

Capturing Data from Gigasample Analog-to-Digital Converters

Interfacing National Semiconductor's ADC08D1500 to the Virtex-4 FPGA allows quick-start customer application development.

by Ian King
Application Engineer
National Semiconductor
ian.king@nsc.com

Data conversion within the test and measurement domain and communications industry is moving into the gigasamples per second (GSPS) range. Developing a system capable of processing data at these speeds requires diverse engineering disciplines from the initial system concept through to board design, FPGA logic design, signal processing, and application software.

National Semiconductor has developed a leading-edge analog-to-digital (A/D) converter that can deliver as many as three billion samples per second to an 8-bit resolution. One of the main system design questions from customers regarding this product is how data can be reliably captured and processed at this speed. Therefore, National's applications team designed a development platform to provide a solution to this query and demonstrate a reliable data-capture method. This allows the design focus to shift away from the high-speed front end so that developers can focus on their intended application.

The platform also demonstrates that high clock speeds can be reached while maintaining low power dissipation sufficient for the entire system to be housed in a small enclosure, as would be required for a commercial or industrial system. In this article, I'll explain the techniques and analysis involved in achieving this goal.



Power Considerations

When selecting an FPGA for data capture that can achieve low power levels and performance, a 90 nm device is the first choice. In applications where data is captured in bursts (such as oscilloscopes and radar), the static power of the FPGA device becomes an important factor. This is because the high-speed data transfer between devices takes place over a very short time period, so the capture logic will be static while the application consumes the data.

Figure 1 shows a comparison of Xilinx® Virtex™-4 FPGA static power figures over device density. This indicates that the static power is significantly less than the power consumed by the National Semiconductor ADC08D1500 A/D converter, which is typically 1.8W when running from a 1.5 GHz sample clock. Therefore, for systems processing the captured data in bursts, the ADC can be the main source of heat and power dissipation. Having an ADC with low power figures is a key parameter in the design of products, especially those that are required to be small and portable. The design of this development platform confirms that these qualities are achieved by interfacing the ADC08D1500 to the Virtex-4 device.

Data Transmission

The next consideration for systems using the ADC08D1500 and Virtex-4 FPGA is the signaling between these devices. There are two key issues when handling two channels (each providing data at a rate of 1.5 billion (1.5×10^9) conversions per second):

- Signal integrity between the ADC and FPGA
- The rate of data transfer for each clock cycle

The ADC08D1500 uses low voltage differential signaling (LVDS) for each of its data outputs and clock signal. The main advantage of the LVDS signaling method is that you can achieve high data rates with a very low power budget. Two wires are used for each discrete signal that is to be carried across the circuit board, which should be designed to have a characteristic impedance

of 100 Ohms (defined by the LVDS standard). These traces are differentially terminated at the receiver with a 100 Ohm resistor to match the transmission line (see Figure 2).

A signal voltage is generated across the terminating resistor by a 3.5 mA current source within the driving output buffer, which provides a 350 mV signal swing for the receiving circuit to detect. The ADC08D1500 has a total of four 8-bit

data buses, plus a clock and over-range signal that require an LVDS type connection to the FPGA (Figure 3). This adds up to a total of 34 differential pairs, all of which require 100 Ohm termination.

The Virtex-4 device offers active digitally controlled impedance (DCI) and a simple passive 100 Ohm termination on-chip within the I/O buffers of the device. These on-chip termination methods eliminate the need to place passive resistors on

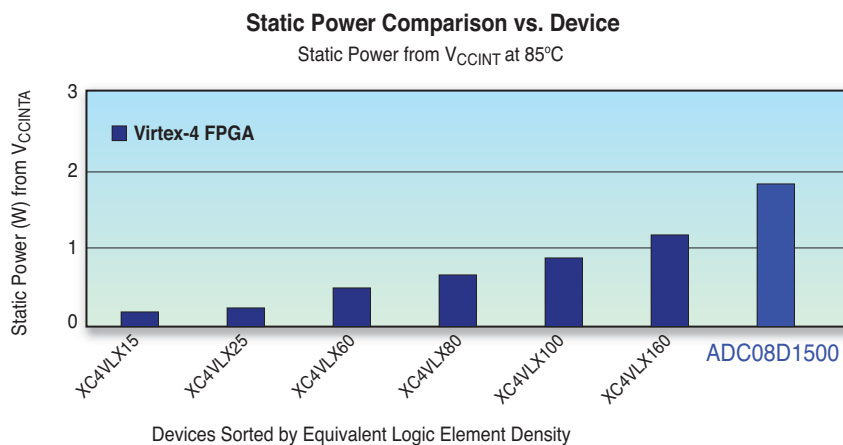


Figure 1 – Comparing the Virtex-4 static power over device density with the operating power of the ADC08D1500

