eSi-Reset

1 Contents

2 Overview

The eSi-Reset core supports the following features:

- Configurable reset delay.
- Configurable number of reset outputs.
- Reset cause register.
- Programmable exception table address register for controlling CPU boot address.
- AMBA 3 APB slave interface.

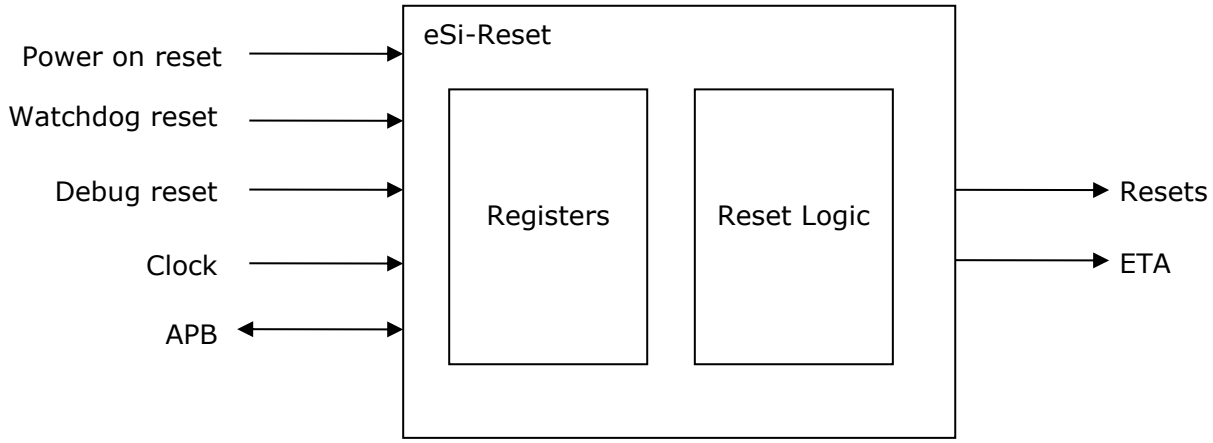

Figure 1: eSi-Reset

3 Hardware Interface

Module Name	cpu_apb_reset
HDL	Verilog
Technology	Generic
Source Files	cpu_apb_reset.v

Port	Type	Description
resets	Integer	Specifies the number of reset outputs
reset_delay	Integer	Number of cycles to wait before releasing reset outputs
reset_state	Vector	Specifies the state of the reset outputs after the delay
reset_delay_counter_width	Integer	Specifies the width of the delay counter

Table 1: Parameters

Port	Direction	Width	Description
power_on_reset_n	Input	1	Power on reset, active-low
watchdog_reset_n	Input	1	Watchdog reset, active-low
debug_reset_cpu	Input	1	Debug reset CPU request, active high
pclk	Input	1	APB clock
presetn	Input	1	APB reset, active-low
paddr	Input	8	APB address
psel	Input	1	APB slave select
penable	Input	1	APB enable
pwrite	Input	1	APB write
eta_reset_0	Input	8/24	Exception table reset address 0
eta_reset_1	Input	8/24	Exception table reset address 1
eta_reset_select	Input	1	Select between eta_reset_0 and eta_reset_1
pclk_cactive	Output	1	Clock active. When deasserted, pclk can be gated
pwdata	Input	16/32	APB write data
pready	Output	1	APB ready
prdata	Output	16/32	APB read data
reset_n	Output	resets	
eta	Output	8/24	Exception table address

Table 2: I/O Ports

For complete details of the APB signals, please refer to the AMBA 3 APB Protocol v1.0 Specification available at:

http://www.arm.com/products/system-ip/amba/amba-open-specifications.php

4 Software Interface

4.1 Register Map

Register	Address offset	Access	Description
reset_n	0x00	R/W	Reset register
cause	0x04	R	Reset cause register
eta	0x08	R/W	Exception table address
auto_deassert	0x0c	R/W	Auto-deassert register

Table 3: Register Map

4.1.1 Reset Register

The reset register allows software control of the reset_n outputs. Each bit in the register corresponds to a bit in the reset_n output signal. The resets are active low, so writing a 0 to a bit in this register asserts the reset, and writing a 1 deasserts the reset output. After any input reset (power on, watchdog or debug), the reset register is first reset to 0, then after a delay of reset_delay clock cycles, set to the value specified by reset_state parameter. A write to the reset register, sets the reset cause register to 3, indicating a software reset.

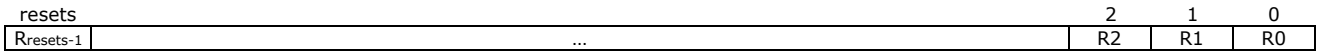

Figure 2: Format of the reset_n register

Register	Values	Description
Rn	0 – Reset asserted 1 – Reset deasserted	Reset state for reset_n[n] output

Table 4: Fields of the reset_n register

4.1.2 Reset Cause

The reset cause allows software to determine the cause of the last reset.

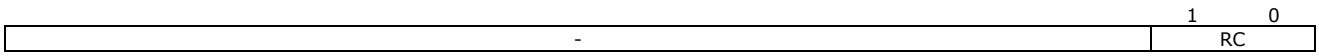

Figure 3: Format of the reset_cause register

Register	Values	Description
RC	b00 – Power on b01 – Watchdog b10 – Debug b11 – Software	Cause of the last reset

Table 5: Fields of the reset_cause register

4.1.3 Exception Table Address

The exception table address register can be used to provide a software programmable reset value for an eSi-RISC CPU's exception table address reset input. The eta output of this reset

module should be connected to the `eta_reset` input of the eSi-RISC CPU. The value of this register is reset (on power up, watchdog or debug reset) to the value input on this modules `eta_reset` input.

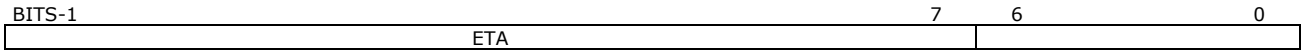

Figure 4: Format of the `eta` register

Register	Values	Description
ETA		Exception table address (8 or 24 MSBs)

Table 6: Fields of the `eta` register

4.1.4 Auto-Deassert Register

The auto deassert register allows the reset module to automatically deassert a reset output after it has been asserted by a write to the reset register. This is needed in the case where a CPU is trying to reset itself, and thus after asserting its own reset, it would not be able to perform a further write to deassert its own reset. The auto deassert register is reset to 0.

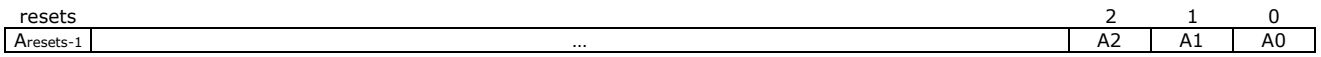

Figure 5: Format of the `auto_deassert` register

Register	Values	Description
An	0 – Manual deassert 1 – Auto deassert	Manual or automatic deassertion of reset

Table 7: Fields of the `auto_deassert` register

5 Revision History

Hardware Revision	Software Release	Description
1	3.3.10	Initial release
2	4.1.14	Add additional `eta_reset` inputs and `eta_reset_select` inputs

Table 8: Revision History

