

# Meeting Memory Interface Design Challenges with Virtex-5 FPGAs

Virtex-5 devices support the latest generation of high-speed memory interfaces.

by Richard Chiu  
Staff Applications Engineer  
Xilinx, Inc.  
[rich.chiu@xilinx.com](mailto:rich.chiu@xilinx.com)

When not supporting new interface protocols, memory interface designers are constantly supporting faster and faster bus speeds for existing interfaces. Today's source-synchronous double-data-rate (DDR) memory devices, such as DDR2 SDRAM, QDR II SRAM, and RLDRAM II, present designers with challenges at chip and PCB levels. Higher clock frequencies result in a rapidly shrinking data valid window. Signal integrity issues, clock jitter, memory uncertainties, varying silicon delays, PCB trace skew mismatch, and other factors now have a proportionally larger impact on meeting timing with a smaller data valid window.

## Virtex-5 FPGAs Enhance Memory Interface Design

The Xilinx® Virtex™-4 FPGA family introduced a number of on-chip resources, in particular ChipSync™ technology,

which adds to each I/O block an adjustable delay element (IDELAY) compensated over process, voltage, and temperature changes as well as enhanced DDR capture support. These features help meet the challenges of designing with source-synchronous memory interfaces. With Virtex-4 memory interface designs, you can employ calibration algorithms to factor out many of the skews and delays in the timing path and operate your design at higher frequencies.

The Virtex-5 architecture adds additional features that allow you to push the limits of operating frequency. Enhancements to the Virtex-5 device integral to memory interface design include:

- The addition of ExpressFabric™ technology. This architectural enhancement enables internal logic to run at higher clock frequencies. The basic slice look-up table (LUT) has increased from a four- to a six-input LUT (6-LUT), reducing the number of required logic levels. The technology also offers additional routing

resources to provide more direct routes within a slice and between configurable logic blocks (CLBs).

- Reduction of the maximum bank size from 64 I/O (or 80 I/O in select Virtex-4 part/package combinations) to 40 I/O, and an increase in the number of banks. This leads to a more efficient implementation of the usual myriad of I/O voltage levels on the same FPGA. More I/O clocking resources have also been added to each bank.
- The availability of phase-locked loop (PLL) blocks as clocking resources in addition to digital clock manager (DCM) blocks. PLLs are useful for low-jitter clock generation and input clock jitter filtering.
- Enhanced block RAM/FIFOs that have doubled in size to 36 Kb and support a maximum width of 72 bits. Applications requiring error-correcting code (ECC) detection and correction can now take advantage of ECC encode/decode logic built into each

block RAM, reducing logic usage and allowing much higher performance over implementing the same functionality in general logic.

- Support for digitally controlled impedance (DCI) on-chip split-Thevenin termination for bidirectional I/O only when the driver is 3-stated. Similar to the on-die termination (ODT) feature implemented in many memory device families, this support is provided for certain HSTL and SSTL I/O standards and can be used to save power when the FPGA is writing to memory.
- The incorporation of low-inductance bypass capacitors directly on the package substrate, simplifying PCB layout by reducing the amount of external bypassing required.

**Virtex-5 Data Interface Techniques**

Meeting read and write timing for a high-speed source-synchronous bus demands that you keep uncertainties to a minimum. Typically, the capture of read data is the most challenging part of the design.

Write timing for Virtex-5 FPGAs is supported in the same way as in the Virtex-4 device. The DCM (or PLL) generates quadrature phase outputs of the base (“system”) clock. The memory strobe is forwarded using an output DDR register clocked by an in-phase copy (CLK0) of the system clock. The write data is clocked by a DCM clock output that is 90 degrees ahead (CLK270) of the system clock. This ensures that the strobe is center-aligned to the data on a write at the outputs of the FPGA.

Both Virtex-4 and Virtex-5 memory interface designs support two kinds of read capture techniques:

- The “direct-clocking” technique delays the read data so that it can be directly registered using the system clock in the input DDR flop of an I/O block. The memory strobe is only used during calibration to determine the optimal time to delay the associated data. Figure 1 shows the direct-clocking read capture path.
- The “strobe-based” technique uses the memory strobe to capture correspon-

ding read data and register it with a delayed version of the strobe distributed through a localized I/O clock buffer (BUFIO). This data is then synchronized to the system clock domain in a second stage of flops. The input serializer/deserializer (ISERDES) feature in the I/O block is used for read capture – the first two levels of flops in the ISERDES transfer the data from the delayed strobe to the system clock domain. Figure 2 shows the read capture path for a Virtex-5 memory interface design.