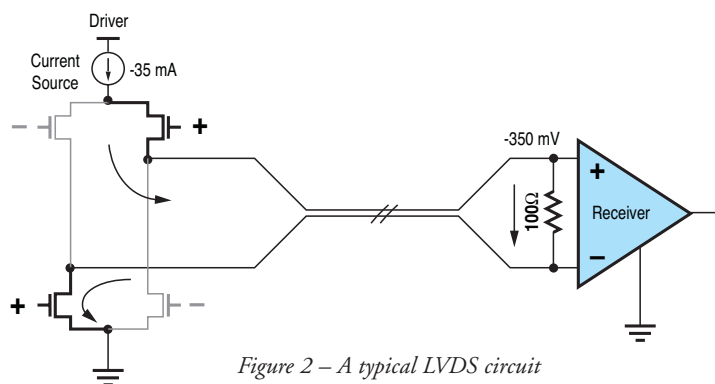


Figure 2 – A typical LVDS circuit

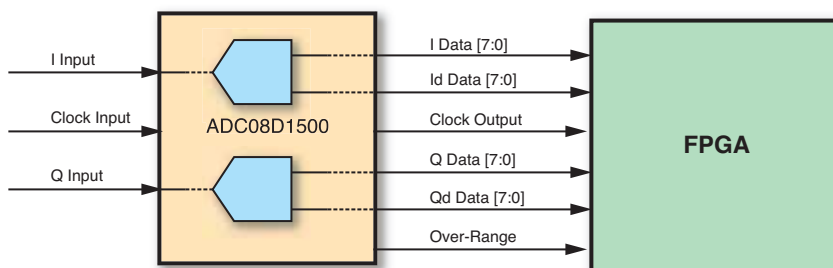


Figure 3 – ADC08D1500 connections to the FPGA

The ADC08D1500 provides a de-multiplexed data output for each of its two channels. Instead of providing a single 8-bit bus running at a data rate equal to the sampling speed, the ADC outputs two consecutive samples simultaneously on two 8-bit data buses (1:2 de-mux).

the circuit board and simplify the routing on the PCB. The DCI option consumes significantly more power than the passive option in this case, simply because of the number of discrete signal lines (68 total) that require termination. Therefore, I would advise turning on the DIFF_TERM feature within each of the IOBs (I/O buffers) to which the ADC signals are connected.

Data Capture

After transmitting data at high speeds using a robust signaling method, it is necessary to store this data into a memory array for post processing. The ADC08D1500 provides a de-multiplexed data output for each of its two channels. Instead of providing a single 8-bit bus running at a data rate equal to the sampling speed, the ADC outputs two consecutive samples simultaneously on two 8-bit data buses (1:2 de-mux).

If the ADC is configured as a single-channel device and put into DES (dual-edge sampling mode), then the sampling speed can be doubled (from 1.5 GSPS to 3.0 GSPS); thus, four consecutive samples are available simultaneously on each of the four buses (1:4 de-mux). This method of de-multiplexing the digital output reduces the data rate to at least half the sampling speed (1:2 de-mux), but increases the number of output data bits from 8 to 16.

For a 1.5 GHz sample rate, the conversion data will be output synchronous to a 750 MHz clock. Even at this reduced speed, FPGA memories and latches would not be able to accept this data directly. It is therefore beneficial to make use of a DDR method, where data is presented to the outputs on the both the rising and falling edges of the clock (Figure 4).

Although the data rate remains the same for DDR signaling, the clock frequency is halved again to a more manage-

able 375 MHz. This frequency is now in the realms of the FPGA IOB data latches. Before this data can be stored away to memory, a small pipeline constructed from a series of data latches is required. Starting with the inputs, for each data line connected to an IOB pair on the FPGA, two latches will be used to capture the incoming data. One latch is clocked on the rising edge of a phase-locked data clock, while the second latch is clocked using a signal that is 180 degrees out of phase.

The relative position of these clocks should be adjusted so that the edges are aligned with the center of the data eye, taking into account the propagation delay of the signal as it enters the FPGA (Figure 5). To simplify this clocking scheme, the Virtex-4 device is equipped with DCMs that allow these clock signals to be generated internally and can be phase-locked to the incoming data clock.

