CoreWatchdog

DirectCore

Product Summary

Intended Use

- Intended for Use with Processor-Based Systems to Protect and Recover from Software Errors

Key Features

- Optimized for Use with CoreMP7 and Cortex™-M1
- Configurable 16-Bit or 32-Bit Watchdog Timer
- Runs from the Advanced Peripheral Bus (APB) Clock (PCLK) – No Additional Clock Required
- Pre-scale Provides Clock Division by up to 1,024
- Supplied in SysBASIC Core Bundle

Benefits

- Prevent an Embedded System from Hanging from Software Errors
- Automatically Connect Clocks and Resets in the System in CoreConsole
- Compatible with Advanced Microcontroller Bus Architecture (AMBA), Cortex-M1, and CoreMP7

ARM Supported Families

- ProASIC®3 (M7A3P, M1A3P)
- ProASIC3E (M7A3PE, M1A3PE)
- Fusion (M7AFS, M1AFS)
- IGLOO™ (M1AGL)
- IGLOOe (M1AGLE)

Synthesis and Simulation Support

- Supported in the Actel Libero® Integrated Design Environment (IDE)

Verification and Compliance

- Compliant with AMBA

Contents

General Description

CoreWatchdog is an APB slave that provides a means of recovering from software crashes. When enabled, CoreWatchdog will generate a soft reset for the system if the microprocessor fails to refresh it on a regular basis. Figure 1 shows a top-level block diagram of CoreWatchdog.
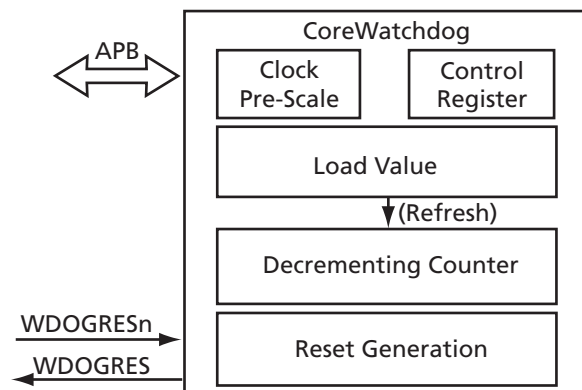

Figure 1 • **CoreWatchdog Block Diagram**

Functional Description

CoreWatchdog is similar to a timer module in that it is based on a decrementing counter which can be used to assert a reset signal if it is allowed to time out.

The width of the decrementing counter in the CoreWatchdog module can be statically configured as either 16 or 32 bits. Processor-accessible registers provide a means to control and monitor the operation of the watchdog.

Operation

The operation of CoreWatchdog is based around a counter. The counter is loaded with a value stored in a load register, and if enabled, it will count down towards zero. If the count reaches zero, the watchdog reset (WDOGRES) is asserted and should be used to reset the system. Under normal operation, the count is prevented from reaching zero by the processor refreshing it at regular intervals.

Counter refresh is effected by writing to the address of the refresh register; any data value may be written. A write to the refresh register address results in the counter being reloaded with the value stored in the load register.

When the counter is already running, writing to the load register will cause the counter to immediately restart at the new value.

The WDOGRES reset signal from CoreWatchdog is asserted for four cycles of PCLK.

A pre-scale unit is used to provide a clock enable pulse for the decrementing counter. The pre-scaler is driven by the APB clock (PCLK) and can be programmed via the watchdog control register to provide an enable pulse every 2, 4, 8, 16, 32, 64, 128, 256, 512, or 1,024 periods of PCLK. This effectively allows the counter to operate from a lower frequency than that at which PCLK is running.

Connecting CoreWatchdog in CoreConsole

Table 1 lists the ports present on the CoreWatchdog module and describes how to connect these in CoreConsole.

Table 1 • **CoreWatchdog Connections**

Connection	CoreConsole Label	Description
Required Connections		
APB Slave Interface	APBslave	Connect this interface to any available slave slot on the APB Bus.
PCLK	PCLK	APB clock Signal.
		Normally connected to the HCLK output of the MP7Bridge or Cortex-M1.
PRESETn	PRESETn	Active low APB reset input.
		Normally connected to the HRESETn output of the MP7Bridge or Cortex-M1.
Watchdog reset input	WDOGRESn	Active low reset input. Asserted when hardware reset is asserted. Resets the circuitry that generates the WDOGRES output.
Watchdog reset output	WDOGRES	Active high reset output, asserted for four cycles of PCLK.
		This signal indicates that the watchdog counter has timed out, and should be used to reset the system.

