2025

WATCHDOG TIMER

MICROARCHITECTURE SPECIFICATION

RAZI MUDASSIR 241213

Contents

1. Features

Watchdog timer is used to regain control when the system has failed due to software error or to the failure of an external device to respond in the expected way. The WDT has following features:

- 32-bit down counter with programmable load register.
- Programmable interrupt generation logic with interrupt masking and optional NMI function.
- Lock register protection from runaway software.
- Reset generation logic with enable.
- User-enabled stalling when the MCU asserts the CPU halt flag during debug.

2. Block diagrams

2.1 Level 0 diagram:

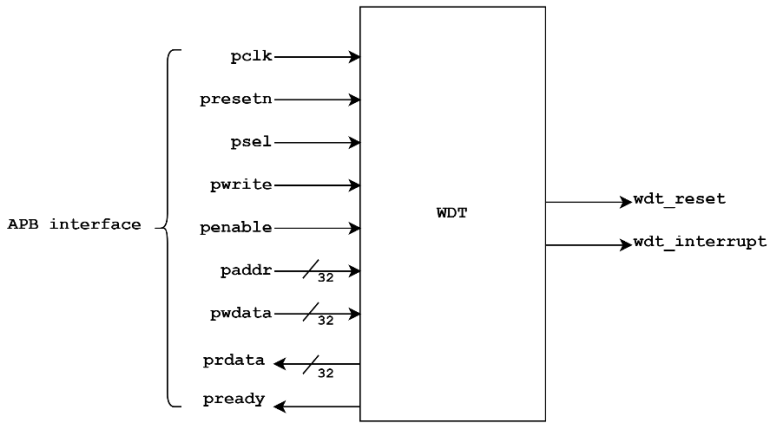

Figure 1: Top level diagram of WDT

2.2 Level 1 diagram:

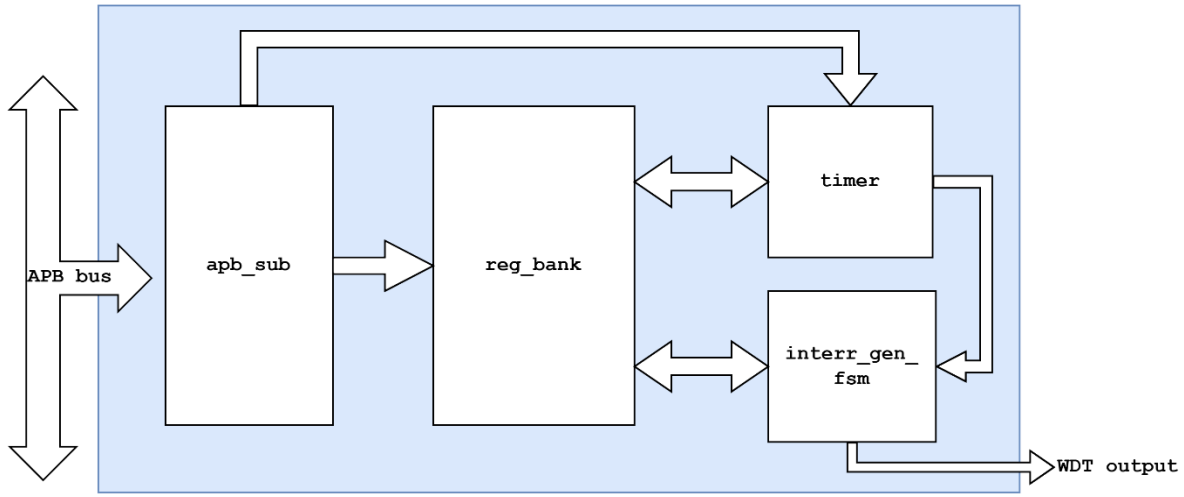

Figure 2: Block diagram for WD

3. Signal description:

The top-level signal description is given as follows:

Type	Signal	Width	Direction	Description
APB interface	pclk	1	input	System clock signal with a frequency of 100MHz.
	presetn	1	input	System reset signal (Active LOW).
	psel	1	input	APB selection signal, high means the completer is required to do a transfer.
	pwrite	1	input	APB write signal, used to select read or write operation. Low indicates a read operation and high indicates write.
	penable	1	input	Enable signal used to initiate access state, data is read or written when it's high.
	paddr	32	input	Address bus for the register.
	pwdata	32	Input	Data bus driven by the requester to write in the timer registers.
	pready	1	Output	Indicates if the device is ready to make a transfer. HIGH value indicates the completer is ready to make a transfer.
	prdata	32	Output	Read data bus, driven by the completer when there is a read request by the requester for a specific value.
wdt	Wdt_interrupt	1	Output	Watchdog timer interrupt signal. HIGH when an interrupt is triggered.
	Wdt_reset	1	Output	External reset. When the WDT is timed out for the second time with no interrupt clear, external interrupt is asserted HIGH for the system to trigger a reset. After asserting this signal, WDT expects to be reset.

