HUNT ENGINEERING
Chestnut Court, Burton Row,
Brent Knoll, Somerset, TA9 4BP, UK
Tel: (+44) (0)1278 760188,
Fax: (+44) (0)1278 760199,
Email: sales@hunteng.co.uk
www.hunteng.co.uk
www.hunt-dsp.com

The HERON FPGA12 Example3

Rev 1.0 R. Williams 14-02-06

The HERON-FPGA12 is a module that has a Virtex-4 FX Xilinx FPGA and 128Mbytes of DDR SDRAM memory. The Virtex-4 FX FPGA includes a Power PC processor core.

Users can use the HERON-FPGA12 to provide a custom storage capability using the DDR SDRAM to store large amounts of data. The module may also use the FPGA and Power PC processor core to process the data stored in the DDR SDRAM.

With 128Mbytes of DDR memory available the HERON-FPGA12 is ideally suited for creating a large FIFO, with the memory interface of the DDR SDRAM providing the storage for the FIFO.

The DDR memory of the HERON-FPGA12 is used to create a single 128Mbyte FIFO that is interfaced between one HERON input FIFO and one output FIFO.

The DDR memory interface of the HERON-FPGA12 enables high speed data storage, with access speeds of 1.6Gbytes/sec possible per bank between the DDR memory and FPGA. This data rate is demonstrated by the example.

As such Example3 is highly suitable for applications that require buffering between high speed data sources like A/D converters of IO modules, and slower non-real-time host PC interfaces such as PCI.

History

Example revision 1.0	14-02-06	Developed from Example3 for the HERON-FPGA9.

What the Bit-stream Does

Example3 (SDRAM FIFO) for the HERON-FPGA12 is supplied on the HUNT ENGINEERING CD, and web site. The FPGA source code is supplied along with a bit-stream that can be loaded directly onto the HERON-FPGA12.

If you make changes to the project and re-build it you can change the functionality to be whatever you want, but if you use the supplied bit-stream you need to know what it is doing. This document describes that for you.

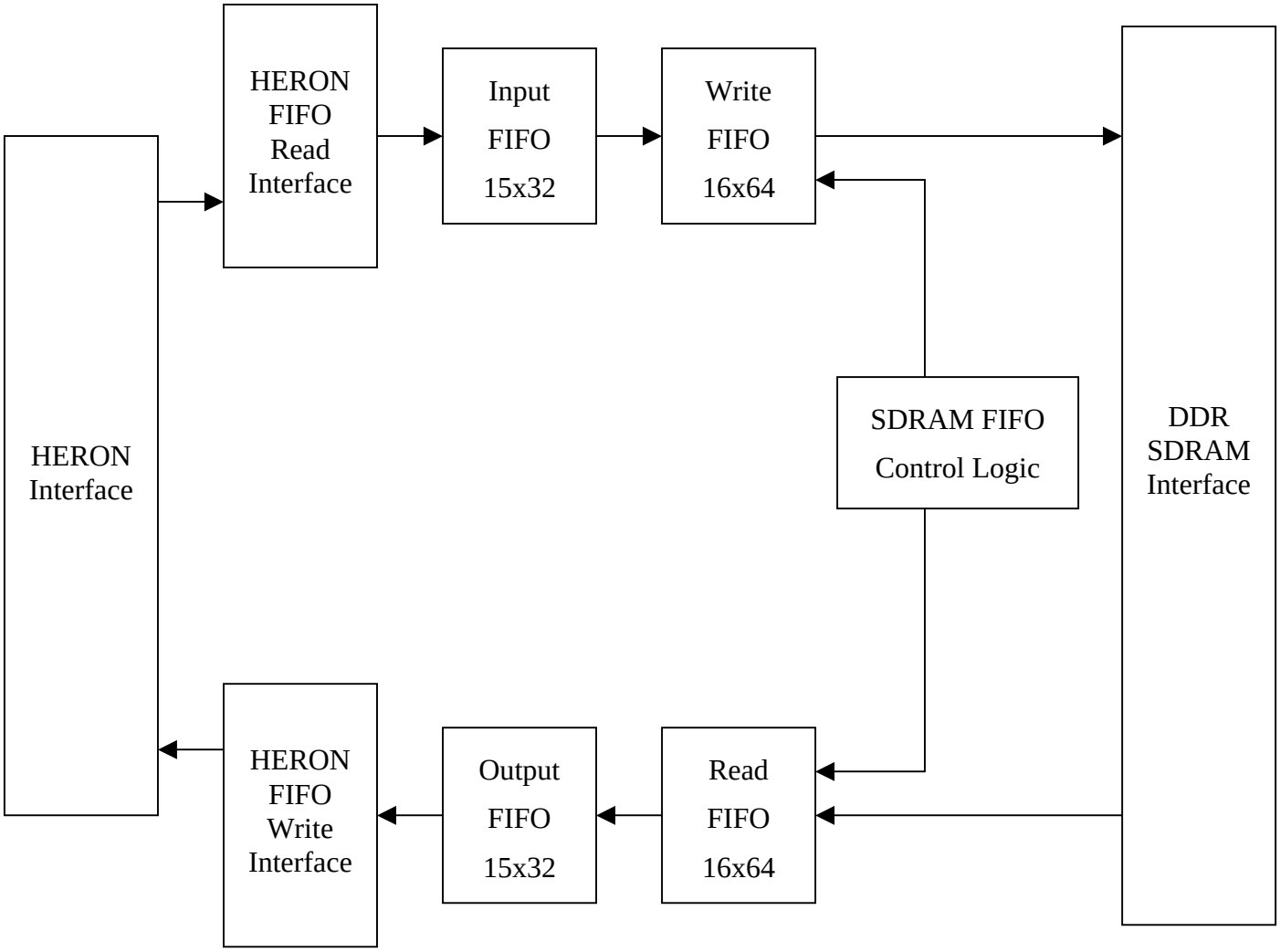
The HERON-FPGA12 is fitted with a 50MHz oscillator which is used to generate a 200MHz clock source for the FPGA. The 200MHz clock source is used to generate all DDR SDRAM clock signals. This is automatically handled by the Hardware Interface Layer.