Most Virtex-4 designs use the direct-clocking method for read data capture. Beginning with the Virtex-4 SERDES DDR2 design and continuing with the new generation of Virtex-5 memory interface designs, the strobe-based method is best to meet the tighter timing requirements at higher clock speeds.

Both techniques involve the use of IDELAY elements that are varied during a calibration routine. This routine is performed during system initialization, delaying both the strobe and data to determine and set the optimal phase between strobe/data and

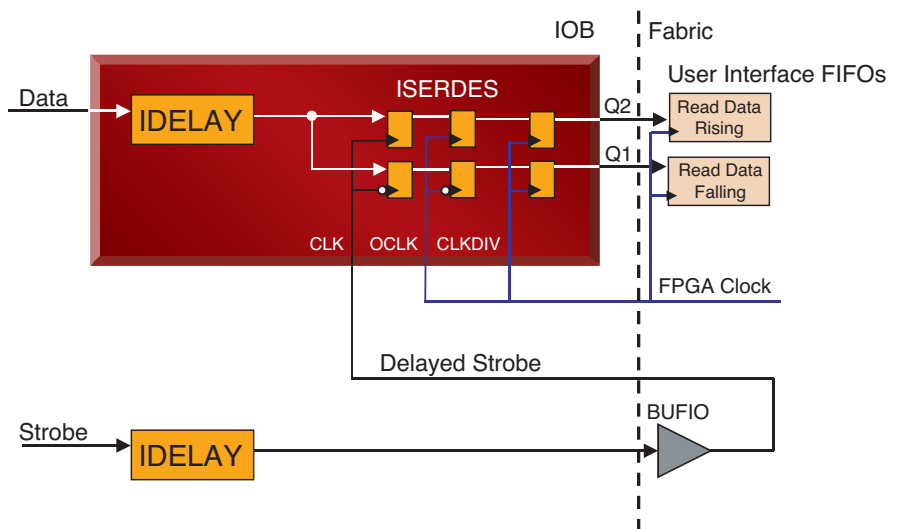


Figure 1 – Virtex-4 direct-clocking read data capture path

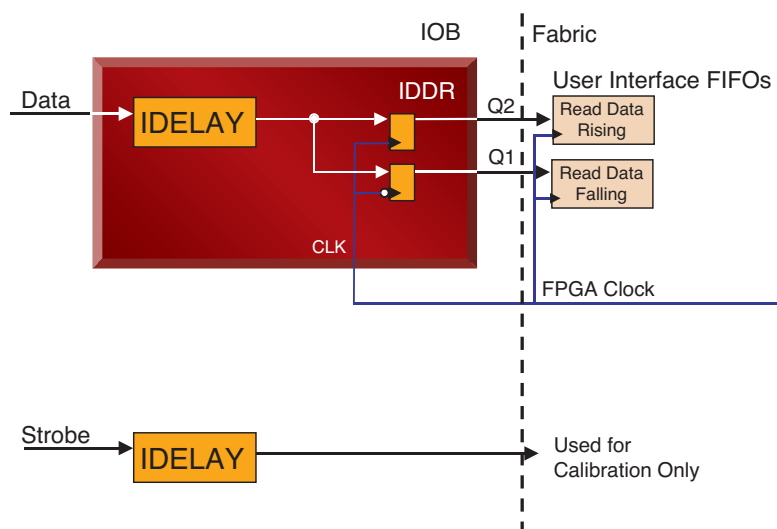


Figure 2 – Virtex-5 strobe-based read data capture path

the system clock to maximize timing margins. Calibration removes any uncertainty caused by process-related delays, compensating for components of the path delay that are static to any one board. These components include PCB trace delays, package delays, and process-related components of propagation delays (both in the memory and FPGA), as well as setup/hold times of capture flops in the FPGA I/O blocks. Calibration accounts for variation in delays that are process-, voltage-, and temperature-dependent at the system initialization stage – you should also factor additional operating temperature and voltage variations separately into your interface timing budget.