4. Functional description:

WDT (Watchdog timer) is a peripheral device used to regain control when the system has failed due to a software error or to the failure of an external device to respond in the expected way. The timer works by loading a value from a programmable register into the down counter. The counter counts down and when it reaches 0, it generates an interrupt. If the interrupt is no cleared by the CPU, the WDT triggers a reset request to the CPU when it reaches a second timeout. The configuration and other registers in the WDT are protected by lock mechanism to prevent any unintentional alteration by software.

4.1 Timer operation:

The WDT is a 32-bit binary down counter. When it's enabled, the counter loads the initial value from the LOAD register. Once it reaches zero for the first time, it generates an interrupt. It then loads the value again from the register and when it reaches 0 for the second time and reset it enabled, external reset (`wdt_reset`) signal is asserted. The system is expected to reset when this reset signal is asserted. When the system resets, the WDT also resets as well.

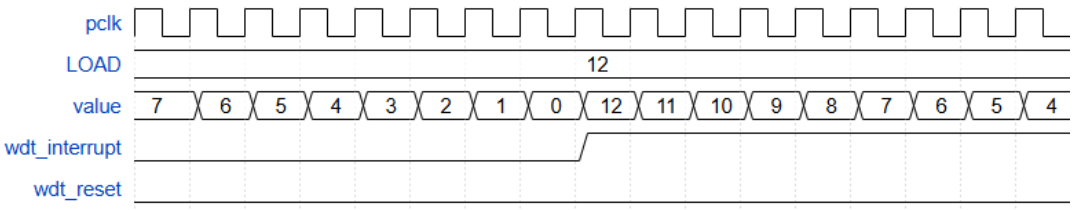

Figure 3: Interrupt trigger for the first timeout

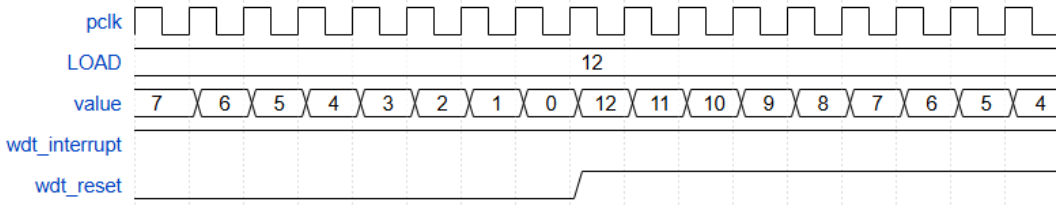

Figure 4: External reset generation after second timeout

If the interrupt is cleared before the second timeout, the external reset is not triggered. Instead, it just generates an interrupt again.

4.2 Disabling interrupt generation:

The enable bit once asserted, can't be de-asserted without a hardware reset. To stop the timer from generating interrupt, TEST.STALL bit is set. This pauses the counter and no interrupt is generated. Once the bit is reset, the counter resumes counting. Stalling the WDT doesn't clear any existing interrupt.

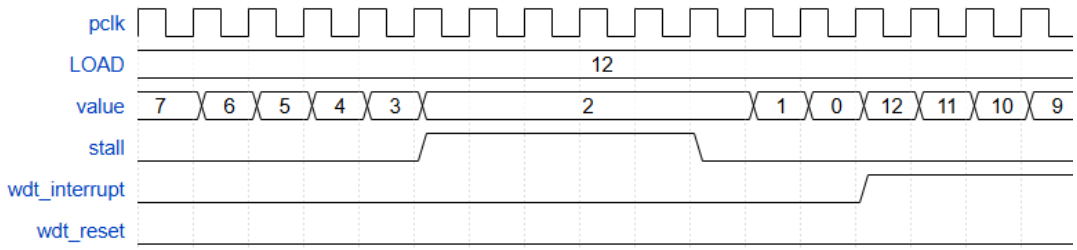

Figure 5: Pausing and resuming the WDT

4.3 Disabling external reset:

External reset generation can be disabled using CTL.RESEN. Setting this bit enables reset generation and disables otherwise. Another way to disable reset generation is that the TEST.EN bit is asserted, the second timeout doesn't trigger a reset, instead, it sets the INT_CAUS.CAUSE_RESET bit high which can be read through the APB interface to check if the external reset has been triggered. However, CTL.RESEN can be locked away by LOCK register but TEST.EN is not lockable, so It can be used to disable interrupt even when the WDT locked.

