Low-cost Spartan FPGAs Used in ADSL Modems

Using Spartan FPGAs to manage DMA transfers and to implement the complex system-level glue logic required for the USB interface—Spartan devices are very cost effective in these applications.

by Marc Baker, Applictions Engineer, Xilinx,
marc.baker@xilinx.com

ADSL modem technology can expand the useable bandwidth of existing copper telephone lines, delivering high speed data communications at rates of up to 8 Mbps. The recent G.Lite standard allows for a lower-speed, lower-cost implementation.

In this ADSL Modem design example (Figure 1), the Spartan device sits between the CPU, USB interface controller, and the ADSL chipset, and manages DMA transfers of ATM cells. While the design is targeted at solving a specific problem, interfacing an ADSL chipset to USB, it illustrates solutions to a number of general technical issues, including the implementation of Utopia interfaces for ATM devices and remote configuration of Spartan devices.

Overview

The design objectives for this application were:

- To achieve the lowest possible cost. In this case, the target cost for the USB interface was significantly less than $10 in high volumes.
- To deliver the best possible performance. Current solutions are able to deliver 2 to 3 Mbps of bandwidth across USB at a much higher price. The minimum target for this design was to support the full 1.5 Mbps data rate of G.Lite and at the same time get as close as possible to the full G.992.2 (6.1 Mbps) data rate.
- To configure the Spartan device from the host via the USB interface. This has the dual benefit of eliminating the requirement for FPGA configuration memory in the modem and the ability to update the configuration in the field.

ADSL Modem System Block Diagram

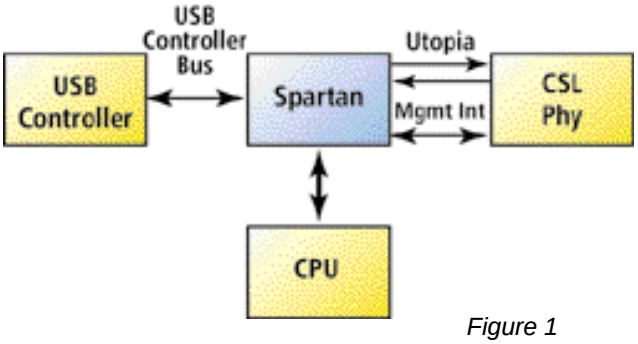

Figure 1

(Continued)

Interfacing ASSPs

Much of the complex logic required for implementing an ADSL modem is provided in highly integrated chipsets known as Application-Specific Standard Products (ASSPs). However, these ASSPs are often not designed to talk to each other. Xilinx low-cost FPGAs, such as the Spartan Series, provide the complex system-level glue required to complete these applications.

The Alcatel DynaMiTe ADSL three-chip set consists of the MT-20134 Analog Front End, the MT-20135 ADSL Modem and ATM Framer, and the MT-20136 ADSL Transceiver Controller. The Spartan device interfaces to the MT-20135, and MT-20136.

The MT-20135 sends and receives data in the form of ATM over a standard Utopia level 2 interface. The function of the Spartan device is to handle the handshaking required to convert the full duplex ATM cell stream from the Utopia interface into half-duplex transfers to and from the USB controller. Because both the MT-20135 and the USB controller have internal FIFOs capable of storing a complete ATM cell, the transfers consist of moving data one cell at a time between these FIFOs.

The MT-20136 is essentially a single-chip processor dedicated to managing the ADSL modem. The Spartan device interfaces to this device via a specialized management interface called CTRLE, used to control the modem and to query status.

The USB interface in the design is based on a National Semiconductor USBN9602 controller. This device contains all of the logic necessary to transfer data frames to and from the host with minimal processor intervention.

The Spartan device supports two primary functions:

- DMA logic that manages the transfer of ATM cells between the USBN9602 and the Utopia interface.
- Arbitrating access to the USBN9602 by an 80C51 microcontroller. The 80C51 provides a low-cost microcontroller with functionality that includes initializing the Spartan device at startup and responding to status queries from the host received via USB messages.

