STM32MP15-Ecosystem-vx.x.x Status" of the table below. It refers to the availability of a workaround for a given STM32MP15 ecosystem release:

- I = workaround **I**mplemented
- P = workaround **P**artialy implemented
- N= workaround **N**ot implemented
- NA= workaround **N**ot **A**pplicable

Wave your mouse over a status in the table to display the corresponding legend.

Previous ecosystem releases [Expand/Collapse]

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
	2.1.1	Memory locations might be accessed speculatively due to instruction fetches when HCR.VM is set	A	A	N	N	Workaround not implemented: STM32MPU Embedded Software distribution does not activate Arm® Cortex®-A7 Hypervisor mode and hence the Virtual Memory second stage of translation. It is customer responsibility to implement the workaround if the Hypervisor mode is used in his/her product.
	2.						Limited impact on the system. Implementation accepted by the community since Linux® kernel v5.3 in

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
Arm® Cortex®-A7 core	1.2	Cache maintenance by set /way operations can execute out of order	A	A	N	N	arch/arm/Kconfig (refer to ARM_ERRATA_814220) so it can already be backported if need be.
	2.1.4	PMU event counter 0x14 does not increment correctly	A	A	N	N	No impact on the system. Minor impact on performance measurement. No workaround provided by Arm for Linux kernel PMU driver.
Arm® Cortex®-M4 core	2.2.1	Interrupted loads to SP can cause erroneous behavior	A	A	N	N	Limitation only on hand-written assembly code. The customer must implement the workaround in the product assembly code.
	2.2.2	VDIV or VSQRT instructions might not complete correctly when very short ISRs are used	A	A	N	N	STM32CubeMP1 Package provided as example. It is customer responsibility to implement one of the proposed workarounds according to the user code and product configuration.
	2.2.3	Store immediate overlapping exception return operation might vector to incorrect interrupt	A	A	N	N	Minor impact on the system. Workaround to be implemented by customer according to the MPU configuration.
	2.3.1	TPIU fails to output sync after the pattern generator is disabled in Normal mode	A	A	N	N	No workaround implemented. No impact on the system since this issue occurs only on the trace port.

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
System	2.3.3	HSE external oscillator required in some LTDC use cases	P	P	I	I	Hardware implementation of the external oscillator connected to the HSE pins available on STM32MP157C-EV1 MB1263 Rev.C (aka "MB1263C") and STM32MP157X-DKX MB1272 Rev.C (aka "MB1272C"). HSE configuration implemented in TF-A fdts/stm32mp157c-ed1.dts for STM32MP157C-EV1 MB1263 and fdts/stm32mp157c-dk1.dts for STM32MP157X-DKX MB1272.
	2.3.4	RCC cannot exit Stop and Low-power Stop modes	A	A	I	I	Implemented in TF-A plat/st/stm32mp1/bl2_plat_setup.c
	2.3.5	Incorrect reset of glitch-free kernel clock switch	P	P	I	I	By default, STPMIC1 performs a VDDCORE reset on NRST activation.
	2.3.6	Limitation of aclk/hclk5/hclk6 to 200 MHz when used as SDMMC1/2 kernel clock	P	P	I	I	Implemented in TF-A fdts with a clock tree that uses a SDMMC1/SDMMC2 kernel clock source different from the aclk/hclk5/hclk6 bus clock.
	2.3.8	eMMC boot timeout too short	A	-	I	I	Rev.B workaround implemented in STM32MP157C-EV1 MB1263 Rev.C (aka "MB1263C") that uses an eMMC that meets the required timing.
	2.	Cortex-M4 cannot use I/O					The examples delivered with