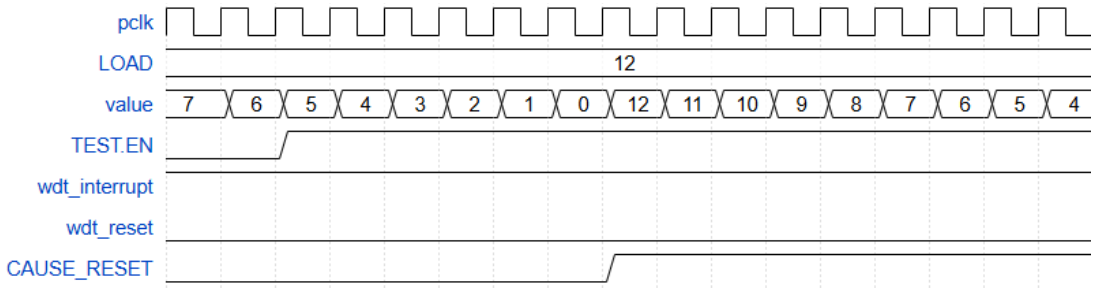

Figure 6: External reset disabling with TEST.EN

4.4 Locking registers:

To prevent the register values to altered unintentionally by software, a lock mechanism is used. By writing ***0x1ACC_E551*** to the LOCK register, the other register can be unlocked for write access. Writing any other value other than this result in locked state with no write access to the registers. TEST.EN bit is not lockable.

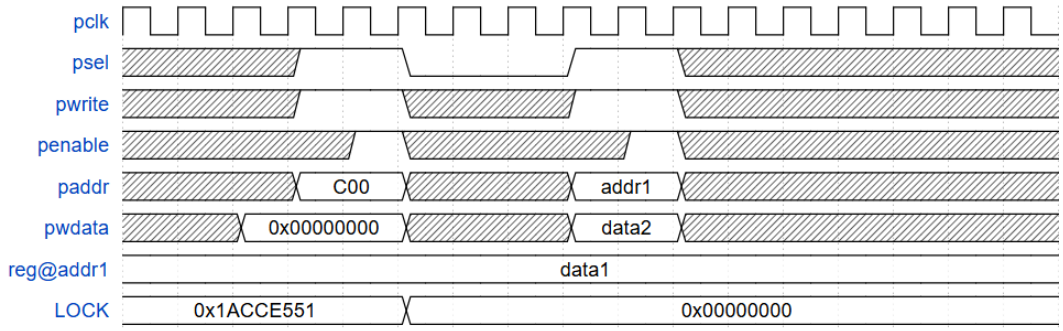

Figure 7: Locking write access with LOCK register

Writing the ***0x1ACC_E551*** value to the LOCK register unlocks write access and the register can be configured using APB interface.

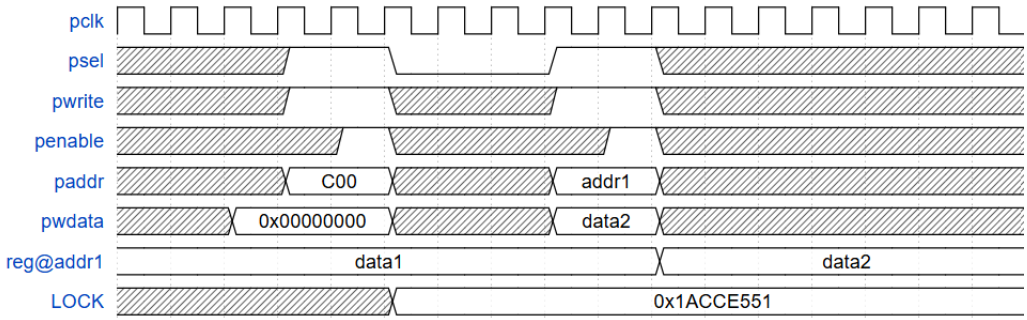

Figure 8: Unlocked write access with LOCK register

5. Block description:

5.1 APB_SUB:

The APB subordinate block is used to maintain the APB protocol to read and write data to timer registers. The block ensures the maintenance of the protocol. `wr_en` and `rd_en` signals are input to the register bank to access read and write registers. `pready` signal is set to a constant HIGH.

