CEC internal peripheral

CEC internal peripheral

Stable: 21.01.2020 - 15:27 / Revision: 21.01.2020 - 15:26

Contents

1 Article purpose 2
2 Peripheral overview 2
2.1 Features 2
2.2 Security support 3
3 Peripheral usage and associated software 3
3.1 Boot time 3
3.2 Runtime 3
3.2.1 Overview 3
3.2.2 Software frameworks 3
3.2.3 Peripheral configuration 3
3.2.4 Peripheral assignment 3
4 How to go further 4
5 References 4

1 Article purpose

The purpose of this article is to:

- briefly introduce the CEC peripheral and its main features
- indicate the level of security supported by this hardware block
- explain how each instance can be allocated to the three runtime contexts and linked to the corresponding software components
- explain, when necessary, how to configure the CEC peripheral.

2 Peripheral overview

The CEC (consumer electronics control) or HDMI-CEC is an STM32 internal peripheral that allows to receive/send messages from/to devices, such as TV or tuner, through a HDMI cable.

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 implemented.

Refer to the STM32 CEC presentation [1] for an overview of the CEC hardware block capabilities.

2.2 Security support

The CEC is a **non-secure** peripheral.

3 Peripheral usage and associated software

3.1 Boot time

The CEC is not used at boot time.

3.2 Runtime

3.2.1 Overview

The CEC internal peripheral can be allocated to:

- the Arm® Cortex®-A7 non-secure core to be controlled in Linux® by the CEC framework

or

- the Arm® Cortex®-M4 to be controlled in STM32Cube MPU Package by STM32Cube CEC driver

Chapter Peripheral assignment describes which peripheral instance can be assigned to which context.

3.2.2 Software frameworks

Domain	Peripheral	Software frameworks			Comment
Cortex-A7 secure (OP-TEE)	Cortex-A7 non-secure (Linux)	Cortex-M4 (STM32Cube)			
Low-speed interface	CEC		CEC framework	CEC HAL driver	

3.2.3 Peripheral configuration

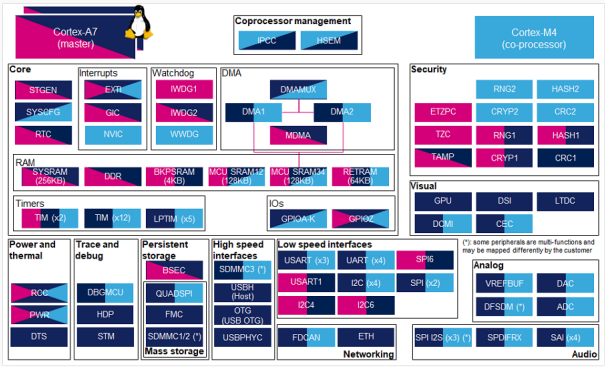
The configuration is applied by the firmware running in the context to which the peripheral is assigned. The configuration can be done alone via the STM32CubeMX tool for all internal peripherals, and then manually completed (particularly for external peripherals), according to the information given in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

3.2.4 Peripheral assignment

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

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

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.



Internal peripherals

Domain	Peripheral	Cortex-A7 secure (OP-TEE)	Cortex-A7 non-secure (Linux)	Cortex-M4 (STM32Cube)	Runtime allocation		Comment
Instance							
Visual	CEC	CEC			â	â	Assignment (single choice)

4 How to go further

Refer to the STM32 CEC application note (AN4066) [2] for a detailed description of the CEC peripheral and applicable use-cases.

Even if this application note is related to STM32 microcontrollers, it also applies to STM32 MPUs.

5 References

- â STM32 CEC presentation

- â STM32 CEC Application Note (AN4066)

Consumer Electronics Control (HDMI standard)

High-Definition Multimedia Interface (HDMI standard)

Microprocessor Unit

Open Portable Trusted Execution Environment