ATA-3 IP-Core

DAT_O	32	Output	Data from the core	ALL
SEL_I	4	Input	Byte select signals	ALL
WE_I	1	Input	Write enable input	ALL
STB_I	1	Input	Strobe signal/Core select input	ALL
CYC_I	1	Input	Valid bus cycle input	ALL
ACK_O	1	Output	Bus cycle acknowledge output	ALLALLBd 36e TD 0

ALLALL
1 ALLA 1 Input

OpenCores

2.2.14 INTA_O
The interrupt request output [INTA_O], when asserted indicates that a connected device needs servicing. The [INTA_O] output is a relay from the [INTRQ] ATA interface signal.

2.2.15 DMA_req
The DMA request output [DMA_req], when asserted indicates that the controller wants to transfer data between the MASTER and the core via DMA. The [DMA_req] signal is asserted when the controller's DMA Transmit Buffer is empty during a DMA write transfer, or when the controller's DMA Receive Buffer Contains data. This signal should be connected to an external DMA engine, like the OpenCores WISHBONE DMA core.

2.2.16 DMA_ack
The DMA acknowledge input [DMA_ack], when asserted indicates the termination of the current DMA cycle.

2.3 ATA signals

Port	Width	Direction	Description	Device
RESETn	1	Output	IDE hardware reset	ALL
DDi	16	Input	Device Data (from ATA devices)	ALL

3

Registers

3.1 Core Registers list

Name	Address	Width	Access	Description	Device

3.4 PIO Compatible Timing Register [PCTR]

Bit #	Access	Description

When the BeLeC bit is cleared (‘0’) for the selected device, the core does not perform endian conversion. It transfers Data1 first and then Data2, without the MSB and LSB swapped.
First DMA cycle: Data1(15:8) => DD(15:8), Data1(7:0) => DD(7:0)
Second DMA cycle: Data2(15:8) => DD(15:8), Data2(7:0) => DD(7:0)

When the host writes to the buffer while it is full (DMATxFull = ‘1’), the core

OpenCores

4

0 8 O p e n C o r e s 0 9 4 0 . 3 2 0 0 9 4 0 . 0 w

R.07

DMA Receive Buffers are not full. The core asserts the WISHBONE DMA_req signal as soon as data is available in the DMA Receive Buffers. When the buffers become full the core negates DMACK-, until the host empties the receive buffers by reading from the DMA Buffer address. If the ATA devices still have the DMARQ line asserted, the core asserts the DMACK- line again and continues the DMA transfer.

4.8 DMA Big Endian versus Little Endian conversion

The core can perform automatic big versus little endian conversion during DMA transfers. This feature can be enabled per device, by setting (‘1’) the appropriate BeLeC bits in the control register. Figure 4.7 shows a 32bit DMA WRITE without endian conversion. Figure 4.8 shows a 32bit DMA WRITE with endian conversion. In both examples the data to be written is 0x12345678.

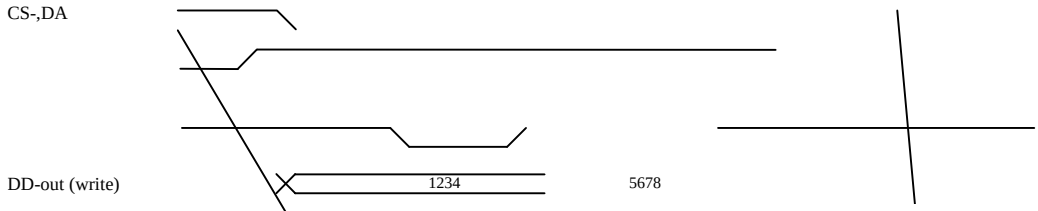

Figure 4.7 32bit DMA write transfer, without endian conversion