

# DDR2 Block Data Read Write

[Show Contents...](#)[Hide Contents...](#)

## Contents

- [Overview](#)
- [Block Level Diagram](#)
- [Interface Signals](#)
- [Waveform](#)
- [Simulation](#)
  - [Simulation Tests](#)
    - [Test 1: Self test](#)
    - [Test 2: Sending Packets](#)
- [Synthesis](#)
- [Downloading Bitfile](#)
- [Throughput Analysis](#)

## Overview

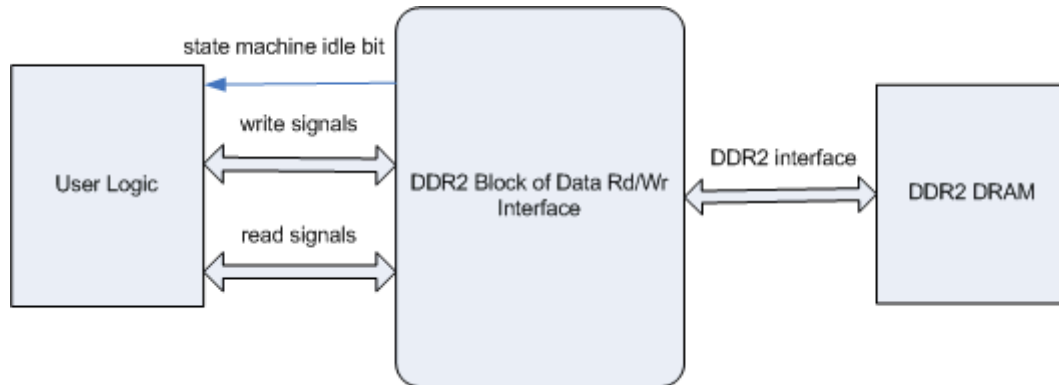
This module provides a NetFPGA developer the block-of-data read/write interface to the 64MB DDR2 DRAM chips on the NetFPGA card. It has the following salient features:

- The block-of-data read/write signals from and to the user logic accessing the DRAM are all synchronous to the user logic clock. The user logic is insulated from the DDR2 clock domain signals.
  - The user logic can request read and/or write of a block of data to the DDR2 DRAM. The read/write requests are arbitrated and granted by a round-robin arbiter. The granted request can then transfer the block of data into/out of the DRAM. No new read/write request will be granted until the outstanding block of data transfer has been completed.
  - DRAM operation (including DRAM refresh) details are invisible to the user logic.
  - It's required that a combined throughput for read and write to DDR2 DRAM be greater than 9.9 Gbps.
  - The data width for DDR2 is 64 bits. The DDR2 clock frequency is 200 MHz. The DDR2 block size is 2048 bytes.
  - By changing Verilog parameter, two configurations of the packet data width for the user logic are provided.
1. The packet data width is 144 bits and therefore the block size for the user logic to read/write DDR2 DRAM is 2034 bytes. This configuration should work with the user logic clock frequency 125 MHz to achieve the required DDR2 DRAM throughput.
  2. The packet data width is 288 bits and therefore the block size for the user logic to read/write DDR2 DRAM is 2016 bytes. This configuration should work with the user

