SYSRAM internal memory

Stable: 12.02.2019 - 09:18 / Revision: 07.01.2019 - 17:33

Contents

1 Article purpose

The purpose of this article is to briefly introduce the SYSRAM internal memory and indicate the level of security supported by this memory.

2 Peripheral overview

The STM32MP15 **SYSRAM** is a 256-Kbyte internal memory peripheral. It is physically located near the Arm$^{®}$ Cortex-A to optimize the core performance.

2.1 Features

Refer to the STM32MP15 reference manuals for the complete list of features, and to the software components, introduced below, to see which features are really implemented.

2.2 Security support

The SYSRAM is a **secure** peripheral (under ETZPC TrustZone memory adapter (TZMA)): it can be split into a secure and a non-secure regions with a 4-Kbyte granularity.

3 Peripheral usage and associated software

3.1 Boot time

The STM32MP15 ROM code mainly configures the SYSRAM as a secure peripheral during its execution. The ROM code uses 9 Kbytes located at the beginning of the SYSRAM to store its read and write data. The ROM code stores the boot context in the first 512 bytes of SYSRAM: this boot context contains several information (such as the selected boot device) and pointers to the ROM code services (used for secure boot authentication). The ROM code loads the FSBL just after the boot context, into the remaining 247 Kbytes of SYSRAM, and eventually branches the Cortex®-A7 core 0 execution to this FSBL.

The FSBL code can use the whole SYSRAM, but it must take care not to overwrite the boot context before taking it into account.

3.2 Runtime

3.2.1 Overview

In STMicroelectronics distribution, the SYSRAM runtime mapping is the one reached at the end of the boot. It is consequently fully secure and contains a minimal secure monitor (from TF-A or U-Boot) or a secure OS (like OP-TEE).

You may decide to split the SYSRAM at runtime. In this case:

- set the SYSRAM bottom secure, for a Cortex®-A7 secure monitor (from TF-A or U-Boot) or a secure OS (such as OP-TEE)

and

- set the SYSRAM top non-secure, for instance for using in Linux® as reserved memory

3.2.2 Software frameworks

Domain	Peripheral	Software frameworks			Comment
		Cortex-A7 secure (OP-TEE)	Cortex-A7 non-secure (Linux)	Cortex-M4 (STM32Cube)	
Core/RAM	SYSRAM	TF-A overview	Linux reserved memory		

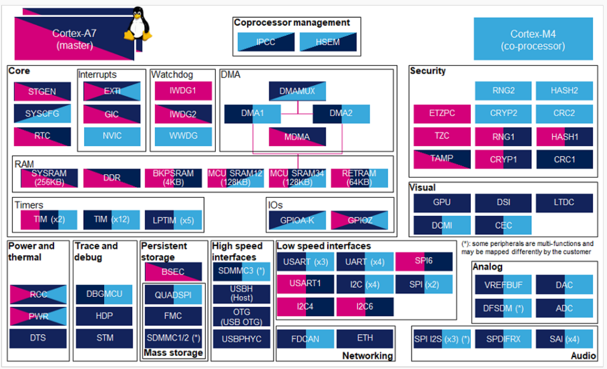
3.2.3 Peripheral configuration

The configuration is applied by the firmware running in the context to which the peripheral is assigned. The configuration can be done alone via the STM32CubeMX tool for all internal peripherals, and then manually completed (particularly for external peripherals), according to the information given in the corresponding software framework article.

3.2.4 Peripheral assignment

Check boxes illustrate the possible peripheral allocations supported by STM32 MPU Embedded Software:

- ☐ means that the peripheral can be assigned (☑) to the given runtime context.
- ✓ is used for system peripherals that cannot be unchecked because they are statically connected in the device.



Internal peripherals

Refer to How to assign an internal peripheral to a runtime context for more information on how to assign peripherals manually or via STM32CubeMX.
The present chapter describes STMicroelectronics recommendations or choice of implementation. Additional possiblities might be described in STM32MP15 reference manuals.

Domain	Peripheral	Runtime allocation				Comment
		Instance	Cortex-A7 secure (OP-TEE)	Cortex-A7 non-secure (Linux)	Cortex-M4 (STM32Cube)	
Core/RAM	SYSRAM	SYSRAM	☐	☐		Shareable (multiple choices supported)

4 References

Operating System

Open Portable Trusted Execution Environment

Random Access Memory