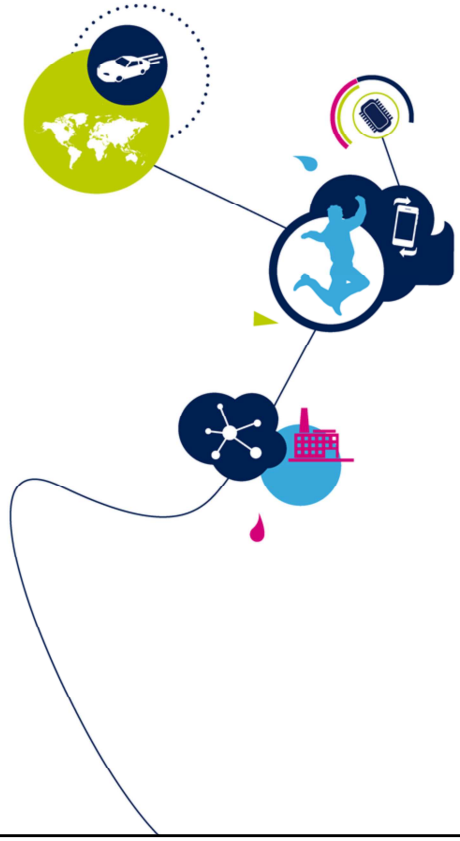
STM32H7 – IWDG

Independent watchdog

Revision 1.0



life.augmented

Hello and welcome to this presentation of the STM32 independent watchdog. It covers the main features of this peripheral which can be used either as a watchdog to reset the device when a problem occurs, or as a free-running timer for application timeout management.

Overview

- Serves to detect and resolve malfunctions due to software failures:
  - Triggers a system reset when it is not refreshed within the expected time window
  - Always active even if the main clock fails
  - Cannot be disabled once activated, and needs to be refreshed

Application benefits

- Totally independent process outside the main application
- Selectable hardware or software start
- Selectable low-power freeze in Standby or Stop modes

life.augmented

The independent watchdog is used to detect and resolve malfunctions due to software failures.

It triggers a reset sequence when it is not refreshed within the expected time-window.
Since its clock is an independent 32-kHz low-speed internal RC oscillator (LSI), it remains active even if the main clock fails.
Once enabled, it forces the activation of the low-speed internal oscillator, and it can only be disabled by a reset.

One of the main benefits for applications is its ability to run independently from the main clock.

Key features

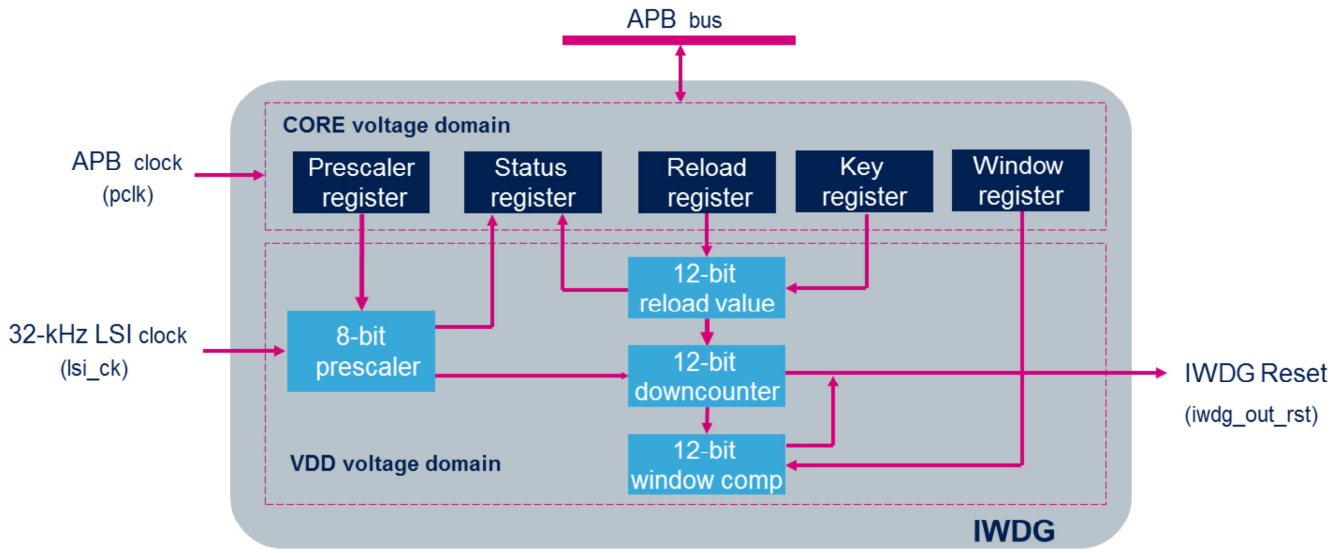
- IWDG main features
  - Programmable timeout range from 125 µs to 32.8 seconds
  - Programmable time-window width
  - Clocked from an independent RC oscillator (LSI)
  - Generates a reset when:
    - The timeout value is reached
    - The refresh occurs outside the window
  - Can be frozen in Debug, Stop or Standby mode
  - Can be configured to be automatically enabled

