AN205319

FR, MB91460, Hardware Watchdog

This application note describes the functionality of the Hardware Watchdog Timer and Watchdog Reset and gives some examples. The Watchdog is an up-counting Timer which has to be cleared in a given time interval. Otherwise a reset is performed. It can be used for restarting an application, if the program gets stuck.

Contents

1 Introduction....1
1.1 Key Features....1
2 The Watchdog Timer....2
2.1 Block Diagram....2
2.2 Registers....3
3 Watchdog Timer Examples....4
3.1 Timer Clear by writing CL bit....4
3.2 Determining Watchdog Reset....4
4 Important Notes....5
5 Additional Information....5
Document History....6

1 Introduction

This application note describes the functionality of the Hardware Watchdog Timer and Watchdog Reset and gives some examples.

The Watchdog is an up-counting Timer which has to be cleared in a given time interval. Otherwise a reset is performed. It can be used for restarting an application, if the program gets stuck.

1.1 Key Features

- RC oscillator as a Clock Sources (100kHz)
- Timer clear by writing 0 to HWWD: CL bit
- Timer stopped in Stop Mode and in Sleep Mode
- Software disabling is not possible
- Timer Interval : ~328ms to ~10486ms

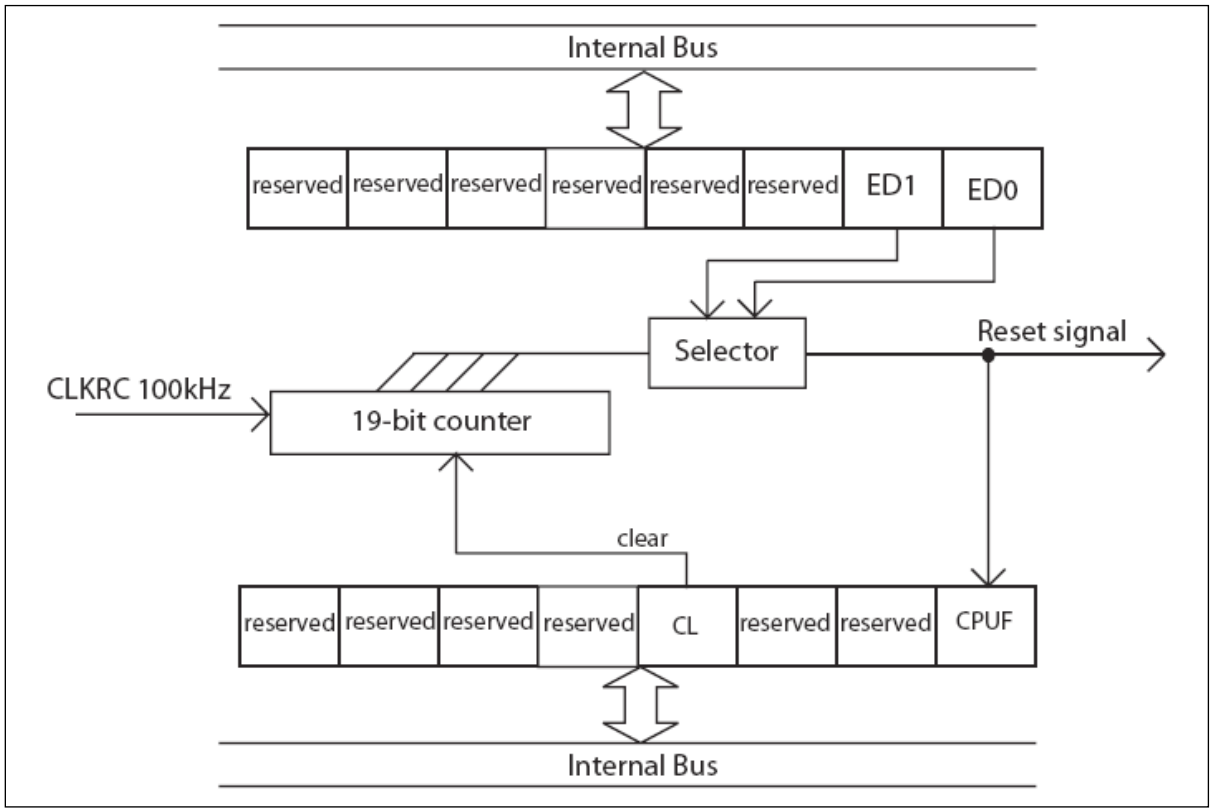
2 The Watchdog Timer

The basic functionality of the watchdog timer

2.1 Block Diagram

Figure 1 shows the internal block diagram of the Watchdog Timer.