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
	3.9	compensation on Standby mode exit	A	A	I	I	STM32Cube use only IOSPEEDR[1:0] settings 00 and 01.
DDRPHYC	2.4.1	DDRPHYC overconsumption upon reset or Standby mode exit	A	-	P	NA	DDRPHYC is correctly reinitialized by TF-A after reset and Standby mode exit. On Rev.B, the issue is present if Cortex®-M4 standalone wakeup as TF-A is not executed and hence DDRPHYC not reinitialized.
	2.4.2	DDR_CLK jitter out of JEDEC requirement for 32-bit LPDDR2/LPDDR3 at low device Tj	A	A	NA	NA	ST boards use DDR3 instead of LPDDR2/3.
	2.4.3	Data corruption at low device Tj combined with low 32-bit LPDDR2/LPDDR3 I/O supply voltage	A	A	NA	NA	ST boards use DDR3 instead of LPDDR2/3.
DMAMUX	2.6.4	Wrong input DMA request routed upon specific DMAMUX_CxCR register write coinciding with synchronization event	A	A	P	P	Not applicable to OpenSTLinux distribution since DMA Synchronous mode is not used. It is customer responsibility to provide the right DMAMUX signal polarity configuration when calling the HAL_DMAEx_ConfigMuxSync() function provided in STM32CubeMP1 Package Src/stm32mp1xx_hal_dma_ex.c .
QUADSPI	2.7.1	Memory-mapped read of last memory byte fails	P	P	A	A	Implemented in OpenSTLinux distribution drivers/spi/spi-stm32-qspi.c

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
ADC	2.8.1	New context conversion initiated without waiting for trigger when writing new context in ADC_JSQR with JQDIS = 0 and JQM = 0	A	A			Not applicable to Linux stm32-adc driver , since JQDIS = 1.
	2.8.2	Two consecutive context conversions fail when writing new context in ADC_JSQR just after previous context completion with JQDIS = 0 and JQM = 0	P	P			Not applicable to Linux stm32-adc driver , since JQDIS = 1.
	2.8.3	Unexpected regular conversion when two consecutive injected conversions are performed in Dual interleaved mode	A	A			Not applicable to Linux stm32-adc driver , since Dual mode is not used.
	2.8.4	ADC ANA0/ANA1 resolution limited when Gigabit Ethernet is used	P	P	P	P	The customer must implement this workaround by limiting the ADC data resolution in OpenSTLinux distribution device tree configuration or in the STM32 CubeMP1 Package application.
	2.8.5	ADC missing codes in differential 16-bit static acquisition	P	P	P	P	The customer must implement this workaround by limiting the ADC data resolution in OpenSTLinux distribution device tree configuration or in the STM32 CubeMP1 Package application.
							The Linux DAC driver uses only the Normal mode. It never needs to modify the MODE bitfield.

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
DAC	2.9.1	Invalid DAC channel analog output if the DAC channel MODE bitfield is programmed before DAC initialization	A	A	P	P	Both Normal and Sample and hold modes are supported by the HAL drivers. It is up to the user to properly call HAL_DAC_Init before HAL_DAC_ConfigChannel to avoid the issue.
DTS	2.10.1	Mode using PCLK & LSE (REFCLK_SEL = 1) should not be used	P	P	I	I	Implemented in OpenSTLinux distribution drivers/thermal/st/stm_thermal.c
TIM	2.12.1	One-pulse mode trigger not detected in master-slave reset + trigger configuration	P	P	N	N	This workaround is proposed only as a recommendation.
LPTIM	2.13.1	MCU may remain stuck in LPTIM interrupt when entering Stop mode	A	A	N	N	The LPTIM interrupt is not used in Open STLinux distribution. This workaround is not implemented in STM32CubeMP1 Package. It is customer responsibility to implement it in MspDeinit().
	2.13.2	MCU may remain stuck in LPTIM interrupt when clearing event flag	P	P	I	I	The LPTIM interrupt is not used in Open STLinux distribution. It is implemented in STM32CubeMP1 Package Src/stm32mp1xx_hal_lptim.c .
RTC an	2.						

