CONTROLLER DIRECT MEMORY ACCESS DMA

SPECIFICATION

1 FEATURES

- Proved with 180nm SMIC CMOS
- DMA Controller allows transferring blocks of data without CPU
- Eight DMA channels
- Possibility data transfer such as memory-memory, memory-device and device-memory
- VHDL RTL compatible with Synopsys, Autologic, etc. synthesizing tools
- Option of synthesis to FPGA

2 APPLICATION

- Built-in custom integrated circuits

3 OVERVIEW

Controller Direct Memory Access (DMA) allows the exchange of data between the memory unit and peripherals without CPU. An internal or external memory or peripherals (USB, UART, SPI, NAND and LCD) can be used as a source or receiver of the data. The controller connects to the 32-bit data buses APB and AHB and uses the input signals DREQ_xxx [31: 0] as inquires for exchange of peripherals. DMA Controller has eight independent communication channels. Each can be programmed for communication between a particular source and receiver. All exchanges can be carried out in parallel in time-sharing mode. If there is a simultaneous request to enable multiple channels, the selection of the active channel is carried out by a special arbitrator. After each exchange cycle, the priorities of channels vary.
After reading the data from the source DMA places them in the internal buffer and then performs a data transfer to the receiver.
Channel turn on if the registers are programmed, and the corresponding enable bit in register DMA_EN was set. The processor programs the channel registers for turn on the channel.
The channel is active if is currently performing data transfer from the source to the receiver. The channel can be activated by signal Ready from a source or receiver. Channel is active only in the time interval required for transferring programmed data block. This may be one or four words (half-words, bytes). An interrupt can be generated after activating the channel. A re-activation of the channel is possible only on the request and with permission from the channels arbiter. That eliminates channel in the seizing of the DMA controller.
Channel turns off when the packet of information that programmed in the counter of data exchange has been delivered. The registers of the channels contain the double buffers that allow program the new exchange when the channel has turned on without waiting for the end of the current exchange.
The controller can perform transfers: "Memory-Memory", "Memory-Device", "Device-Memory" and "Device-Device". Transferring data between memories ("memory-memory") don’t require generate an external request to activate the channel. Such a request will be always present if it setup using software. When data transfer is involving a device, you must program the

activation of the channel by a signal Device Ready. If the memory is considered to be "always ready" the peripheral device must independently produce signal Ready.
DMA contains the individual channels registers and the registers of the common control the channels. All registers are connected to the peripheral bus and have the base addresses which defined by data on bus ps_ba[9:0].

4 BLOCK-DIAGRAM

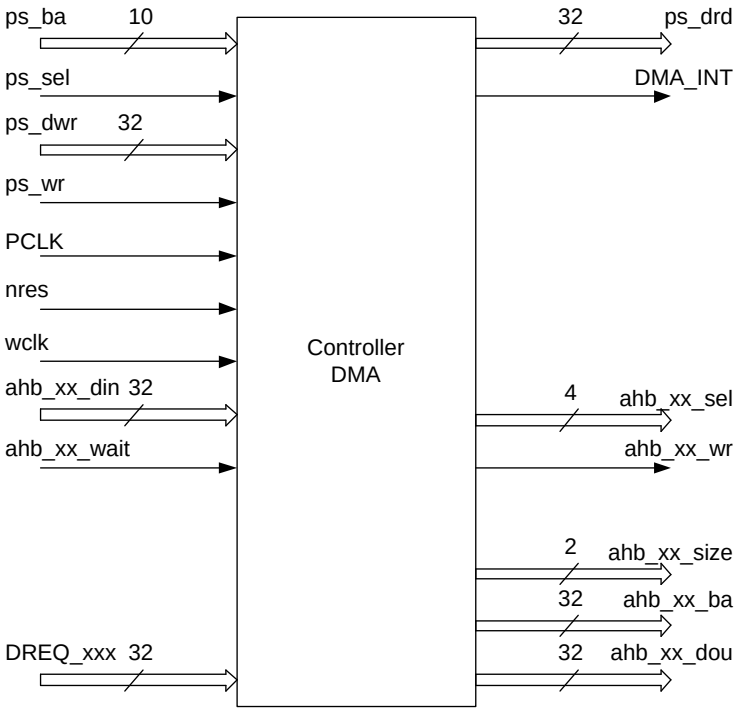

Figure 4.1: DMA controller interface

5 PIN DESCRIPTION

Name	Direction	Description
System interface (bus APB)		
ps_ba[9:0]	I	Registers address
ps_sel	I	Register select
ps_dwr[31:0]	I	Data to write to registers
ps_wr	I	Signal to write data
PCLK	I	APB Clock
nres	I	Reset (low – active)
DMA_INT	O	Interrupt Signal
ps_drd[31:0]	O	Data to read from registers
DREQ_xxx[31:0]	I	External inquiries signals to DMA
System interface (bus AHB)		
wclk	I	AHB Clock
ahb_xx_wait	I	Signal wait
ahb_xx_sel[3:0]	O	Device select
ahb_xx_wr	O	Signal AHB write
ahb_xx_size[1:0]	O	Type of data (byte, half-word, word)
ahb_xx_ba[31:0]	O	Address bus
ahb_xx_din[31:0]	I	Input data bus
ahb_xx_dou[31:0]	O	Output data bus

6 DELIVERABLES

IP contents:

- RTL Code
- Test benches (optional)
- Synthesis scripts
- Documentation