eSi-CRC

1 Contents

 © 2019 eSi-RISC, All Rights Reserved

2 Overview

The eSi-CRC core can be used to calculate cyclic redundancy check (CRC) values for blocks of data. It supports the following features:

- Programmable CRC polynomial, supporting: CRC-32, CRC-32C, CRC-16, CRC-16-CCITT and CRC-8 as well as others.
- Integrated DMA – zero CPU overhead.
- AMBA 3 AHB-lite slave interface for control register access.
- AMBA 3 AHB-lite master interface for data transfers.

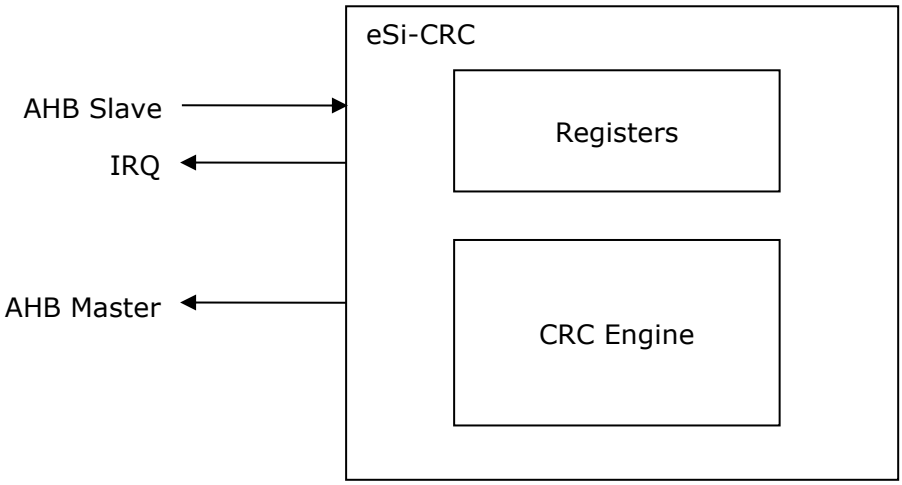

Figure 1: eSi-CRC

 © 2019 eSi-RISC, All Rights Reserved

3 Hardware Interface

Module Name	cpu_ahb_crc
HDL	Verilog
Technology	Generic
Source Files	cpu_ahb_crc.v, cpu_crc.v

Port	Direction	Width	Description
s_hclk	Input	1	Slave interface, AHB clock
s_hresetn	Input	1	Slave interface, AHB reset, active-low
s_haddr	Input	BITS	Slave interface, AHB address
s_hburst	Input	3	Slave interface, AHB burst type
s_hmastlock	Input	1	Slave interface, AHB locked transfer
s_hprot	Input	4	Slave interface, AHB protection
s_hsize	Input	3	Slave interface, AHB size
s_htrans	Input	2	Slave interface, AHB transfer type
s_hwdata	Input	BITS	Slave interface, AHB write data
s_hwrite	Input	1	Slave interface, AHB write
s_hready	Input	1	Slave interface, AHB ready
s_hsel	Input	1	Slave interface, AHB select
s_hready	Output	1	Slave interface, AHB ready
s_hrdata	Output	BITS	Slave interface, AHB read data
s_hresp	Output	1	Slave interface, AHB response
m_hclk	Input	1	Master interface, AHB clock. Must be the same frequency and synchronous to s_hclk
m_hresetn	Input	1	Master interface, AHB reset, active-low
m_hready	Input	1	Master interface, AHB ready
m_hrdata	Input	BITS	Master interface, AHB read data
m_hresp	Input	1	Master interface, AHB response
m_haddr	Output	BITS	Master interface, AHB address
m_hburst	Output	3	Master interface, AHB burst type
m_hmastlock	Output	1	Master interface, AHB locked transfer
m_hprot	Output	4	Master interface, AHB protection
m_hsize	Output	3	Master interface, AHB size
m_htrans	Output	2	Master interface, AHB transfer type
m_hwdata	Output	BITS	Master interface, AHB write data
m_hwrite	Output	1	Master interface, AHB write
interrupt_n	Output	1	Interrupt request, active-low

Table 1: I/O Ports

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

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

 © 2019 eSi-RISC, All Rights Reserved

4 Software Interface

4.1 Register Map

Register	Address offset	Access	Description
address	0x00	R/W	Address register
count	0x04	R/W	Count register
polynomial	0x08	R/W	Polynomial register
crc	0x0c	R/W	CRC register
status	0x10	R/W	Status register
control	0x14	R/W	Control register

Table 2: Register Map

4.1.1 Address Register

The address register contains the base address of the data to calculate the CRC value for.