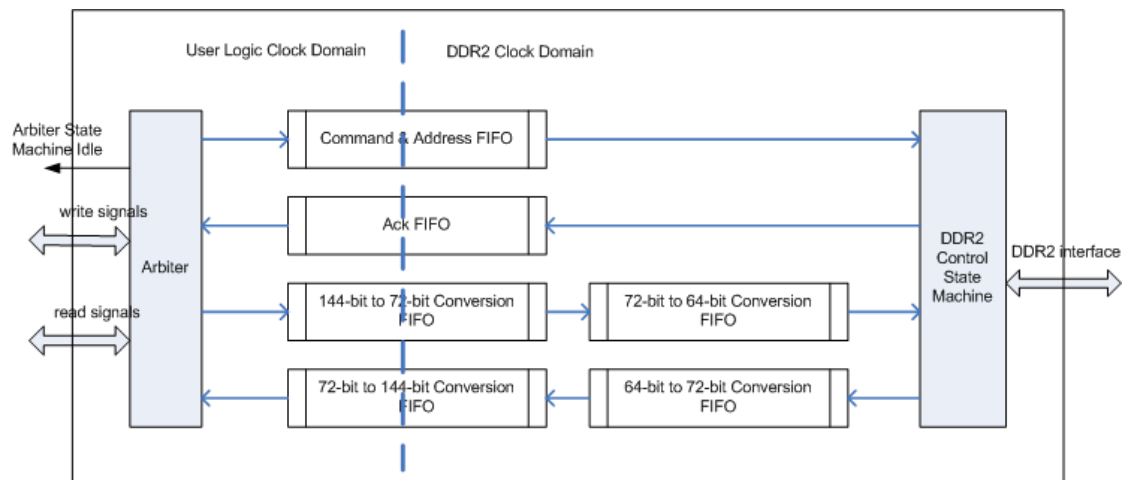
logic clock frequency 62.5 MHz or 125 MHz to achieve the required DDR2 DRAM throughput.

## Block Level Diagram

- The first diagram shows how the DDR2 block of data read/write interface is used by the user logic to access the DDR2 DRAM.



- The second diagram shows the composition of the DDR2 block of data read/write interface module.



The configuration of packet data width 144 bits is shown in the diagram. If the packet data width is 288 bits, "144-bit to 72-bit Conversion FIFO" is replaced by "288-bit to 72-bit Conversion FIFO", and "72-bit to 144-bit Conversion FIFO" is replaced by "72-bit to 288-bit Conversion FIFO".

## Interface Signals

- The user logic interfaces the following write signals with the DDR2 block-of-data read/write module.

Note: parameter PKT\_MEM\_PTR\_WIDTH = 22 (DDR2 data bus width is 8 bytes. Only burst length of 2 is supported. So there are 4M pieces of 8\*2 bytes in the 64MB DRAM), parameter PKT\_DATA\_WIDTH = 144 or 288 (data bus width to the user logic) in the design.

Signal Group	Signal Name	Direction	Bits	Description
Request Negotiation	p_wr_req	from user logic to block-of-data rd/wr module	1	1=request for write transfer (data are from user logic to DRAM), 0=otherwise
Request Negotiation	p_wr_ptr	from user logic to block-of-data rd/wr module	PKT_MEM_PTR_WIDTH	the start address of DRAM for transfer. Each unit is 16-byte piece
Request Negotiation	p_wr_ack	from block-of-data rd/wr module to user logic	1	1=the arbiter acknowledges that the write requester can proceed, 0=otherwise
Data Transfer	p_wr_data_vld	from user logic to block-of-data rd/wr module	1	1=the write data is valid, 0=otherwise
Data Transfer	p_wr_data	from user logic to block-of-data rd/wr module	PKT_DATA_WIDTH	the data transferred from user logic to DRAM
Data Transfer	p_wr_full	from block-of-data rd/wr module to user logic	1	1=notify the user logic to pause transfer the next clock cycle until this signal is deasserted, 0=otherwise
Data Transfer	p_wr_done	from block-of-data rd/wr module to user logic	1	1=this is the last write and no more write will be accepted for this block-of-data, 0=otherwise

- The user logic interfaces the following read signals with the DDR2 block of data read/write module.

