WATCHDOG TIMER STM32

What are watchdog timers?

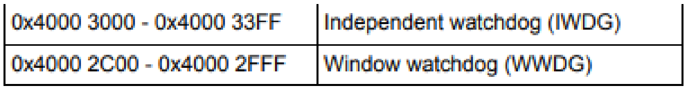
Watchdog timers are used to detect and resolve malfunctions due to software failure, and to trigger system reset or an interrupt when the counter reaches a given timeout value.

IDWG

There are two types of watchdog that we can use in STM32. Independent watchdog and Window Watchdog.This document talks only about the IDWG (Independent watchdog).

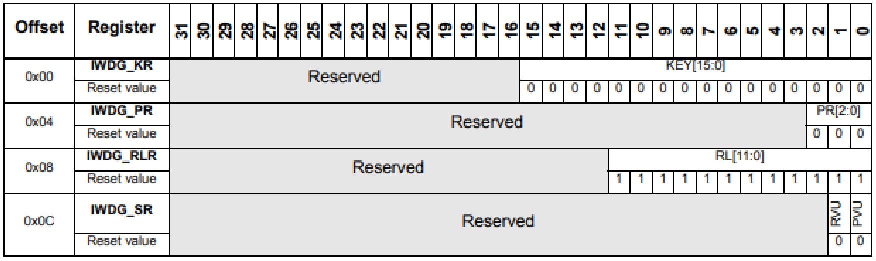
The independent watchdog (IWDG) is clocked by its own dedicated low-speed clock (LSI)
and thus stays active even if the main clock fails.

Registers Address for the watchdog



0x4000 3000 - 0x4000 33FF	Independent watchdog (IWDG)
0x4000 2C00 - 0x4000 2FFF	Window watchdog (WWDG)

Table 97. IWDG register map and reset values



Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	IWDG_KR	Reserved																KEY[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	IWDG_PR	Reserved																													PR[2:0]		
	Reset value																														0	0	0
0x08	IWDG_RLR	Reserved																				RL[11:0]											
	Reset value																					1	1	1	1	1	1	1	1	1	1	1	1
0x0C	IWDG_SR	Reserved																														RVU	PVU
	Reset value																															0	0

Registers for the watchdog

Key Register (IWDG_KR)

The Key Register is the master control for the watchdog.This is a security feature to prevent accidental or buggy software from messing with the watchdog's state.
0xCCCC: Writing this value to the register starts the watchdog timer. Once started, it cannot be stopped.
0xAAAA: Writing this value "feeds" the watchdog. This resets the internal counter and prevents a reset from occurring. This is the value you write in your iwdg_feed() function.
0x5555: Writing this value enables write access to the Prescaler and Reload registers, allowing you to configure the watchdog's timeout period.

Prescaler Register (IWDG_PR)

The Prescaler Register determines the clock speed of the watchdog timer. It divides the independent low-speed internal clock (LSI, typically around 32 kHz) to set the base timing for the watchdog's countdown.
Purpose: The prescaler lets you choose a wide range of timeout values. By dividing the 32 kHz clock, you can extend the timeout from milliseconds to several seconds.
Value: This register holds a value that selects the division factor (e.g., divide by 4, 8, 16, etc.). A larger prescaler value results in a longer timeout.

Reload Register (IWDG_RLR)

The Reload Register holds the timeout value for the watchdog's counter. It's a 12-bit register, so it can hold a value from 0 to 4095.
Purpose: The value in this register is loaded into the counter whenever the watchdog is fed or on a system reset. The counter then begins counting down from this value.
Calculation: The total timeout period is a function of both the prescaler and the reload value. The formula is:
Timeout(seconds)=
(Prescaler Value × (Reload Value+1)) / LSI Clock Speed(Hz)

Status Register (IWDG_SR)

The Status Register provides information about the watchdog's state. It lets you know if the registers are ready for new values or if a new reload value has been applied.
Purpose: This register is primarily used for monitoring. You can check a bit in this register to confirm that a previous write operation to the Prescaler or Reload register has been completed and that the new value is in effect. This is important to ensure proper synchronization between your code and the watchdog hardware.