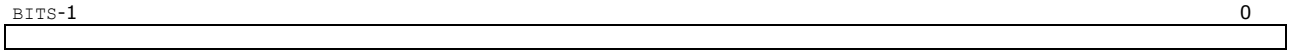

Figure 2: Format of the `address` register

4.1.2 Count Register

The count register contains the number of bytes of data to calculate the CRC value for.

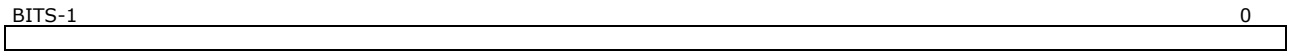

Figure 3: Format of the `count` register

4.1.3 Polynomial Register

The polynomial register contains the CRC polynomial coefficients. The most significant bit of the polynomial is implicit and does not need to be written into this register. For example, CRC-32 has a 33-bit polynomial, but only the lower 32-bits are required.

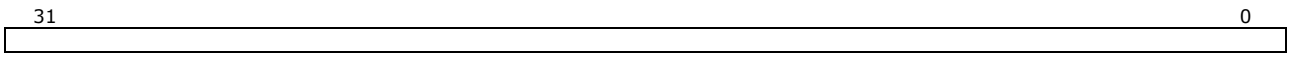

Figure 4: Format of the `polynomial` register

4.1.4 CRC Register

The CRC register contains the computed CRC value. It should be initialised before a CRC operation is started to the initialisation value specified by the corresponding CRC standard (typically all zeros or all ones).

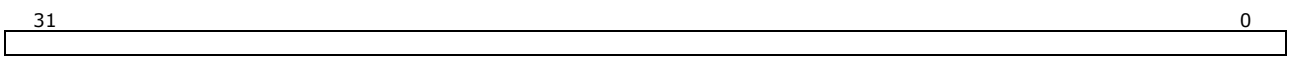

Figure 5: Format of the `crc` register

 © 2019 eSi-RISC, All Rights Reserved

4.1.5 Status Register

The status register contains a selection of flags that indicate the current status of the eSi-CRC module.

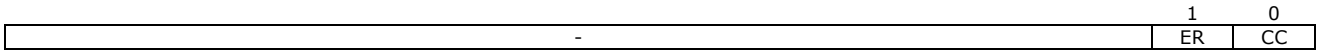

Figure 6: Format of the `status` register

Register	Values	Description
CC	0 - Not complete 1 – Complete	CRC complete
ER	0 – No error 1 – Error	Indicates an error occurred

Table 3: Fields of the `status` register

4.1.6 Control Register

The control register contains a selection of flags that control the operation of the eSi-CRC core.

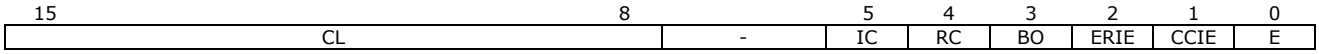

Figure 7: Format of the `control` register

Register	Values	Description
E	0 - Disabled 1 - Enabled	Enables the CRC calculation
CCIE	0 – Disabled 1 – Enabled	CRC complete interrupt enable
ERIE	0 – Disabled 1 – Enabled	Error interrupt enable
BO	0 – LSB first 1 – MSB first	Bit ordering
RC	0 – Do not reflect CRC 1 – Reflect CRC	Reflect CRC (reverse bit order)
IC	0 – Do not invert CRC 1 – Invert CRC	Invert CRC
CL	4 – 32	CRC length

Table 4: Fields of the `control` register

4.2 Usage

The following table details the register settings required to compute standard CRCs.

Standard	Polynomial	Initial CRC	BO	RC	IC	CL
CRC-32	0x04C11DB7	0xFFFFFFFF	0	0	1	32
CRC-32C	0x1EDC6F41	0xFFFFFFFF	0	0	0	32
CRC-16	0x8005	0x0000	0	0	0	16
CRC-16-CCITT	0x1021	0xFFFF	1	1	0	16
CRC-16-CCITT (Kermit)	0x1021	0x0000	0	0	0	16
CRC-8	0x07	0x00	1	1	0	8

 © 2019 eSi-RISC, All Rights Reserved

Table 5: Standard CRC Settings

4.3 Interrupts

The eSi-CRC core supports the following interrupts.

- CRC complete interrupt
- Error interrupt

The CRC complete interrupt will be raised when the CRC value has been calculated and written to memory. The `CC` flag in the `status` register will be set 1 to indicate this. When the `CC` flag in the `status` register is set to 1 and the `CCIE` flag in the `control` register is set to 1, the CRC complete interrupt will be asserted.

The error interrupt will be raised then the `ER` flag in the `status` register is 1 and the `ERIE` flag in the `control` register is set to 1. This indicates an error was detected while reading data from memory.

 © 2019 eSi-RISC, All Rights Reserved

5 Revision History

Hardware Revision	Software Release	Description
1	2.6.4	Initial release

Table 6: Revision History

 © 2019 eSi-RISC, All Rights Reserved