life.augmented

The independent watchdog offers a wide range of timeout values: from 125 microseconds to 32 seconds.
It is clocked by a 32-kHz RC oscillator which cannot be disabled when the independent watchdog is enabled.
It generates a reset when the programmed timeout value elapses, or when a watchdog refresh occurs outside a programmed time-window.
This window feature is optional, and not present in all independent watchdogs.

It is possible to enable automatically the independent watchdog after a system reset.
It is possible to define the behavior of the independent watchdog in Debug, Stop or Standby mode.

Block diagram



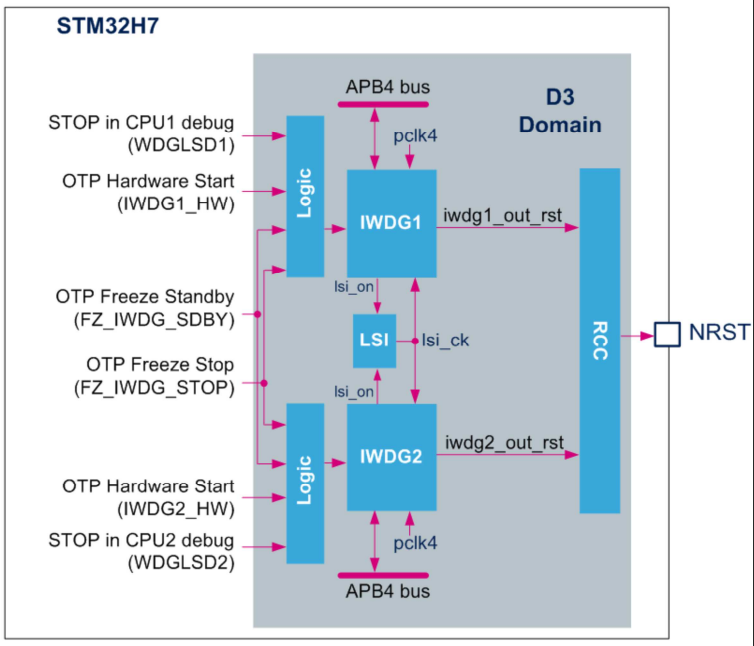
The independent watchdog registers are located in the CORE voltage domain while its functions are in the VDD voltage domain.
Two clocks are needed:

- The APB clock is required in order to access registers
- The LSI clock is required for the functional part of the watchdog

This architecture allows the independent watchdog to work even in Stop and Standby modes.
A programmable 8-bit prescaler is used to divide the LSI oscillator frequency.
The 12-bit downcounter defines the timeout value.

WatchDog Integration

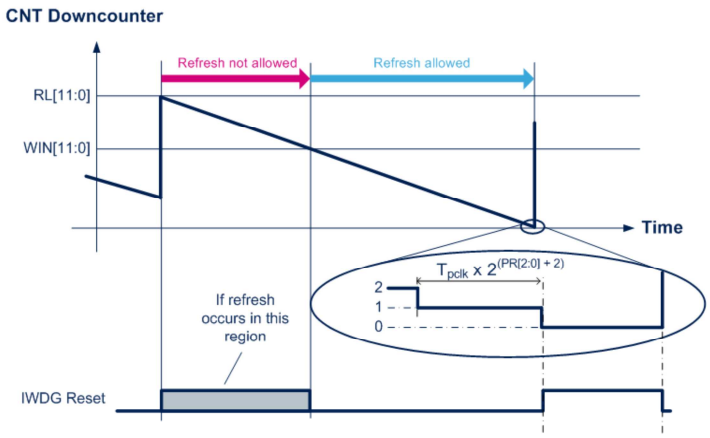
- IWDG1 and IWDG2 are located into the D3 Domain
- IWDG1 is dedicated to CPU1
- IWDG2 is dedicated to CPU2
- Can generate a system reset