During calibration, IDELAY for strobe and data are incremented to perform edge detection by continuously reading back from memory and by sampling either a prewritten training pattern or the memory strobe itself until either the leading edge or both edges of the data valid window are determined. The IDELAY for data or strobe is then set to provide the maximum timing margin. In the case of direct clocking, the optimal delay for the strobe is used to delay the associated data.

For strobe-based capture, the strobe and data can have different delay values because there are essentially two stages of synchronization: one to first capture the data in the strobe domain and another to transfer this data to the system clock domain.

The direct-clocking capture method is simpler in design complexity, and compared to the strobe-based capture method, it has fewer pin-out restrictions. However, the strobe-based capture method becomes necessary at higher clock frequencies. Its two-stage approach offers better capture timing margins for two reasons:

- The DDR portion of the timing is restricted to the first rank of flops in the ISERDES. Because the strobe is used to register the data, timing is limited largely by the strobe-to-data variation; for example, in the case of DDR2, these are given by the tDQSQ and tQHS parameters of the part. For direct clocking, you must

consider the data-to-clock variation (for DDR2, this is  $t_{AC}$ ) because the system clock is used to both drive the memory clock and capture read data. This is a larger uncertainty than the strobe-to-data variation.

- The strobe-to-clock variation is important for the second stage of capture, when the data is transferred from the delayed strobe to the system clock domain. However, by this time the data is split into two separate single-data-rate paths; therefore, aligning the delayed strobe to the system clock can take place over a much larger timing window.

The strobe-based capture method is more pinout-restrictive, as it requires the memory strobes to be placed on clock-capable I/O pins. This can limit the I/O utilization over a given bank. Virtex-5 devices have smaller banks and more I/O clocking resources per bank (for example, the number of BUFIO local clock buffers per bank has increased from two to four), easing this restriction and allowing more strobes and their accompanying I/O (data, mask) to be placed in each bank.

Other significant differences in Virtex-5 memory controllers include:

- Full-speed operation. Both the Virtex-4 SERDES design and Virtex-5 designs use the ISERDES for memory capture. However, Virtex-5 designs do not use the width expansion feature of the ISERDES, and the controller runs at the same speed as the memory clock. The Virtex-4 ISERDES design runs at half the memory clock speed but twice the bus width. Running at the same clock speed as the memory is made possible by the higher performance of the Virtex-5 fabric. This minimizes read-data latency through the ISERDES – as well as controller latency – and simplifies bank-management logic.
- Bank management. The Virtex-5 DDR2 controller employs a least-recently-used (LRU) bank-management algorithm that keeps banks open to reduce the overhead associat-

ed with opening and closing banks. In an LRU algorithm, banks are left open at the end of accesses. If a new bank needs to open, the controller closes the bank least recently used. At any time, as many as four banks can be left open.

### Generating Virtex-5 Memory Designs

You can generate a custom memory controller by using the Memory Interface Generator (MIG) tool. The MIG tool is accessed through CORE Generator™ software and outputs HDL source (Verilog or VHDL) design files, along with accompanying constraint and build scripts.

The latest version of the MIG tool (1.6) supports DDR2 SDRAM-registered DIMM and QDR II SRAM component interfaces for Virtex-5 devices. The DDR2 controller supports operation of bus clock speeds as fast as 333 MHz (667 Mbps). The QDR II supports operation of bus clock speeds as fast as 300 MHz (600 Mbps).

Virtex-5 designs generated by the MIG tool also allow the physical layer interface portion of the design to be easily separated from the controller portion. You can then incorporate your own specific controller but retain the memory initialization and high-performance source-synchronous calibration logic.

### Conclusion

The Virtex-5 device family builds on the Virtex-4 FPGA, with additional features to ease memory interface design and meet the challenges of supporting ever-increasing bus speeds.

To download the MIG tool and for more information about the implementation and design details of Virtex-5 memory controller reference designs, visit the Xilinx Memory Corner at [www.xilinx.com/memory/](http://www.xilinx.com/memory/).

Virtex-5 memory controllers are also available as reference designs for downloading from the Memory Corner:

- XAPP858 (DDR2 SDRAM)
- XAPP853 (QDR II SRAM)
- XAPP852 (RLDRAM II)
- XAPP851 (DDR SDRAM) 