



New FPGAs Transform Real-Time Systems Architectures

FPGAs have undergone some significant shifts that now influence the way design engineers can take advantage of them for solving dozens of other hardware and software requirements.

By Rodger Hosking, Pentek

FPGAs regularly steal headlines in current industry news for impressive feats of digital signal processing in radar and software radio applications. And, rightfully so! Many new FPGA features added in recent generations of devices from both Altera and Xilinx specifically target these kinds of algorithms. However, other less glamorized FPGA enhancements are dramatically changing the architecture and implementation of virtually all new board-level products for real-time embedded systems.

The lifeblood of today's real-time embedded computer systems is the steady stream of new technology and components. These include transducer interfaces, processors, network interfaces, memory devices, state machines, high-capacity storage devices, DSP engines, timing and synchronization components, parallel digital interfaces, high-speed data links and standard bus interfaces.

As soon as new devices become available, system integrators look for them on board-level products so they can compete more effectively with that new technology. But these devices use high-density packaging that mandates sophisticated electrical, mechanical and thermal design, as well as complex assembly and test procedures in manufacturing. All this leads to longer development cycles—a trend directly at odds with increasingly shorter product life cycles!

Because they are reconfigurable, FPGAs not only support many of the key resources on these boards, but also help extend product life cycles. Following are some specific examples of how and why they are used in products for embedded systems.

SDRAM Memory Interfaces

Synchronous DRAMs offer the densest and most economical solution for large memory arrays. DDR SDRAMs deliver extremely fast read/write cycles to support the latest embedded processors and high-speed peripherals. DDR (double data rate) means data at the I/O pins is always twice the external bus clock rate. Figure 1 shows three generations of these devices, in which the ratio of the internal memory clock to the external clock is 1x, 2x and 4x for DDR, DDR2 and DDR3 devices, respectively.

Internal DDR Type	Internal Ram Clock	External Cycle Time	Bus Clock	Data Rate
DDR-200	100 MHz	10 ns	100 MHz	200 Mbts/sec
DDR-266	133 MHz	7.5 ns	133 MHz	266 Mbts/sec
DDR-333	166 MHz	6 ns	166 MHz	333 Mbts/sec
DDR-400	200 MHz	5 ns	200 MHz	400 Mbts/sec
DDR2-400	100 MHz	10 ns	200 MHz	400 Mbts/sec
DDR2-533	133 MHz	7.5 ns	266 MHz	533 Mbts/sec
DDR2-667	166 MHz	6 ns	333 MHz	667 Mbts/sec
DDR2-800	200 MHz	5 ns	400 MHz	800 Mbts/sec
DDR2-1066	266 MHz	3.75 ns	533 MHz	1066 Mbts/sec
DDR3-800	100 MHz	10 ns	400 MHz	800 Mbts/sec
DDR3-1066	133 MHz	7.5 ns	533 MHz	1066 Mbts/sec
DDR3-1333	166 MHz	6 ns	667 MHz	1333 Mbts/sec
DDR3-1600	200 MHz	5 ns	800 MHz	1600 Mbts/sec

Figure 1 Characteristics of three generations of DDR SDRAM, and types currently now supported by FPGA memory controllers shown in color.

Timing and control of these complex DDRs at these very high rates is extremely challenging. For example, sequential read or write bits for DDR2-800 devices are spaced at 1.25 nsec, and data must be read or written precisely within that time window. For this reason, memory controllers must include high-resolution programmable delay elements and training algorithms, so that optimum timing parameters can be calibrated each time the system is powered up.

This specialized hardware is far beyond the scope of general-purpose configurable logic, so FPGA vendors have added dedicated DDR memory controller blocks to the latest devices. Xilinx now supports DDR and DDR2 devices up to 667 Mbts/s in the Virtex-4 family and 800 Mbit/s DDR3 devices in the Virtex-5. Altera supports DDR and DDR2 devices up to 667 Mbts/s in Stratix II devices and up to 800 Mbts/s in the Stratix III and IV. Both vendors include the self-calibrating timing engines in their memory controllers to help ensure a reliable interface.

Data Converters

Acquiring real-world analog signals from antennas and transducers, and generating analog signals for complementary output signals typically requires A/D and D/A converters. Monolithic devices found in today's embedded systems support sampling at audio rates below 100 kHz for speech, vibration and sonar applications, but commonly range upward into hundreds of MHz for communications, telemetry and radar applications. In some cases, sampling rates extend into the GHz region.