Spartan Device Selection

The following criteria were used to select the device used in this application:

- I/O Pins – The design requires a total of 60 pins.
- Voltage – The design operates at 3.3V.
- Density – The estimated size of the design is 8K gates, broken down as follows: 4K for the USB controller interface, 1K each for the Utopia Tx and Rx state machines, 1K for the microcontroller interface, and 1K for the remaining logic.
- Performance – The highest clock speed used in the device is 48 MHz, used to clock the USB interface bus state machine. The remaining logic runs at 3 MHz (the Utopia data rate) or 16 MHz for the microcontroller.
- Packaging – The size constraints imposed on most modem designs dictate a high density surface mount package.

Based on these criteria the device selected for this design is the XCS10XL-4VQ100C. This device offers 10K gate density, 3.3V operation with 5V compatibility, 77 user I/O, and is packaged in a space saving VQ100 package. The -4 speed grade is sufficient for this design's performance requirements.

Spartan Device Configuration

The microcontroller loads configuration data into the Spartan device using slave serial mode. The data comes from the host via the USB interface. On the host side, the configuration data image is stored in the device driver. The advantages of this approach are that the modem does not need dedicated configuration storage, and updating the configuration is easily accomplished by distributing an updated device driver.

At startup the USB interface pins on the Spartan device are placed into a 3-state condition

(Continued)

and the microcontroller can read and write registers in the USBN9602 by manipulating the I/O port pins. When the Spartan device has been configured and is ready to take control of the USB interface, the microcontroller reconfigures these port pins as inputs to avoid contention.

Interface Architecture

Figure 2 and Figure 3 illustrate how all of this fits together. Sitting in the middle of the interface, the Spartan device is the glue that pulls the whole thing together. The design lets the USB controller and the ADSL modem directly interchange ATM cells with no intervention from the microcontroller. At the same time, the microcontroller has access to control and status registers in all of the devices. This allows the software in the host driver to directly manage the modem.

(Continued)