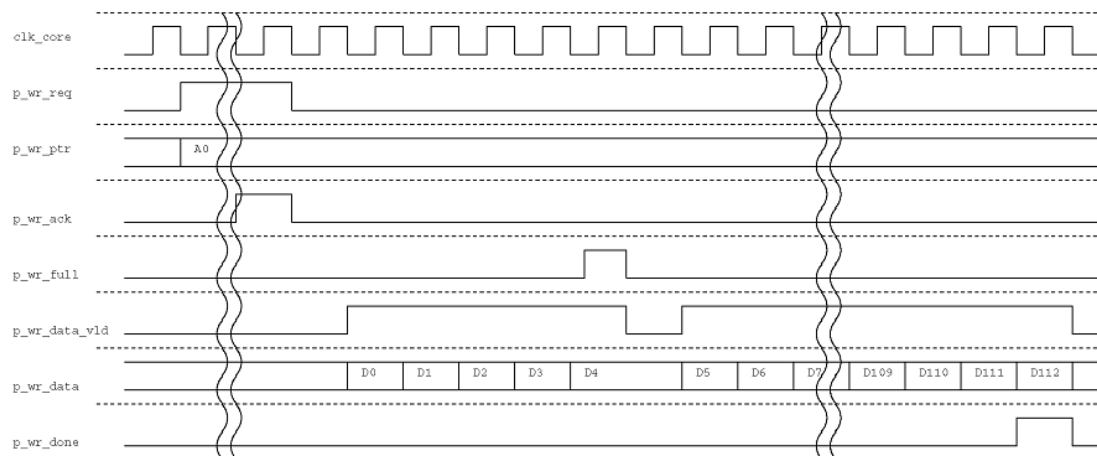
Note: parameter PKT\_MEM\_PTR\_WIDTH = 22 (DDR2 data bus width is 8 bytes. Only burst length of 2 is supported. So there are 4M pieces of 8\*2 bytes in the 64MB DRAM), parameter PKT\_DATA\_WIDTH = 144 or 288 (data bus width to the user logic) in the design.

Signal Group	Signal Name	Direction	Bits	Description
--------------	-------------	-----------	------	-------------

Signal Group	Signal Name	Direction	Bits	Description
Request Negotiation	p_rd_req	from user logic to block-of-data rd/wr module	1	1=request for read transfer (data are from DRAM to user logic), 0=otherwise
Request Negotiation	p_rd_ptr	from user logic to block-of-data rd/wr module	PKT_MEM_PTR_WIDTH	the start address of DRAM for transfer. Each unit is 16-byte piece
Request Negotiation	p_rd_ack	from block-of-data rd/wr module to user logic	1	1=the arbiter acknowledges that the read requester can proceed, 0=otherwise
Data Transfer	p_rd_rdy	from block-of-data rd/wr module to user logic	1	1=block-of-data rd/wr module has data for user logic to read, 0=otherwise
Data Transfer	p_rd_en	from user logic to block-of-data rd/wr module	1	1=user logic reads out one word of data from the block-of-data rd/wr module, 0=otherwise
Data Transfer	p_rd_data	from block-of-data rd/wr module to user logic	PKT_DATA_WIDTH	data transferred from block-of-data rd/wr module to user logic
Data Transfer	p_rd_done	from block-of-data rd/wr module to user logic	1	1=this is the last read data and no more data will be read for this block-of-data, 0=otherwise