At these very high data rates, two problems immediately surface: how to successfully interface to these high-speed streaming parallel ports, and what to do with the data after it is received. Fortunately, FPGAs offer one of the few viable solutions to both problems.

For example, the Texas Instruments ADS5485 200 MHz 16-bit A/D converter delivers digitized samples using an 8-bit LVDS double data rate output bus operating at 400 Mbts/s.

The latest FPGAs excel at these kinds of fast, custom parallel digital interfaces, featuring user-configurable logic levels to meet a wide range of peripherals. Altera and Xilinx now offer I/O drivers delivering differential LVDS rates up to 1.6 and 1.2 Gbits/s, respectively.

At these high rates, interconnecting traces require controlled impedances, matched lengths and proper termination. To ease these onerous printed circuit board constraints, FPGAs now include per-bit skew adjustments to help align bits in a data word. They also include digitally controlled termination networks for tuning optimum performance while eliminating the need for external discrete resistors.

Clocking

Clock design is one of the most difficult tasks for embedded systems, because of the diverse requirements presented by peripherals, data converters, memory controllers, state machines and system buses. For example, an A/D converter may need to be clocked at a frequency that is locked to an external frequency source. Memory resources may need to operate at a much higher clock rate, often unrelated to the A/D clock. And the system bus, such as PCI-X, must operate at yet a different system-synchronous frequency.

FPGA vendors solve these clock incompatibilities by dividing the FPGA into several clock domains, each capable of operating from independent digital clock generators. Xilinx offers twelve digital clock managers in the representative Virtex-4 and Virtex-5 devices shown in Figure 2. Phase lock loops in these clock managers can slave the frequencies of these generators to external clocks.

	V-4 FX FX100	V-5 LX LX330	V-5 LXT LX330T	V-5 SXT SX240T	V5-FXT FX200T	V-5 TXT TX240T
Logic Cells	94,896	331,776	331,776	239,616	196,608	239,616
Block RAM (Kbits)	6,768	10,368	11,664	18,576	16,416	11,664
Digital Clock Mgrs	12	12	12	12	12	12
DSP48 Slices	160	192	192	1,056	384	96
Power PC Cores	2	-	-	-	2	-
3.75 GHz Serial Ports	20	-	24	24	-	-
6.5 GHz Serial Ports	-	-	-	-	24	48
Gbit ENET MACs	4	-	4	4	8	4
PCIe End- points	-	-	1	1	4	1

Figure 2 Key features of representative Xilinx FPGAs show strong support for block RAM, clock managers, PowerPCs, gigabit serial ports, MACs and PCI Express.

Even though each domain operates independently, data must still flow between clock domains. To solve this problem, FPGAs harness internal FIFO memories that accept data at one rate and deliver it at another rate.

FPGA clock managers also support sophisticated clocking schemes like spread spectrum clocks, where the instantaneous frequency of a clock is randomly modulated around a central value to distribute energy uniformly across a band. This reduces single frequency radiated emission levels for compliance with regulatory standards.

On the other hand, clocks for A/D and D/A converters must have excellent phase noise, low spurious levels and minimum jitter in order to preserve the signal quality characteristics of the devices. Because of their circuit complexity, FPGAs are definitely not recommended as clock generators for high-speed data converters. Instead, external discrete clock drivers and switches should be used for the clock signal path itself; however, FPGA signals may be used to control them.

Triggering and Synchronization

Many embedded applications require strict timing control of digitized transducer signals, such as radar applications that operate in a triggered mode to generate the outgoing radar pulse and then capture return signals during a specified window of time called the range gate. Inherent accuracy of these systems relies on the precise timing of the triggered operations.

In other applications, multiple sensor or antenna signals must be synchronized. Common examples include beamforming, sonar, direction finding, 1D and 2D phased arrays, steered antennas and diversity receivers. In all of these applications, control of the relative phase of the received and generated signal channels is the governing principle of operation.

Because FPGAs are ideal for implementing complex state machines, they are commonly used to generate control signals to serve as triggers and gates for all of these systems. FPGA development tools from every vendor include extensive high-level resources to help simplify state machine design.