The HERON-FPGA12 is also fitted with a 100MHz oscillator on OSC3. This clock is used to drive the FIFO clocks directly.

The following bit-stream is supplied as part of the example.

4vfx12ff668.hcb	HERON-FPGA12 fitted to HEART based carrier e.g. HEPC9 (100Mhz FIFO clocks)

FUNCTIONAL BLOCK DIAGRAM



SDRAM FIFO Operation

The HERON-FPGA12 provides a 128Mbyte bank of DDR SDRAM memory. This memory is organised as a 32-bit wide memory interface, with 32Mlocations. The SDRAM FIFO example uses this bank of memory to create a FIFO of 32Mlocations by 32-bits.

All data from HERON Input FIFO 0 is passed to the SDRAM FIFO. Data output from SDRAM FIFO is passed to HERON Output FIFO 0.

Between the HERON input FIFO and SDRAM FIFO are three small FIFOs. This first small FIFO is a 15x32 asynchronous FIFO generated using the Core Generator.

The next two FIFOs are synchronous FIFOs that directly interface to the DDR interface of the USER_AP entity.

Between the SDRAM FIFO and HERON output FIFO are again, three more small FIFOs. The first two FIFOs are synchronous FIFOs to directly interface to the DDR memory, and the third FIFO is the CoreGen 15x32 FIFO.

With six small FIFOs associated with the SDRAM FIFO there is a total FIFO depth from HERON input FIFO to HERON output FIFO of 33554526 words.

Two LEDs are used to represent the state of the SDRAM FIFO, as follows:

- LED 0 is illuminated when the SDRAM FIFO is empty
- LED 1 is illuminated when the SDRAM FIFO is full

For each LED there is a small counter that ensures the LED is illuminated long enough to be seen, even if the full or empty condition lasts for only a few clock cycles.

LED 4 will always flash to indicate that the system FIFO clock FCLK_G is running.

Typical Use of the Example Bit-stream

The typical use of the example bit-stream is to provide a large FIFO, with over 32Mlocations, at 32-bits wide. The SDRAM FIFO reads and writes from HERON FIFO 0.

Therefore, the HERON-FPGA12 can be placed in a system with other modules and by correctly connecting FIFOs from surrounding modules to FIFO 0 of the HERON-FPGA12, a large data buffer can be inserted in the flow of data through the system.

Where are the Bit-streams and Example Program?