The STM32H7 microcontroller includes two independent watchdogs (IWDG).
IWDG1 is dedicated to CPU1 usage, and IWDG2 is dedicated to CPU2 usage.
For each independent watchdog, it is possible to select the hardware or software start via option bytes.
For each independent watchdog, it is possible to select if the watchdog will freeze when the associated CPU is in Debug (core halted) mode. Please refer to the Microcontroller debug unit (DBGMCU) description for details.
The control of the behavior in Stop or Standby mode is common for both independent watchdogs.
When ‘OTP Freeze Stop’ is activated, the independent watchdogs are frozen when the system is in Stop or Standby mode.
When ‘OTP Freeze Standby’ is activated, the

independent watchdogs are frozen when the system is in Standby mode.
Both watchdogs are connected to the APB4 bus of the D3 domain.
Both watchdogs can perform a system reset handled by the RCC block.

IWDG functional description



If the software reloads the counter while the counter is greater than the value stored in WIN[11:0], then a reset is generated

To prevent a IWDG reset, refresh the watchdog while the counter value is lower than the time-window value WIN[11:0].

life.augmented

This diagram illustrates how the independent watchdog operates.
When the downcounter reaches zero, the watchdog reset is activated.
This happens when the application software did not refresh the window watchdog on time.

If the software refreshes the watchdog while the downcounter is greater than the value stored in the Window register, then a reset is generated as well.
To prevent a watchdog reset, the refresh must occur when the downcounter value is other than zero, and lower than the time-window value.