After latching the incoming data using a DCM, the clock domain must be shifted

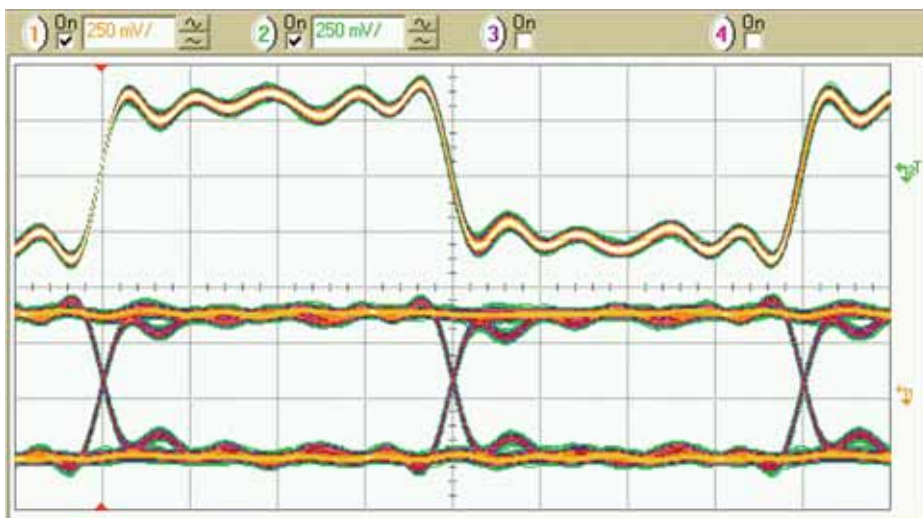


Figure 4 – Oscilloscope plot of clock (top trace) and data from the ADC in DDR mode