Data Buffering

Once high-speed peripherals have been successfully interfaced to the FPGA, the designer must now deal with managing the staggering flow of data to and from other system resources. While A/D and D/A converters operate at a constant clock rate, networks and system buses transfer data in packets or blocks.

Block RAM resources of FPGAs can be used as FIFOs to provide an elastic data buffer for some applications. In other cases, a swinging buffer memory is more appropriate, especially for block-oriented bus interfaces. Also built from FPGA internal block RAM, the swinging buffer allows one memory bank to be filled from one resource (like an A/D converter) while another bank is being emptied by another resource (like the PCI bus). These schemes are extremely effective when the average data rate of peripheral is less than the average rate of the system bus.

However, transient capture applications like radar require a large amount of data to be captured at a very high rate in real time during a range gate, even though the duty cycle of the gate is relatively low. In this case, because FPGA block RAM is too small, external memory must be

used, and the specialized SDRAM interfaces discussed above come into play. In these applications, duty cycle averaging allows the system bus to operate at a much lower speed with no data loss.

For example, a 2 GHz 8-bit A/D converter for radar generates samples at 2 Gbytes/s. For a range gate of 100 msec, the capture buffer size must be 200 Mbytes. If the duty cycle is 10%, the buffer must be emptied once per second at a 200 Mbyte/s rate, which is quite easily handled by the PCI bus.

Gigabit Serial Links

System buses have become serious bottlenecks for system boards because of higher speed peripherals and processors and high-density packaging. Just as desktop PCs are migrating to serial interconnects like PCI Express (PCIe) and Gigabit Ethernet (GigE), today's major shift in embedded system architectures is away from common backplane buses and toward switched serial fabrics and gigabit serial links. The two main advantages are higher speed interconnects and multiple simultaneous paths between system boards and components. More than any other device, FPGAs are the enabling technology for this significant transition. Figure 3 shows the most popular protocols in use and FPGAs support all of them.

Serial Standard	Application Space
Gigabit Ethernet	Computer Networking
PCI Express	Peripheral Interconnect
FibreChannel	Data Storage
Serial RapidIO	System Interconnect
Aurora	Streaming Data
Serial ATA	Data Storage
Infiniband	System Interconnect
Hypertransport	Peripheral Interconnect

Figure 3 Popular serial protocols used in real-time embedded systems and supported by FPGA gigabit interfaces and engines.

Both Xilinx and Altera have incorporated increasing support for gigabit serial links through several recent generations of FPGA devices. The Xilinx Virtex-II Pro was the first device to offer RocketIO gigabit serial transceivers.

They provide the low level electrical interface, the serializer and de-serializer (SERDES), and the 8B/10B encoding engine that delivers clock and data over a single differential pair of copper lines. This interface constitutes the underlying physical and transport layers common to most of the popular gigabit serial standards, including Ethernet, Aurora, PCI Express, Serial RapidIO, InfiniBand and HyperTransport.

Protocol engines for specific standards can be configured using FPGA logic so that FPGAs can adapt to different protocols as required. They interface to the SERDES and correctly process protocol-specific packets, header information, control functions, error detection and correction and payload data format. The strategy makes FPGA-based modules truly “fabric agnostic” and allows one hardware design to be deployed in several different fabric environments.

This flexibility in using one hardware product to cover several different protocols encourages board vendors to develop FPGA-based products for the general market. It also affords system integrators the luxury of not having to commit to any particular standard when selecting boards for their systems.

Since gigabit serial interfaces on FPGAs were so well received, FPGA vendors took the next step and added additional levels of integration to support the most popular gigabit serial protocol: GigE. The Xilinx Virtex-4 and Virtex-5 incorporate four or more 1 GigE media access controllers (MACs) connected to RocketIO electrical transceivers. These MACs offload a significant amount of low-level protocol from configurable logic resources.

In their latest Virtex-5 devices, Xilinx offers RocketIO GTX transceivers with bit rates up to 6.5 GHz. Not to be outdone, Altera now offers their Stratix-IV GX gigabit transceivers with bit rates up to 8.5 GHz.