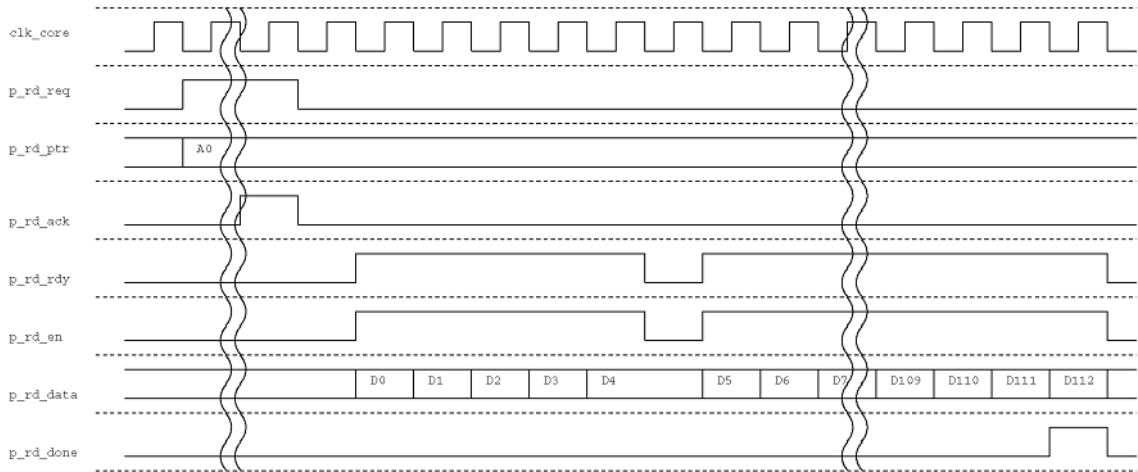
## Waveform

- Block-of-data Write Transaction Timing Diagram



This diagram shows the case of packet data width *144* bits and in total 113 words of 144-bit data are transferred. In case of packet data width *288* bits, in total 56 words of 288-bit data are transferred instead.

- Block-of-data Read Transaction Timing Diagram



This diagram shows the case of packet data width *144* bits and in total 113 words of 144-bit data are transferred. In case of packet data width *288* bits, in total 56 words of 288-bit data are transferred instead.

# Simulation

The simulation tests verify the functionality of the DDR2 Block Data Read/Write module. In order to run the tests, change the working directory to sub-directory: projects/ddr2\_blk\_rdwr/verif/

## Simulation Tests

### Test 1: Self test

Name

test\_dram\_simple

Description

Write all blocks of DRAM and read back the first 50 blocks and verify the counters  
 # Initialize netfpga hardware  
 # Start writing ascending pattern of data to DRAM  
 # Read first 50 blocks of data  
 # Check the read data pattern. \$ Location  
 projects/ddr2\_blk\_rdwr/verif/test\_dram\_simple \$ Command nf21\_run\_test.pl --major  
 dram --minor simple \$ Output

SUCCESS!

## Test 2: Sending Packets

Name

test\_dram\_sendpkt

Description

Send two files - one big and one small - to DRAM and verify the counters

# Initialize netfpga hardware

# Send two files to DRAM (big file followed by small file)

# Check counters to verify number of packets sent. \$ Location  
projects/ddr2\_blk\_rdwr/verif/test\_dram\_sendpkt \$ Command nf21\_run\_test.pl --major  
dram --minor sendpkt \$ Output

FAILURE!

## Synthesis

To run synthesis, mapping, place and route, change the working directory to the sub-directory projects/ddr2\_blk\_rdwr/synth/ and run the following command:

```
make really_clean; make
```

## Downloading Bitfile

You must log on as "root" to load the bit file to NetFPGA card and run the test. Follow the steps below:

1. Change the working directory to the sub-directory projects/ddr2\_blk\_rdwr/synth/ and run command: nf2\_download nf2\_top\_par.bit
2. Change the working directory to the sub-directory projects/ddr2\_blk\_rdwr/sw/ and run the "make" command to build executable program "throughput".
3. Take note for the PKT\_DATA\_WIDTH parameter used in the test circuit. Select an allowed combination of and from the table below:

system clk frequency select	pkt_data_width_bits
0	288
1	144
1	288

## Throughput Analysis

Change the working directory to the sub-directory  
projects/ddr2\_blk\_rdwr/sw/ and run the following command for your  
selected combination:

```
./throughput
```

The program "throughput" prints on the monitor screen:

1. Good: the cumulative number of attempts of writing all 32K blocks (2KB each block), reading and comparing without mismatch.
2. Bad: the cumulative number of attempts of writing all 32K blocks (2KB each block), reading and comparing with at least one mismatch.
3. Iteration: the sum of Good and Bad.
4. The time-average measured throughput for user logic to access DDR2 DRAM from the moment "throughput" was run to the current time. The throughput includes both write and read to DDR2 DRAM.