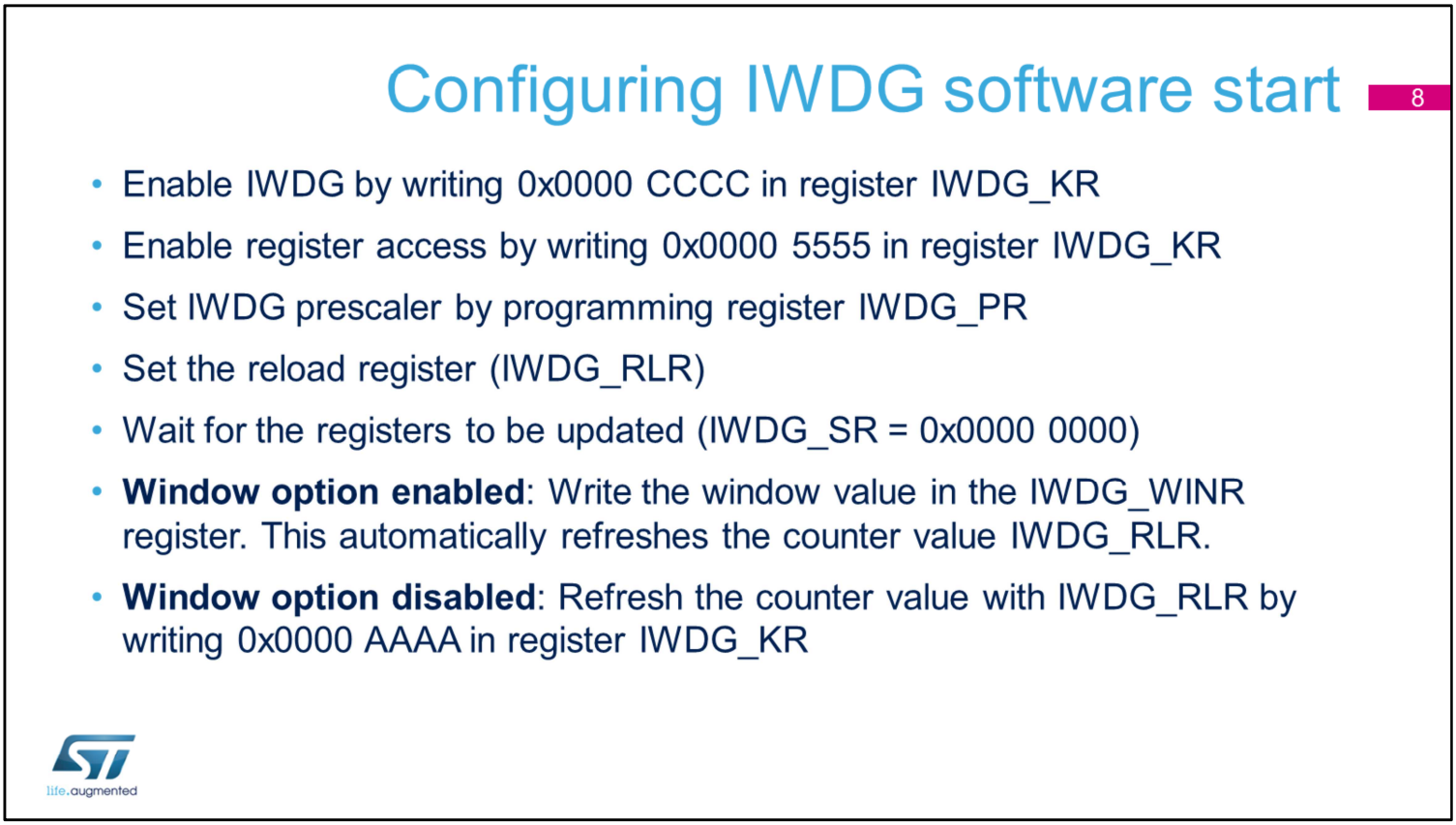
Configuring IWDG hardware start

- In IWDG hardware start, the IWDG is automatically enabled after a system reset.
- Key register (IWDG_KR) must be written by the software with 0x0000 AAAA at regular intervals before the counter reaches 0 and within the window (if the window option is enabled)

life.augmented

The independent watchdog hardware is enabled by the device's option bytes.
If the hardware mode is enabled, after every system reset, the watchdog automatically loads the down-count with 0xFFF, and starts to count down.

To prevent any reset, the Key register must be refreshed at regular intervals before the counter reaches 0 and within the time window, if this option has been selected.



The independent watchdog software start is configured in only a few steps.

- The first step is to write the Key register with value 0x0000 CCCC which starts the watchdog.
- Then remove the independent watchdog register protection by writing 0x0000 5555 to unlock the key.
- Set the independent watchdog prescaler in the IWDG_PR register by selecting the prescaler divider feeding the counter clock.
- Write the reload register (IWDG_RLR) to define the value to be loaded in the watchdog counter.

After accessing the previous registers, it is necessary to wait for the IWDG_SR bits to be reset in order to confirm that the registers have been updated.

- Two options are now available: enable or disable the independent watchdog window option.
    - To enable the window option, write the window value in the IWDG_WINR register.
    - Otherwise, refresh the counter by a writing 0x0000 AAAA in the Key register to disable the window option.

IWDG settings and reset flag

- Setting IWDG time base:
  - IWDG time base prescaled from LSI clock (32 kHz)
    - 7 pre-dividers: 4 to 256 selectable by IWDG_PR register (and 12-bit watchdog counter reload value, RLR[11:0])
  - Setting the IWDG timeout by using the following formula:

    $t_{IWDG} = t_{LSI} \times 4 \times 2^{PR} \times (RL + 1)$

    where t_{LSI} = 1/32000 = 31.25 µs, PR and RL are fields of IWDG registers
- An IWDG reset can be identified via RCC registers

life.augmented

The IWDG time base is prescaled from the LSI clock at 32 kHz. The IWDG_PR prescaler register can divide the LSI clock frequency by 4 up to 256. The watchdog counter reload value is a 12-bit value written in the IWDG_RLR register.
A formula can be used to determine the independent watchdog timeout. The independent watchdog time is based on the LSI period and its prescaler, as well as the selected watchdog counter reload value.

Note that the Reset and Clock Controller (RCC) of the product provides registers giving the source of the reset. In that way the application can check if a reset is caused by an independent watchdog.

Low-power modes

Mode	Description
Run	Active*.
Sleep	Active*.
Stop	Active*
Standby	Active*

* If IWDG enabled

life.augmented

The IWDG can be active in all modes.