CoreWatchdog Configurable Options

The width of the counter in CoreWatchdog can be configured as either 16 bits or 32 bits (Table 2)

Table 2 • **CoreWatchdog Configurable Options**

Configurable Option	Default Setting	Description
Counter width	32 Bits	Sets the width of the counter in CoreWatchdog.
		Possible settings are "32 bit" and "16 bit"

Programmer's Model

Table 3 gives the CoreWatchdog registers.

Table 3 • **FROM Data**

Offset	Type	Width	Reset Value	Name	Description
0x00	Read/Write	16 or 32	0x0000 or 0x00000000	WdogLoad	Load value for counter
0x04	Read	16 or 32	0xFFF or 0xFFFFFFFF	WdogValue	Current counter value
0x08	Read/Write	5	0x00	WdogControl	Control register
0x0C	Write	—	—	WdogRefresh	Refresh register

Load Register – WdogLoad

This register contains the value from which the watchdog counter is to decrement. When this register is written to, the counter is loaded with the value written and begins to decrement if the watchdog is enabled.

The load register is either 16 or 32 bits wide, depending on how CoreWatchdog is configured.

Current Value Register – WdogValue

This register gives the current value of the decrementing counter. The reset output signal is asserted if the counter reaches zero.

The current value register is either 16 or 32 bits wide, depending on how CoreWatchdog is configured.

Watchdog Control Register – WdogControl

The bit assignments for the WdogControl register are given in Table 4.

Table 4 • **Bit Assignments for the WdogControl Register**

Bit(s)	Name	Type	Function
31:5	–	–	Enable bit for watchdog:
4	Watchdog Enable	Read/Write	Enable bit for watchdog: 0 = Watchdog disabled (default) 1 = Watchdog enabled
3:0	Pre-scale	Read/Write	Pre-scale field. Determines effective clock rate for counter based on PCLK: 0000 = divide by 2 (default) 0001 = divide by 4 0010 = divide by 8 0011 = divide by 16 0100 = divide by 32 0101 = divide by 64 0110 = divide by 128 0111 = divide by 256 1000 = divide by 512 1001 = divide by 1,024 Others = divide by 1,024

Watchdog Refresh Register – WdogRefresh

A write to this register will cause the counter to be reloaded with the value in the load register. Any data may be written. This register reads zero.

Resource Usage

In a ProASIC3E device, the utilization is as follows:

Configured with 16-bit counter: 280 tiles

Configured with 32-bit counter: 490 tiles

Ordering Information

CoreFROM is included in the SysBASIC core bundle that is supplied with the Actel CoreConsole IP Deployment Platform tool. The obfuscated RTL version of SysBASIC (SysBASIC-OC) is available for free with CoreConsole. The source RTL version of SysBASIC (SysBASIC-RM) can be ordered through your local Actel sales representative. CoreFROM cannot be ordered separately from the SysBASIC core bundle.

List of Changes

The following table lists critical changes that were made in the current version of the document.

Previous Version	Changes in Current Version (v2.0)	Page
Advanced v0.1	The "Product Summary" section was updated to include Cortex-M1 and IGLOO/e information.	1

Datasheet Categories

In order to provide the latest information to designers, some datasheets are published before data has been fully characterized. Datasheets are designated as "Product Brief," "Advanced," "Production," and "Datasheet Supplement." The definitions of these categories are as follows:

Product Brief

The product brief is a summarized version of a datasheet (advanced or production) containing general product information. This brief gives an overview of specific device and family information.

Advanced

This datasheet version contains initial estimated information based on simulation, other products, devices, or speed grades. This information can be used as estimates, but not for production.

Unmarked (production)

This datasheet version contains information that is considered to be final.

Actel and the Actel logo are registered trademarks of Actel Corporation.
All other trademarks are the property of their owners.

www.actel.com

51700068-1/9.07