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
dTAMP	14.2	Calendar initialization may fail in case of consecutive INIT mode entry	A	A	I	I	This workaround is implemented in OpenSTLinux distribution drivers/rtc/rtc-stm32.c .
I2C	2.15.1	Wrong data sampling when data setup time (tSU;DAT) is shorter than one I2C kernel clock period	P	P	I	I	This workaround is implemented in TF-A fdts with a clock tree that configures the I2C kernel clock source to a frequency higher than 20 MHz. It is valid for both OpenSTLinux distribution and STM32CubeMP1 Package.
	2.15.2	Spurious bus error detection in master mode	A	A	N	N	In order to get real bus error notifications, this workaround is implemented neither within OpenSTLinux distribution nor within STM32CubeMP1 Package.
	2.15.3	Spurious master transfer upon own slave address match	P	P	NA	NA	The multimaster mode implementation of STM32CubeMP1 Package I2C HAL driver prevents such case from happening.
	2.15.5	Transmission stalled after first byte transfer	A	A	N	N	
	2.17						

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
SPI	.1	Master data transfer stall at system clock much faster than SCK	A	A	I	I	SPI is disabled after each EOT in OpenSTLinux distribution drivers/spi/spi-stm32.c and in STM32CubeMP1 Package Src/stm32mp1xx_hal_spi.c .
	2.17.2	Corrupted CRC return at non-zero UDRDET setting	P	P	N	N	Slave mode and CRC are not supported in OpenSTLinux distribution. This workaround is not implemented in STM32CubeMP1 Package.
	2.17.3	TXP interrupt occurring while SPI disabled	A	A	I	I	This workaround is implemented in OpenSTLinux distribution, drivers/spi/spi-stm32.c ensures that all interrupts are disabled before the SPI is disabled. This workaround is implemented in STM32CubeMP1 Package Src/stm32mp1xx_hal_spi.c .
	2.17.4	Possible corruption of last-received data depending on CRCSIZE setting	A	A	N	N	CRC is not supported in OpenSTLinux distribution.
	2.18.1	Desynchronization under specific condition with edge filtering enabled	A	A	N	N	
	2.18						

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
FDCAN	2.2	Tx FIFO messages inverted under specific buffer usage and priority setting	A	A	N	N	
	2.18.3	DAR mode transmission failure due to lost arbitration	A	A	N	N	
	2.20.2	Rx DMA may fail to recover upon DMA restart following a bus error, with Rx timestamping enabled	A	A	N	N	
	2.20.3	Tx DMA may halt while fetching TSO header under specific conditions	A	A	N	N	
	2.20.4	Spurious receive watchdog timeout interrupt	A	A	N	N	
	2.20.						

Function	Section	Limitation	Status for STM32MP15xx Rev.B	Status for STM32MP15xx Rev.Z	Status for ecosystem release v1.1.0	Status for ecosystem release v1.2.0	Comment about the workaround implemented in STM32MP15 ecosystem
ETH	5	Incorrect flexible PPS output interval under specific conditions	A	A	N	N	
	2.20.6	Packets dropped in RMII 10Mbps mode due to fake dribble and CRC error	A	A	N	N	
	2.20.7	ARP offload function not effective	A	A	I	I	ARP software support is used in OpenST Linux distribution.

Previous ecosystem releases [Expand/Collapse]

References

- STM32MP15 Errata sheets

Non Applicable

Power Management Unit (in STPMIC context) or Performance Monitoring Unit (in Arm Cortex-A context)

Microprocessor Unit

High Speed External oscillator (STM32 clock source)

also known as

former spelling for e•MMC ('e' in italic)

Trusted Firmware for Arm Cortex-A

Doubledata rate (memory domain)

Direct Memory Access

Hardware Abstraction Layer

Analog-to-digital converter. The process of converting a sampled analog signal to a digital code that represents the amplitude of the original signal sample.

Digital-to-analog converter (Electronic circuit that converts a binary number into a continuously varying value.)

Low Speed External oscillator (STM32 clock source)

Microcontroller Unit (MCUs have internal flash memory and are intended to operate with a minimum amount of external support ICs. They commonly are a self-contained, system-on-chip (SoC) designs.)

low-power timer (STM32 specific)

Inter-Integrated Circuit (Bi-directional 2-wire bus standard for efficient inter-IC control.)

Serial Peripheral Interface

End Of Transmission (MIPI® Alliance DSI standard)

Cyclic redundancy check calculation unit

Protocol used by the Internet Protocol, specifically IPv4, to map IP network addresses to the hardware addresses used by a data link protocol (https://en.wikipedia.org/wiki/Address_Resolution_Protocol)