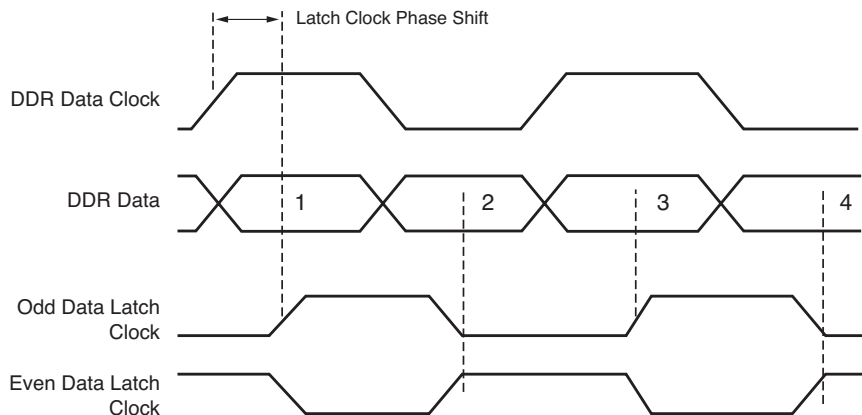


Figure 5 – DDR signaling with DCM-generated data-capture clocks

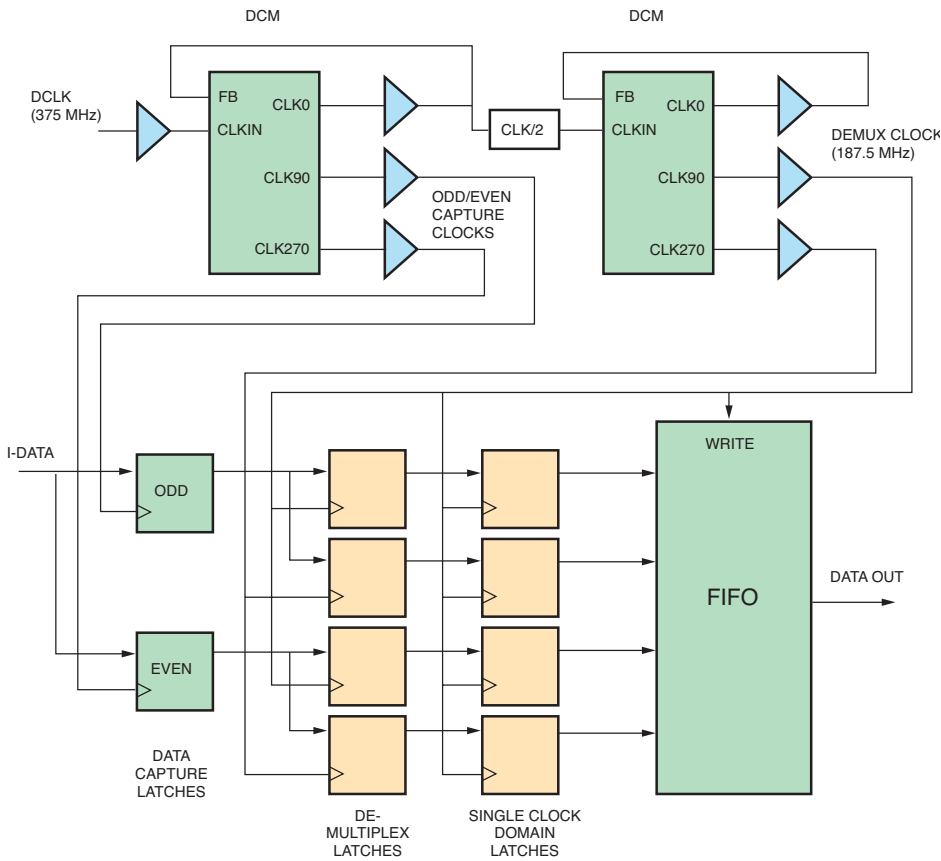


Figure 6 – Data-capture block diagram using two DCMs, latches, and a FIFO memory

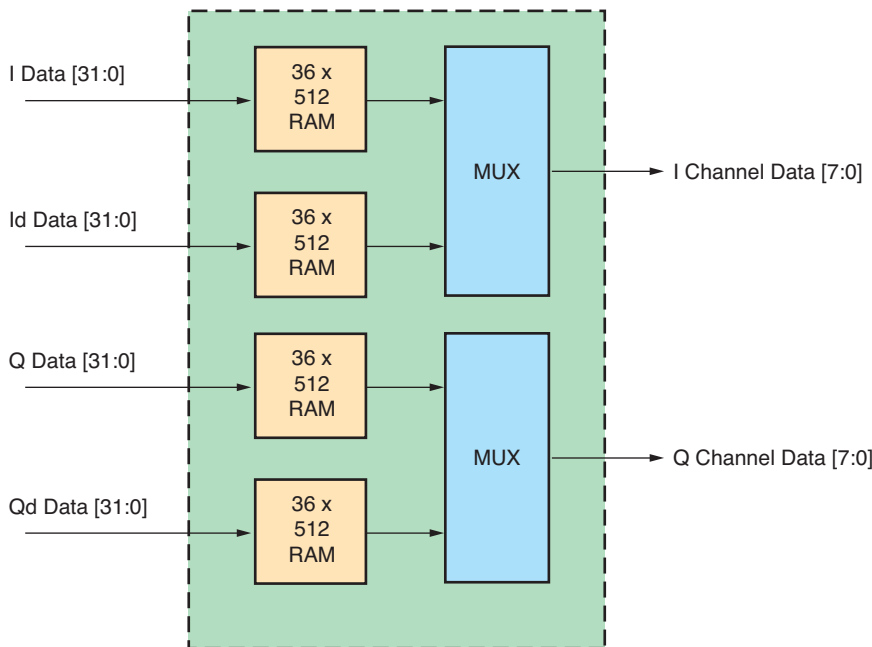


Figure 7 – 128 bit input, 16 bit output, 4 KB deep FIFO

using an intermediate set of latches so that all of the data can be clocked into a memory array on the same clock edge. Because of the speed of the clock, there is not sufficient setup and hold time to re-clock the data; therefore the data must be de-multiplexed again to lower the data rate to 187.5 MHz. Once lowered, the data captured on the out-of-phase clock (even) can be re-captured using the in-phase clock (odd) running at the de-multiplexed rate (see Figure 6).

A second DCM is used to produce the de-mux clock. The clock input frequency is internally divided by two, which produces the 187.5 MHz clock signal. This DCM will provide an output that is phase-locked to the synchronous data clock (DCLK).

Data Storage

As shown in Figure 6, a single 8-bit data bus from the FPGA has been de-multiplexed by four. When all four data buses from the ADC are considered, this method produces a data word 128 bits wide running eight times slower than the sample speed for two-channel operation. The data can now be stored into a FIFO memory buffer.

Creating the custom FIFO for this application is made easy using the Xilinx LogiCORE™ FIFO Generator. Using this software wizard, you can create a FIFO with an input bus width as wide as 256 bits, having an aspect ratio (input-to-output bus width ratio) of 8 to 1. As this design has a 128 bit input bus, the minimum output bus width is 16 bits. This works out well, allowing one 8 bit output bus to be used for I Channel data and the other for the Q channel.

Because the aspect ratio is not 1:1, the FIFO generator will create the memory design using block RAM within the FPGA. A single block RAM can be configured as 36 bits wide by 512 locations deep, so to capture the 128-bit conversion word, the design will use four block RAMs. This gives each channel a 4 KB storage depth without having to cascade FIFO blocks (Figure 7). Having 4K bytes of storage is more than sufficient data for

The low power consumption of the two devices enables systems to operate without forced cooling in small enclosures and does not contribute to a large change in ambient temperature.

a Fast Fourier Transform (see Figure 8) to be applied to the digital conversion of the input signal and represents around 2.7 μ S of time-domain information at the 1.5 GHz conversion rate.