USB Interface Connections

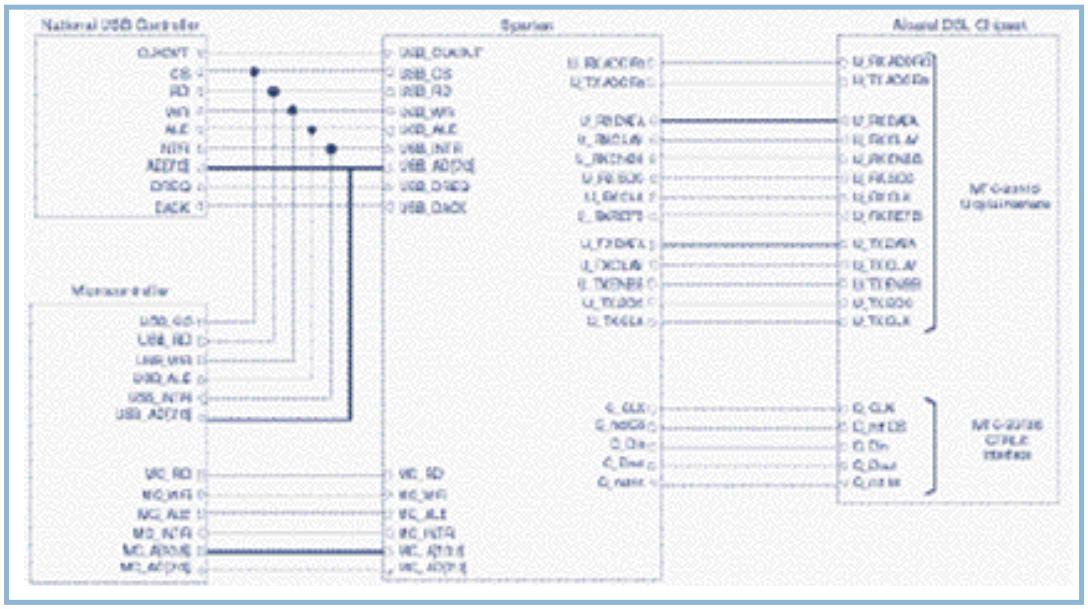

Figure 2

USB Interface Connections

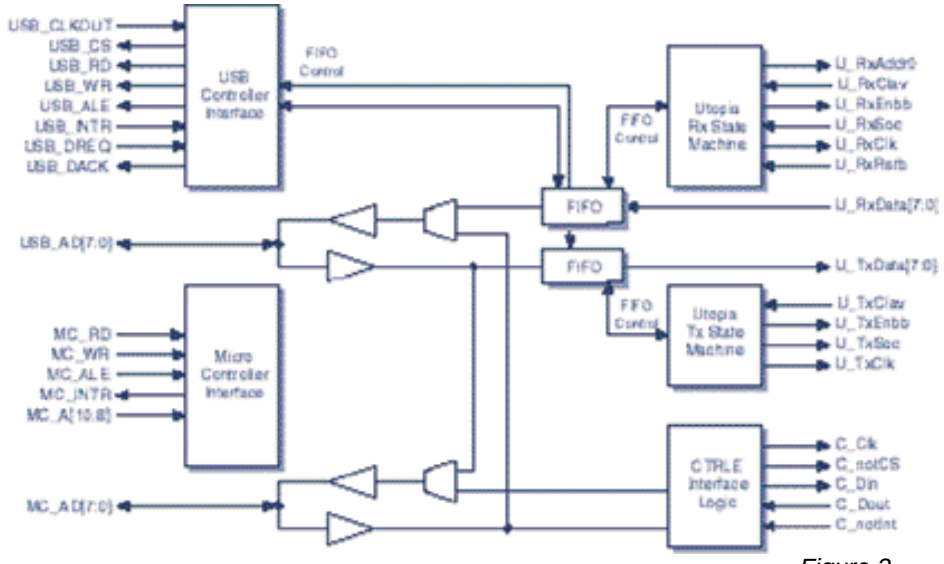

Figure 3

Spartan Device Implementation

The Utopia Tx and Rx state machines interact with the USB interface via 2-word, 9-bit wide interface FIFOs. The ninth bit is set to mark the start of a cell during transfers and reflects the state of the Tx and Rx Start Of Cell (SOC) indicators on the Utopia interface. The use of FIFOs provides a simple means of abstracting the interface between the Utopia state machines and the USB controller interface state machine.

The SpartanXL SelectRAM™ memory is used for the FIFOs. The Utopia Rx state machine continually polls the MT-20135 device to determine if there are any cells in its Rx FIFO. If data is present, it signals this to the USB controller interface and waits for it to indicate that it is ready to transfer a cell to the USB controller. Upon receiving this indication it transfers the cell to the Rx FIFO one byte at a time.

The Utopia Tx State machine continually polls the MT-20135 to determine if there is space for a new cell in its Tx FIFO. When this occurs, it signals the USB controller interface that it is ready to start a transfer. When the USB controller interface is prepared to transfer a cell, it starts loading the Tx FIFO. When the Utopia Tx state machine detects the assertion of the Tx FIFO full flag, it starts transferring the cell data across the Utopia interface. The USB interface arbitrates between the three functions that need to transfer data across the USBN9602 I/O bus. These are ATM cell read operations, ATM cell write operations, and read or write operations initiated by the microcontroller.

The microcontroller interface includes logic for latching the address, and decoding address ranges for the USBN9602 and CTRLE interfaces. In addition it contains a command register for initiating read and write cycles to the USBN9602.

Conclusion

The cost of this solution is well within the design target: $7.68 in 100k quantities (Table 1). The cost of the complete semiconductor solution is $72.68 of which the Alcatel chipset represents almost 90%. Note that the street price of the Alcatel chipset is well below the budgetary number quoted here.

In terms of performance, the design is capable of transferring cells at twice the line rate. While this is just one factor in the overall system performance, we believe that the current limitation in the USB host controllers will be the limitation for system-level performance.

The third objective of in-system configuration was met by supporting the transfer of configuration data across the USB interface. This capability should prove valuable in this and other applications where standard practice, if not the standard itself, is in a state of flux.Σ

For more information on Xilinx applications in digital modems and other high-volume applications, see the Xilinx website at: www.xilinx.com/products/xaw.

ADSL Modem Semiconductor Bill of Materials

Supplier	Part Number	Description	Per System	Unit Cost
Alcatel	MTK-20131	DynaMiTe DSL Modem Chipset	1	$65.00
National Semi.	USBN9602	USB Full Speed Function Controller	1	$1.63
Xilinx	XC510XL-4VQ100C	Spartan FPGA	1	$3.55
Philips	Sc80C51BCCB44	8-bit microcontroller	1	$2.50

Prices are based on budgetary quotes at 100K volume

Table 1