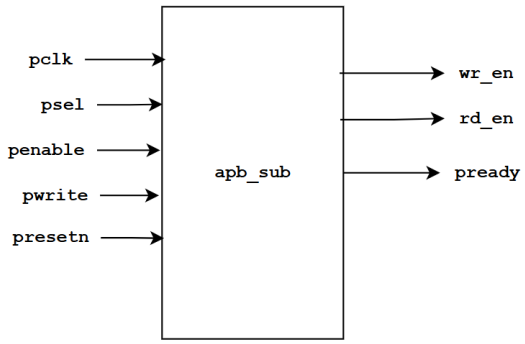

Figure 9: APB-sub block diagram

The WDT has 9 control and status registers which can be configured using APB interface. An APB write or read operation takes two cycles to complete. An APB write can occur as follows:

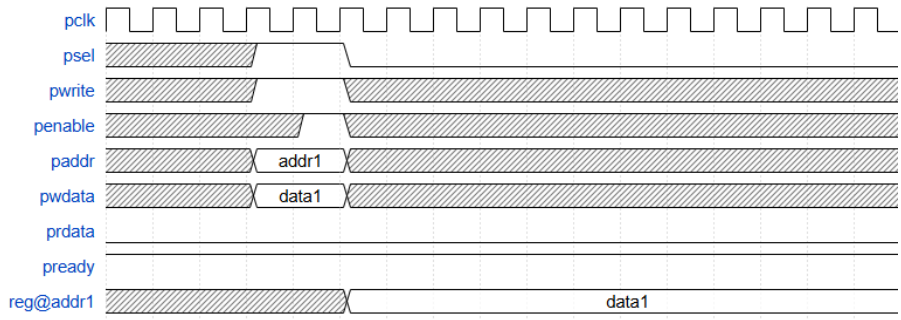

Figure 10: APB write operation

APB read operation occurs in a similar way:

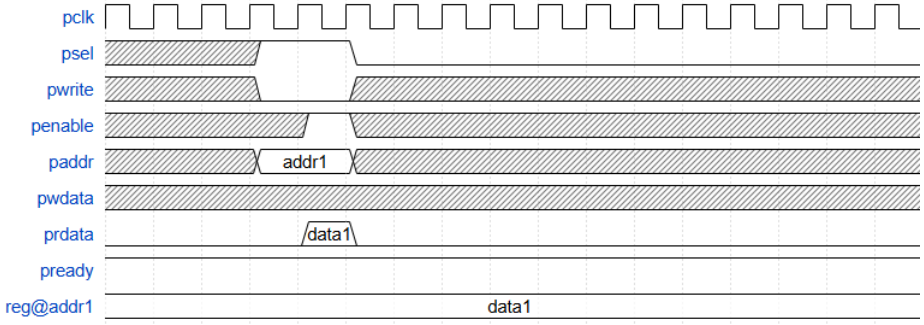

Figure 11: APB read operation

5.2 Reg_bank:

Register bank holds configuration and status register of the WDT. CTL, ICR, RIS, MIS, TEST, INT_CAUS and LOCK register are inside this block. APB interface is used to program these registers. When the register write access is locked using LOCK register, a write from the APB will have no effect on the register except for TEST.EN bit.

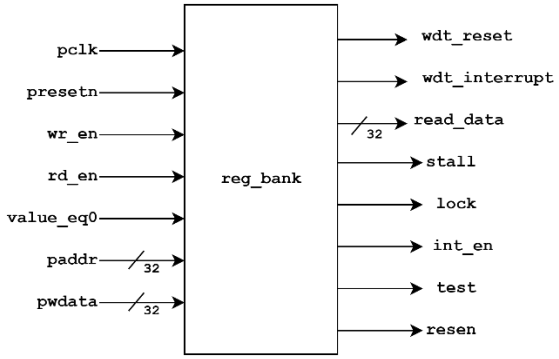

Figure 12: Block diagram of register bank

5.3 Interrupt generation FSM

Interrupt generation FSM block is used to control the generation of interrupt.