The bit streams for this example can be found on the HUNT ENGINEERING CD in the directory `\fpga\fpga12v1\Sdram_Fifo(ex3)`. The name of the bitstream reflects the Xilinx FPGA part number and the Carrier board type, as explained earlier in this document.

An easier way to navigate to the correct directory is to select the “Files” link next to the “SDRAM FIFO” link under the IP sections of the CD browser.

The source files for the FPGA example can be found in the ‘`src`’ sub-directory. The sources in the ‘`\fpga12v1\common`’ directory are also required. A project for the Xilinx ISE development tools is provided for you in the ‘`ISE`’ sub-directory.

FPGA Example Code

You should understand the HUNT ENGINEERING VHDL support for HERON modules before looking at this section. If you do not then please review Example1 again (the 'Getting Started' example for FPGA modules).

As you are expected to be already familiar with Example1, this section only discusses the points that are unique to example3.

For example2 the correct options for a HEPC9 are:

FCLK_G_DOMAIN
True

This will set both input and output FIFOs to be clocked by the same 100Mhz clock signal.

You need to consider the timing constraints that are defined in the '.ucf' file for your design. Actually if you use a time specification that is more strict than needed there is no problem, so the standard '.ucf' file have the FIFO clocks specified as 100Mhz, along with the SDRAM clocks defined as 200MHz. If the project builds (as Example3 does) with this specification it is still guaranteed to work at lower clock speeds. If you add new clock nets into your design then you need to add new timing constraints into your design.

Accessing Memory at High Speed

The HERON-FPGA12 is fitted with 128Mbytes of DDR SDRAM memory. This memory is based on SDRAM technology, and as such, memory is read or written by first opening a row which contains the addressed cells. When the row has been opened the data is then read or written. At the end of the data transfer for that row, the row must then be closed.

It is important when accessing SDRAM memory that many words are transferred while a row is open. This is to ensure that the overhead of the row open and row close operations does not out-weigh the time taken in transferring data.

Conventional SDRAM memory transfers data in bursts of many words where one word of data is transferred on every clock cycle of the memory. 'Double Data Rate' DDR memory doubles this data rate by being able to transfer one word on the rising edge and one word on the falling edge of the memory clock signal.

On the HERON-FPGA12 the user logic inside the FPGA has access to DDR memory via the HE_DDR component provided in the Hardware Interface Layer. The HE_DDR component is designed to access the external DDR memory as efficiently as possible. As such it presents FIFO interfaces for read data and write data that allow bursts to be performed to memory. The FIFO interfaces are organised as rising data FIFOs and falling data FIFOs which makes it possible to burst at high data rates to and from memory.

For memory write operations, the HE_DDR component presents three separate FIFO interfaces. One FIFO for storing the write addresses, one FIFO for storing write rising data and one FIFO for storing write falling data. Similarly for memory read operations the HE_DDR component presents three more FIFO interfaces. The first is used to store the read addresses, the second for storing read rising data and the third for storing read falling data.

The FIFO components implemented inside the HE_DDR component are asynchronous. On the memory side of the HE_DDR FIFOs all accesses are performed from a 200MHz clock signal. On the user side of the HE_DDR FIFOs all accesses are performed at a separate clock rate created by the user.

When accessing memory, the HE_DDR component always bursts at 1.6Gbytes/sec. The HE_DDR component transfers in bursts of four words, two words of rising data and two words of falling data.

The data burst is performed at 200MHz with data transferred on both rising edge and falling edge of the clock. This means that 8 bytes are transferred on every clock cycle, which is a data rate of 1.6Gbytes/sec.

The data rate on the user logic side of the interface FIFOs is set by two factors. The first factor is the clock rate at which the interface FIFOs are accessed and the second factor is whether rising data and falling data FIFOs are accessed sequentially or concurrently.