Figure 1. Watchdog Timer block diagram



2.2 Registers

2.2.1 Hardware watchdog timer control and status register (HWWD)

Table 1. HWWD

Bit No.	Name	Explanation	Value	Operation
7,6,5	*RESV0*	*Reserved Bit*		Always write “0” to these bits
4	*RESV1*	*Reserved Bit*		Always write “1” to this bit
3	CL	counter clear	0	Watchdog timer is cleared
			1	No effect
2,1	*RESV0*	*Reserved Bit*		Always write “0” to these bits
0	CPUF*	CPU reset Flag	0	Watchdog reset not triggered
			1	Watchdog reset triggered (overflow of watchdog timer occurred)

*:- This bit is initialized by external reset input (INITX) or clock supervisor reset, but not by internal reset. Writing ‘0’ clears this bit, writing ‘1’ has no effect.

2.2.2 Hardware watchdog timer duration register (HWWDE)

Bit No.	Name	Explanation	Value	Operation
7 to 2	-	*Reserved Bit*		Always write “0” to these bits
1,0	ED1,ED0*	Elongate watchdog duration	00	The watchdog period is 2^{16} CLKRC cycles
			01	The watchdog period is 2^{17} CLKRC cycles
			10	The watchdog period is 2^{18} CLKRC cycles
			11	The watchdog period is 2^{19} CLKRC cycles

*:- The timer width is 19-bit. Since the RC oscillator is used as clock source of the hardware watchdog timer, the duration of the timer deviates with the RC oscillator accuracy

	ED1-0	Min	Typ	Max
RC Oscillator cycle (µS)		**5**	**10**	**20**
Watchdog Term (ms)	00	327.68	655.36	1310.72
	01	655.36	1310.72	2621.44
	10	1310.72	2621.44	5242.88
	11	2621.44	5242.88	10485.76

3 Watchdog Timer Examples

Examples for the watchdog

3.1 Timer Clear by writing CL bit

```
 /*                                  SAMPLE CODE                                  */
 /*-------------------------------------------------------------------------------*/

void main(void)
{
    . . .

    . . .

    while (1)
    {
       HWWD_CL = 0;  // Clear Watchdog Timer
       . . .
    }

    . . .
       }
```

3.2 Determining Watchdog Reset

```
  /*                               SAMPLE CODE                                   */
  /*-------------------------------------------------------------------------------*/
  void DetermineWtReset(void)
  {
  unsigned char reset_cause;

  reset_cause = HWWD;       // Read reset cause, also clear it

    if (reset_cause & 0x1)  // Is reset caused by watchdog overflow?
    {
           // Reset caused by watchdog overflow,
       // Take appropriate action
  }
  }
```

4 Important Notes

- Software disabling is not possible
- Hardware disabling is only possible on the evaluation device MB91V460
- In modes where the CPU does not work (SLEEP state, STOP state or STOP with RTC active state), the timer is cleared first then the counting is stopped.
- During DMA transfer between D-bus modules, the writing ‘0’ to CL bit is not possible. Thus, if the transfer time is more than 328ms (calculated from the fastest frequency of the RC oscillator as minimum period), a reset occurs.
- Unlike on MB91V460 Rev.A it is possible on flash devices to elongate the duration of the watchdog reset.
- Unlike on MB91V460 Rev.A it is possible on flash devices to change the CLKRC frequency to 2MHz. Even though the watchdog timer is always operated with a frequency of 100 kHz (10us) typical.
- After releasing INITX the hardware watchdog timer starts immediately without stabilization time. If the timer is not cleared periodically, setting initialization (INIT) reset occurs.
- If one of the below condition occurs, the watchdog counter is cleared
  - Writing “0” to CL bit in the HWWD register
  - Initialization Reset (INIT)
  - Operational Reset (RST)
  - Oscillation stops
  - Transition to the SLEEP state, STOP state or “STOP with RTC running” state

5 Additional Information

Information about Cypress Microcontrollers can be found on the following Internet page:

http://www.cypress.com/cypress-microcontrollers

Document History

Document Title: AN205319 - FR, MB91460, Hardware Watchdog

Document Number:002-05319

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	-	NOFL	05/06/2008	V1.0, First draft, HPi
*A	5090817	NOFL	04/12/2016	Converted Spansion Application Note “MCU-AN-300068-E-V10” to Cypress format
*B	5862837	AESATP12	08/24/2017	Updated logo and copyright.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2008-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.