Xilinx Virtex-5 devices advance this technology even further by including a built-in PCI Express Endpoint engine, while the Altera Stratix-IV GX family features their PCI Express Hard IP Block. Both offerings incorporate key layers of the PCI Express protocol stack. This saves FPGA resources for other tasks and offers a standardized solution for sending and receiving data through one of the most popular interfaces.

Embedded Processors

Starting with the Virtex-II Pro family and continuing through some members of the Virtex-5, Xilinx adds one or more embedded PowerPC processors as a dedicated FPGA resource. Connections to external SDRAM and flash are made through internal memory controllers, while the GigE MAC and PHY interface support Ethernet communications. Capable of executing programs for sophisticated analysis, control and decision-making tasks, this on-chip microprocessor delivers a complete, high-level system on a chip.

One example is a remote data acquisition subsystem connected via Ethernet. A single module can accept commands, report status, as well as capture and deliver data over GigE. Another application is a scanning receiver that looks for radio signals using software radio resources and performs an FFT analysis on received signals. Once a signal is detected, the system can report or create a local log of signal frequency and strength and the time and duration of the transmission.

Because these higher-level functions are complex, they are implemented much more easily as a C program running on the microprocessor, rather than as configurable logic for the FPGA. Also, they can be modified and maintained by software engineers who do not need to be FPGA gurus. This minimizes support costs and significantly extends the life cycle and reusability of a hardware design.

Design Example

Figure 4 shows the Pentek Model 7150 PMC/XMC Quad 200 MHz A/D module as an illustration of how FPGAs dominate the design of the latest technology embedded products. Two

Virtex-5 FPGAs handle critical board functions: the SX95T signal processing FPGA on the left and the FX100T interface FPGA on the right.

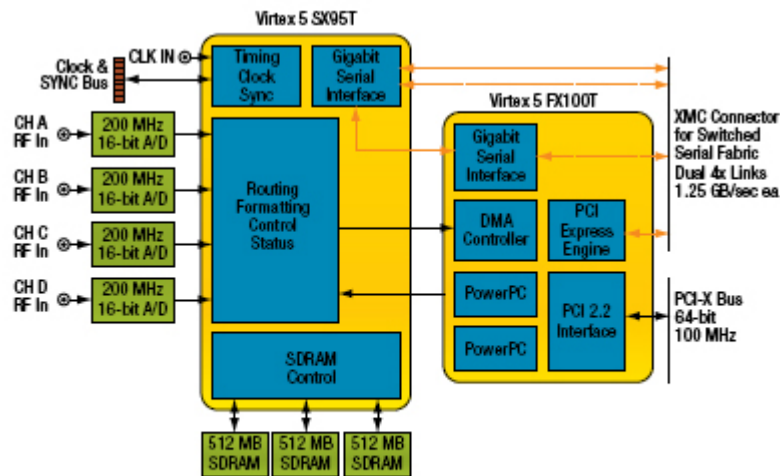


Figure 4 Pentek 7150 Quad 200 MHz 16-bit dynamic A/D XMC module with dual Virtex-5 FPGAs, 1.5 Gbyte SDRAM, gigabit serial links, dual PowerPCs and PCI-X interface.

A sophisticated timing engine supports synchronization, clock control, gating and triggering functions. Four 200 MHz 16-bit A/Ds deliver data to the system through the XMC connectors through gigabit serial interfaces that support GigE, PCIe, Aurora, or other protocols. Alternatively, data can be delivered through a multi-channel DMA controller driving a 64-bit 100 MHz PCI-X bus.

Three banks of DDR2 SDRAM provide an elastic data buffer for averaging data rates and capturing transients. Two PowerPC processors in the interface FPGA can be used as local microcontrollers and for managing an Ethernet stack. It is quite impressive that all of these resources are contained within a single, relatively compact PMC module.

The example above clearly shows that virtually every aspect of the module is implemented with FPGA technology. No other design approach is possible, except for a custom ASIC solution that would be practical only for high volume production. When coupled with their impressive DSP capabilities, FPGAs have clearly revolutionized embedded system board-level product design. As the two major vendors, Altera and Xilinx, continue to compete for design wins by offering new features, better performance, higher density and lower power, designers must constantly keep abreast of frequent announcements to take best advantage of these powerful components.

Pentek
Upper Saddle River, NJ.
(201) 818-5900.
[www.pentek.com].

<http://www.rtc magazine.com/home/printthis.php?id=101078>