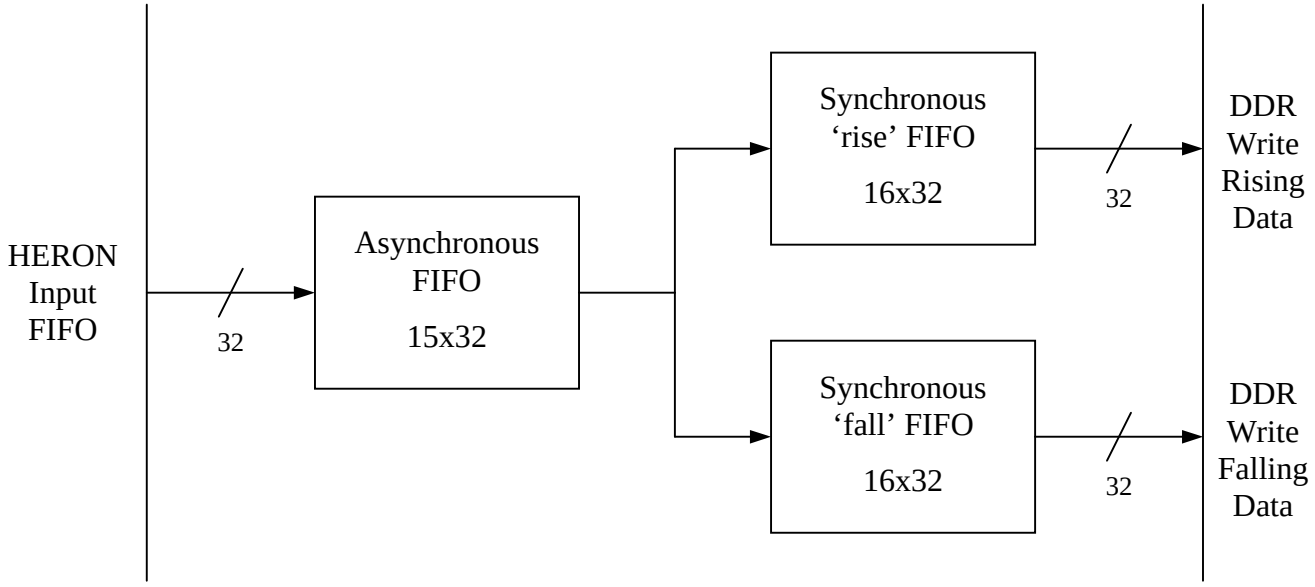
To match the performance of the user logic to the DDR memory, the user clock rate should be set to 200MHz, and the rising data and falling data FIFOs should be accessed concurrently. In transferring to or from the rising data FIFOs and falling data FIFOs in the same clock cycle it becomes possible to transfer 8 bytes every cycle. If this process is performed at a clock rate of 200MHz, the user logic will achieve a data rate of 1.6Gbytes/sec.

In Example3 this is how the HE_DDR interface FIFOs are accessed. In addition to providing a large 128Mbyte FIFO Example3 therefore also provides as an example of how to access the DDR memory at the full rate of 1.6Gbytes/sec.

Writing to Memory at 1.6Gbytes/sec

In Example3, a 128Mbyte FIFO is created by using the bank of DDR memory fitted to the HERON-FPGA12.

Input data arrives via HERON Input FIFO 0. The write process is shown in the diagram below.



Data is first transferred from the HERON Input FIFO to the asynchronous FIFO. This transfer is done in the FCLK_G domain at 100MHz. One word of data can be transferred on every cycle and therefore the data rate at this stage is 400Mbytes/sec.

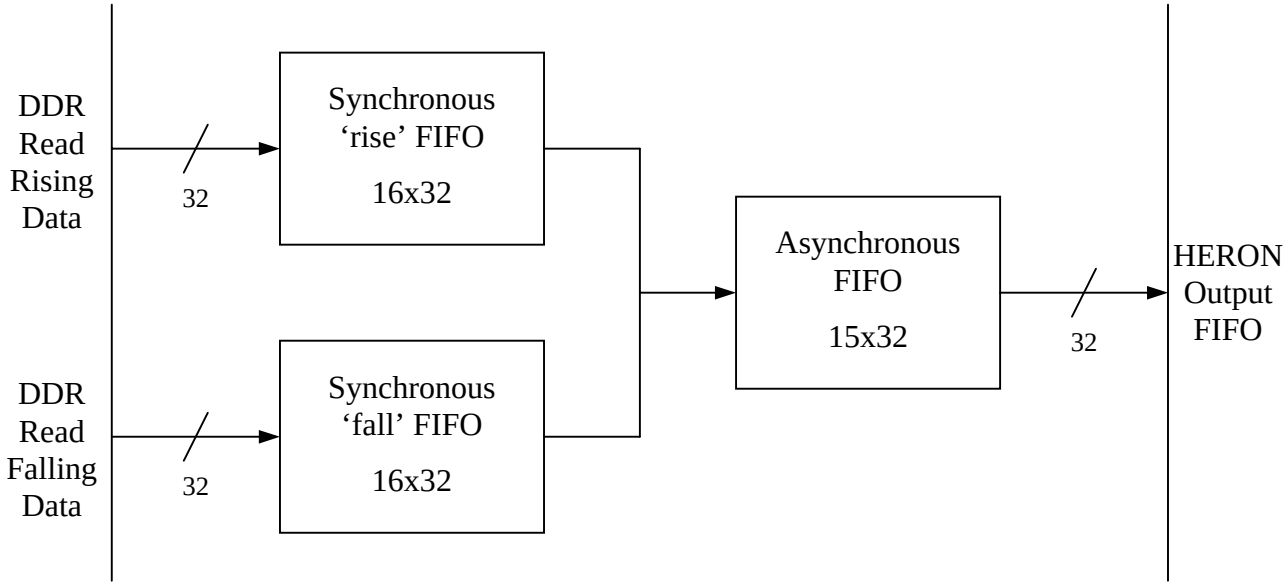
Data arriving in the asynchronous FIFO is then transferred alternately to first the 'rise' FIFO and then the 'fall' FIFO. The read side of the asynchronous FIFO and write side of the synchronous FIFOs are all clocked at 200MHz. One word can be read from the asynchronous FIFO on every cycle and therefore the data rate at this stage is 800Mbytes/sec.