Conclusion

When used for the data capture application described, about 85% of the logic fabric inside the Virtex-4 (LX15) device

low switching noise and to be placed in very close proximity to a high-bandwidth, high-speed data converter without significantly downgrading the measured performance solved my FPGA design challenge.

The two-channel ADC development board discussed in this article is available to order from National Semiconductor in three speed grades: 500 MHz, 1 GHz,

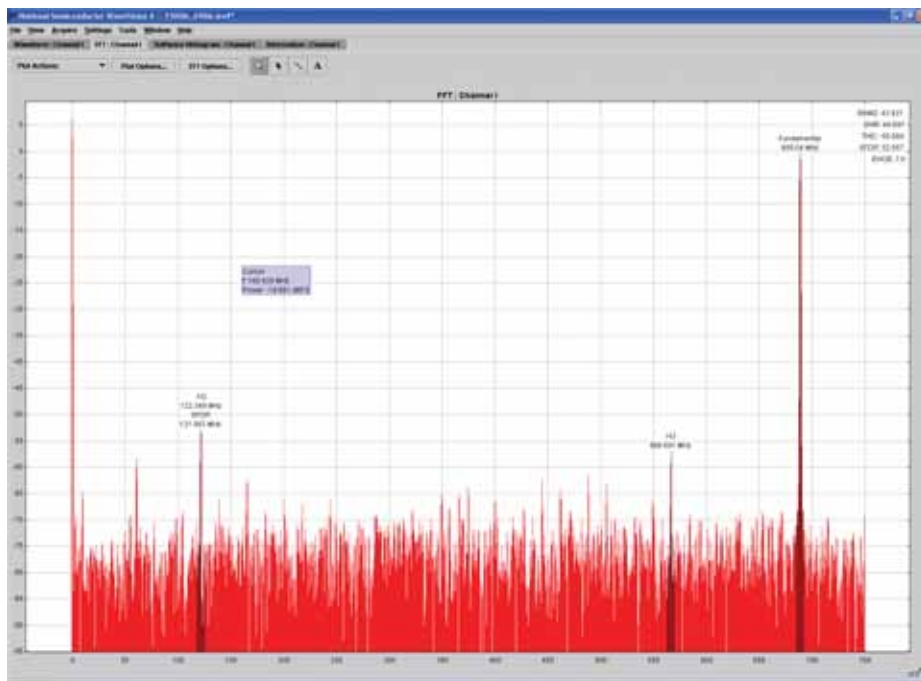


Figure 8 – FFT analysis of 689 MHz input captured by ADC08D1500 and Virtex-4 FPGA

remains available for proprietary firmware development. This leaves space for additional signal processing and data analysis to be performed in hardware, reducing the burden on the software application.

The low power consumption of the two devices enables systems to operate without forced cooling in small enclosures and does not contribute to a large change in ambient temperature. The ability of the Virtex-4 FPGA to operate with

and 1.5 GHz. On-board clocking is provided, so all that is required to get started is to provide an analog signal for sampling, plug in the power supply (included), and connect the USB interface to the host PC.

Single-channel device platforms are also available at 1 GHz and 1.5 GHz sample rates. For more information, visit www.national.com/xilinx and www.national.com/appinfo/adc/ghz_adc.html.

Supporting Your Future
HUNT ENGINEERING
www.hunt-rtg.com

USB connected Programmable FPGA systems

Data Acquisition

- 2 ch 210Mps 12 bit A/D
- 2 ch 160Mps 16 bit D/A
- Configurable digital I/Os
- Virtex-II FPGA (1.5M or 3M)
- USB 2 or Standalone

Software Defined Radio

- Virtex-II FPGA 1M gates
- 2 ch 125Mps A/D and D/A
- TI C6203 DSP
- 32Mbytes SDRAM
- Configurable Digital I/O
- USB 2 or Standalone

Configurable Digital I/O

- Virtex-II FPGA 1M gates
- 90 Configurable Digital I/Os
- RS232/485 and LVPECL
- USB 2 or Standalone

Programmable hardware with cables, device drivers, loading tools, examples and Power Supply. Systems can be used connected to a PC using USB, or can function standalone (without USB) using the initialisation PROMs.

sales@hunteng.co.uk
+44 (0)1278 760188

www.hunt-rtg.com