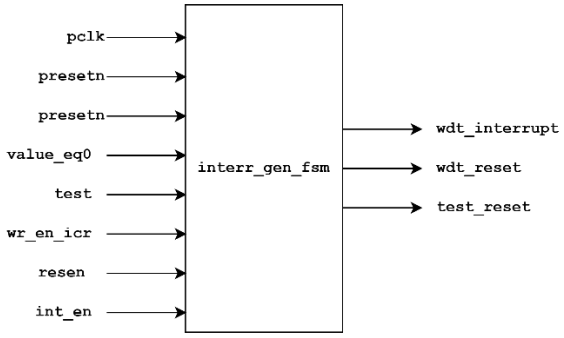

Figure 13: Interrupt generation FSM

The ASM chart for the interrupt generation FSM block is as follows:

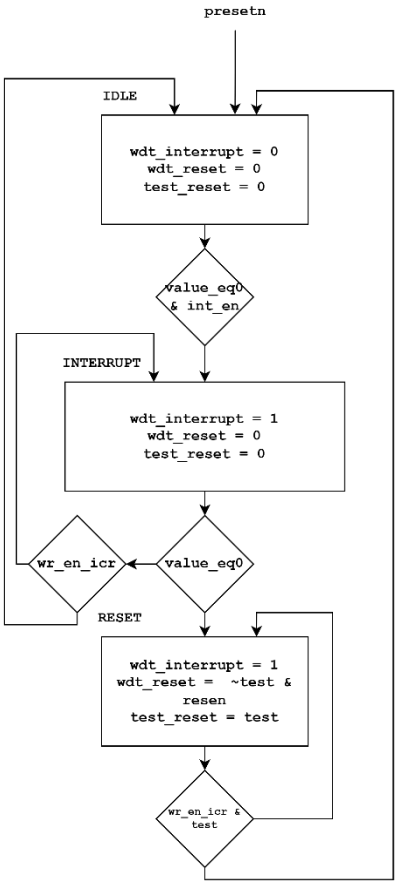

Figure 14: ASM chart for interrupt generation block

5.4 Timer:

Timer block is used to count down and check if the counter value reaches 0. Once it reaches 0, it loads value from the LOAD register and starts counting again. When the LOAD register is receives a new value, it immediately starts counting from that value. Writing to LOAD register doesn't clear any interrupt signal. The counter value can be read from the VALUE register.

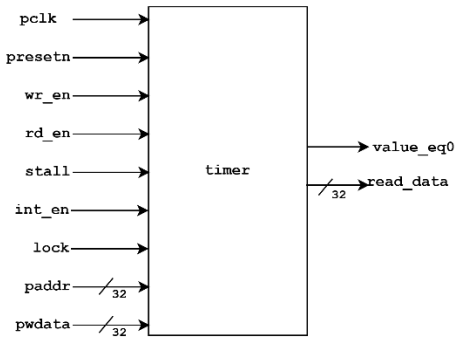

Figure 15: Timer block diagram

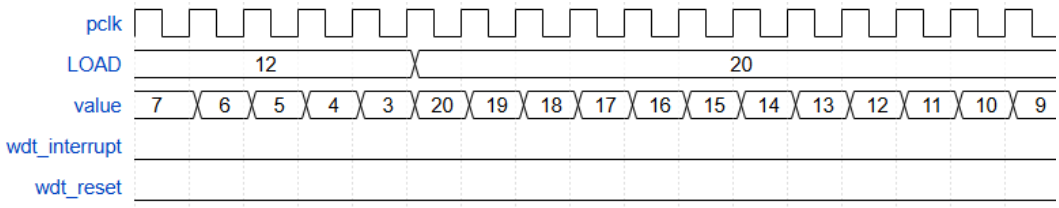

Figure 16: Writing to LOAD register

If the value 0x0000_0000 is written to the LOAD register, it immediately triggers an interrupt.

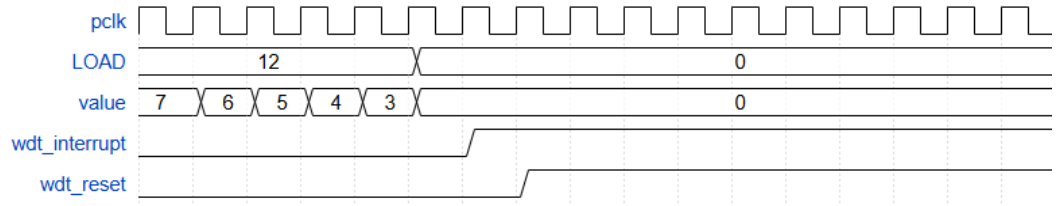

Figure 17: Writing 0x0000_0000 to LOAD register

6. Register description:

6.1 Register map:

The register map for WDT is as follows:

Offset	Acronym	Register name	Section
0h	LOAD	Counter load register	Description
4h	VALUE	Current counter value	Description
8h	CTL	Control register	Description
Ch	ICR	Interrupt clear register	Description
10h	RIS	Raw interrupt status	Description
14h	MIS	Masked interrupt status	Description
418h	TEST	Test mode configuration	Description
41Ch	INT_CAUS	Interrupt cause test mode	Description
C00h	LOCK	Lock register	Description

6.2 LOAD:

[offset 0h] [Summary]

This register is the 32-bit interval value used by the 32-bit counter. When this register is written, the value is immediately loaded and the counter is restarted to count down from the new value. If this register is loaded with 0x0000.0000, an interrupt is immediately generated.

Bit	31:0
Name	WDTLOAD
Access	R/W
Reset value	0xFFFF_FFFF

6.3 VALUE:

[offset 4h] [Summary]

This register contains the current count value of the timer.

Bit	31:0
Name	WDTVALUE
Access	R
Reset value	0xFFFF_FFFF

6.4 CTL:

[offset 8h] [Summary]

This register controls the WDT interrupt enable, interrupt type and reset enable.

Bit	31:3	2	1	0
Name	RESERVED	INTTYPE	RESEN	INTEN
Access	R	R/W	R/W	R/W
Reset value	0x0000_0000	0x0	0x0	0x0

6.5 ICR:

[offset Ch] [Summary]

Interrupt clear register. A write of any value to this register will clear the WDT interrupt and reloads the value from the LOAD register to the 32-bit counter.

Bit	31:0
Name	WDTICR
Access	W
Reset value	0x0000_0000

6.6 RIS:

[offset 10h] [Summary]

Raw interrupt status. This register is the raw interrupt status register. WDT interrupt events can be monitored via this register if the controller interrupt is masked. Value Description
0: The WDT has not timed out
1: A WDT time-out event has occurred

Bit	31:1	0
Name	RESERVED	WDTRIS
Access	R	R
Reset value	0x0000_0000	0x0

6.7 MIS:

[offset 14h] [Summary]

Masked interrupt status. This register is the masked interrupt status register. The value of this register is the logical AND of the raw interrupt bit and the WDT interrupt enable bit CTL.INTEN. Value Description:
0: The WDT has not timed out or is masked.
1: An unmasked WDT time-out event has occurred.

Bit	31:1	0
Name	RESERVED	WDTMIS
Access	R	R
Reset value	0x0000_0000	0x0

6.8 TEST:

[offset 418h] [Summary]

Test mode. This register is used to run WDT in debug mode which prevents it to generate *wdt_reset*. It is also used to stall the timer to pause it, disabling interrupt generation.

Bit	31:9	8	7:1	0
Name	RESERVED	STALL	RESERVED	TEST_EN
Access	R	R/W	R	R/W
Reset value	0x0000_0000	0x0	0x0	0x0

STALL:
WDT Stall Enable
0: The WDT timer continues counting if the CPU is stopped with a debugger.
1: If the CPU is stopped with a debugger, the WDT stops counting. Once the CPU is restarted, the WDT resumes counting.
0h = Disable STALL
1h = Enable STALL

TEST_EN:
The test enable bit
0: Enable external reset
1: Disables the generation of an external reset. Instead bit 1 of the INT_CAUS register is set and an interrupt is generated
0h = Test mode Disabled
1h = Test mode Enabled

6.9 INT_CAUS:

[offset 41Ch] [Summary]

Interrupt cause test mode. This register is used to check the status of WDT interrupt and reset signal when the WDT is in test mode and doesn't generate external interrupt and reset signals.

Bit	31:2	1	0
Name	RESERVED	CAUSE_RESET	CAUSE_INTER
Access	R	R	R
Reset value	0x0000_0000	0x0	0x0

CAUSE_RESET:
Indicates that the cause of an interrupt was a reset generated but blocked due to TEST.TEST_EN (only possible when TEST.TEST_EN is set).

CAUSE_INTR:
Replica of RIS.WDTRIS

6.10 LOCK:

[offset C00h] [Summary]

WDT lock register used to restrict write access to the WDT register from unintentional write from the software. Writing a value of 0x1ACC_E551 unlocks the watchdog resisters for write access. A write of any other value reapplies the lock, preventing any register updates. When read, it returns 0x0000_0000 if unlocked, 0x0000_0001 otherwise.

Bit	31:0
Name	WDTLOCK
Access	W
Reset value	0x0000_0000