

IWDG internal peripheral

Stable: 11.02.2019 - 13:21 / Revision: 20.12.2018 - 17:30

Contents

1 Peripheral overview	1
1.1 Features	1
1.2 Security support	1
2 Peripheral usage and associated software	2
2.1 Boot time	2
2.2 Runtime	2
2.2.1 Overview	2
2.2.2 Software frameworks	2
2.2.3 Peripheral configuration	2
2.2.4 Peripheral assignment	2

1 Peripheral overview

The **IWDG** peripheral is a watchdog unit that can be used to protect application frameworks running on Cortex-A7 from endless loops. This peripheral supports an **independent** clocking source in order to be able to continue running even when the rest of the system is in **low power mode** (STOP, STANDBY). Another important feature of this block is the **early interrupt** feature that allows to trigger an interrupt at a given power supply threshold before reaching the final reset: this gives the opportunity to run a recovery mechanism that will try to revive the system with minimum impact.

1.1 Features

Refer to [STM32MP15 reference manuals](#) for the complete list of features, and to the software components, introduced below, to see which features are implemented.

1.2 Security support

IWDG1 is **secure-aware** (under [ETZPC](#) control).
IWDG2 is **non-secure**.

2 Peripheral usage and associated software

2.1 Boot time

Pay attention to the fact that IWDG can be configured to be **automatically active** at startup (without any software intervention) via [BSEC](#). When this is the case, the watchdog is anyway frozen during [ROM code](#) execution but it will start to decrement its counter as soon as the ROM code is left so it is important to reload the watchdog from the [boot chain](#) in this case. This behavior is implemented for **IWDG2 only** in STMicroelectronics distribution via the [trusted boot chain](#) only.

Notice also that [BSEC](#) features some freeze bits that allow to **freeze IWDG** during platform STOP and STANDBY [low power](#) periods, avoiding to have to wake up (via [RTC](#)) for the only purpose of reloading the watchdog.

2.2 Runtime

2.2.1 Overview

IWDG1 can be allocated to the Cortex-A7 secure to be used in the secure context by the customer application: this instance is not supported in STMicroelectronics distribution.

IWDG2 can be allocated to the Cortex-A7 non-secure to be used with Linux [watchdog](#) framework. In this configuration, the secure monitor (from [OP-TEE](#) -if present- or [TF-A](#)) is able to receive IWDG early interrupts that can be used in a tentative to reset the Cortex-A7 without interfering with Cortex-M4 execution.

2.2.2 Software frameworks

Do mai n	Peri phe ral	Software frameworks			Comment
		Cortex-A7 secure (OP-TEE)	Cortex-A7 non-secure (Linux)	Cortex-M4 (STM32Cube)	
Core /Wat chd og	IWD G	TF-A	Linux watchdog framework		

2.2.3 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.

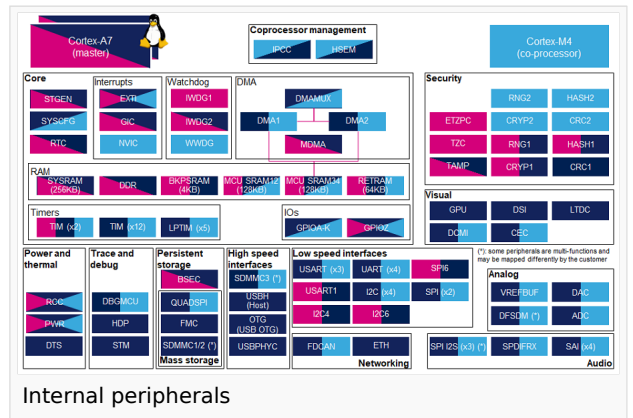
2.2.4 Peripheral assignment

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.

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 possibilities might be described in **STM32MP15 reference manuals**.



Do ma in	Pe ri ph er al	Runtime allocation			Comme nt
		Instance	Cortex-A7 secure (OP-TEE)	Cortex-A7 non-secure (Linux)	
Core /W atc hd og	IWDG	IWDG1	<input type="checkbox"/>		Shared (none or both): <ul style="list-style-type: none"> ■ Cortex-A7 non-secure for reload ■ Cortex-A7 secure for early interrupt handling
		IWDG2	<input type="checkbox"/>	<input type="checkbox"/>	

Independent Watchdog

Read Only Memory

Open Portable Trusted Execution Environment