Finally, data in the synchronous FIFOs is transferred to the write FIFO interfaces of the HE_DDR component. The logic at this stage is written such that two words can be transferred on every cycle at 200MHz. The data rate at this stage is 1.6Gbytes/sec. In this stage, counters are used to generate the write address values. Logic is used in combination with the read process to keep track of the amount of data stored in the DDR FIFO.

Reading from Memory at 1.6Gbytes/sec

In Example3, a 128Mbyte FIFO is created by using the bank of DDR memory fitted to the HERON-FPGA12.

Data is output via HERON Output FIFO 0. The read process is shown in the diagram below.



Data is first transferred from the HE_DDR component to the synchronous FIFOs. The logic in this stage is written such that two words can be transferred on every cycle at 200MHz. The data rate at this stage is 1.6Gbytes/sec. In this stage, counters are used to generate the read address values. Logic is used in combination with the write process to keep track of the amount of data stored in the DDR FIFO.

Data arriving in the synchronous FIFOs is then transferred in alternate cycles from first the 'rise' FIFO and then the 'fall' FIFO to the asynchronous FIFO. The read side of the synchronous FIFOs and write side of the asynchronous FIFO are all clocked at 200MHz. One word can be written to the asynchronous FIFO on every cycle and therefore the data rate at this stage is 800Mbytes/sec.

Finally data is transferred from the asynchronous FIFO to the HERON Output FIFO. This transfer is done in the FCLK_G domain at 100MHz. One word of data can be transferred on every cycle and therefore the data rate at this stage is 400Mbytes/sec.

Example3 Performance

The HERON-FPGA12 is fitted with a bank of DDR memory 128Mbytes in size. For this bank of DDR memory, data can be transferred at the rate of 1.6Gbytes/sec.

Data can either be read at 1.6Gbytes/sec or data can be written at 1.6Gbytes/sec. It is not possible to both read and write at the same time.

The HERON Input FIFO interface of the HERON-FPGA12 can transfer data at 400Mbytes/sec. This bandwidth must be shared among the six input FIFOs. In Example3 however, only one HERON Input FIFO is used to transfer data to the DDR FIFO so the full 400Mbytes/sec bandwidth is available.

Similarly, the HERON Output FIFO interface can transfer data at 400Mbytes/sec, but this must also be shared among the six output FIFOs. For Example3 this again means that the 400Mbytes/sec bandwidth is available for the single output FIFO connection.

As the DDR memory is directly interfaced to HERON FIFOs the sustained data rate through one DDR FIFO cannot be greater than 400Mbytes/sec. The HE_DDR interface however is capable of 1.6Gbytes/sec.

In Example3 logic is provided that demonstrates a1.6Gbyte/sec connection to memory within the user logic section of the project, but the full bandwidth is only used between the Hardware Interface Layer (HE_DDR component) and the synchronous 16x32 FIFO components.

Of course, if the HERON-FIFO connection was not directly used, internal FPGA logic could be developed that made greater use of the available bandwidth.