XAPP906 (v1.1) September 14, 2007

Supporting Multiple SD Devices with CoolRunner-II CPLDs

Summary

There has been an increasing demand to add multiple Secure Digital (SD) devices in a single system. Whether the system application calls for a combination of SD memory ports, 802.11 SDIO cards or any other SDIO expansion cards, there is no question that the SD protocol is currently hitting its stride. The problem, however, is that most host devices (i.e. Intel PXA270, TI OMAP, or Qualcomm MSM processors) only provide a single SD interface. Fortunately, CoolRunner-II CPLDs can be used to allow host devices to support any number of SD devices. This application note details a scalable, auto-sensing bidirectional multiplexer based design.

Introduction

Creating an SD Multiplexer Using CoolRunner-II

Figure 1 shows a generalized CoolRunner™-II usage model to incorporate any number of SD ports for a given host device that only has a single native SD interface. The CoolRunner-II CPLD is placed between the host controller and the SD devices. As such, the CoolRunner-II part performs a bidirectional multiplexing function, allowing the host to communicate with any selected SD Device. More importantly, this design has no directional control pins, which means that the CoolRunner-II automatically detects the direction of data flow.

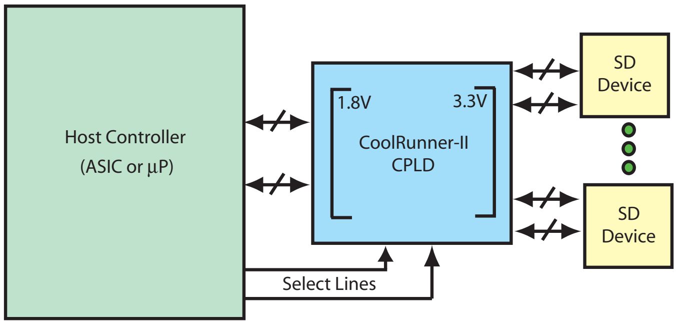

Figure 1: **Using CoolRunner-II CPLDs to provide additional SD ports**

This implementation is extremely flexible and scalable, meaning that the number of SD ports can be increased or decreased as desired. The design also supports any of the defined SD card modes -- SPI, 1-bit, or 4-bit data modes.

While the primary purpose of using a CoolRunner-II device in this type of application is to provide additional SD ports to the host controller, secondary benefits include level translation and logic isolation between the host and the SD card. Figure 1 shows the case where the host is 1.8V, but the SD Devices are 3.3V. CoolRunner-II CPLDs provide negligible standby current and ultra low dynamic power consumption. Hence, incorporating a CoolRunner-II CPLD will not have a significant impact upon your power budget.

Compliance With the SDA Specification

The SDA specification states that one SD bus can only support one SD device. The clock pin can be shared, but DAT[3:0] and CMD lines must be unique for every SD device. See Figure 2 for additional details.

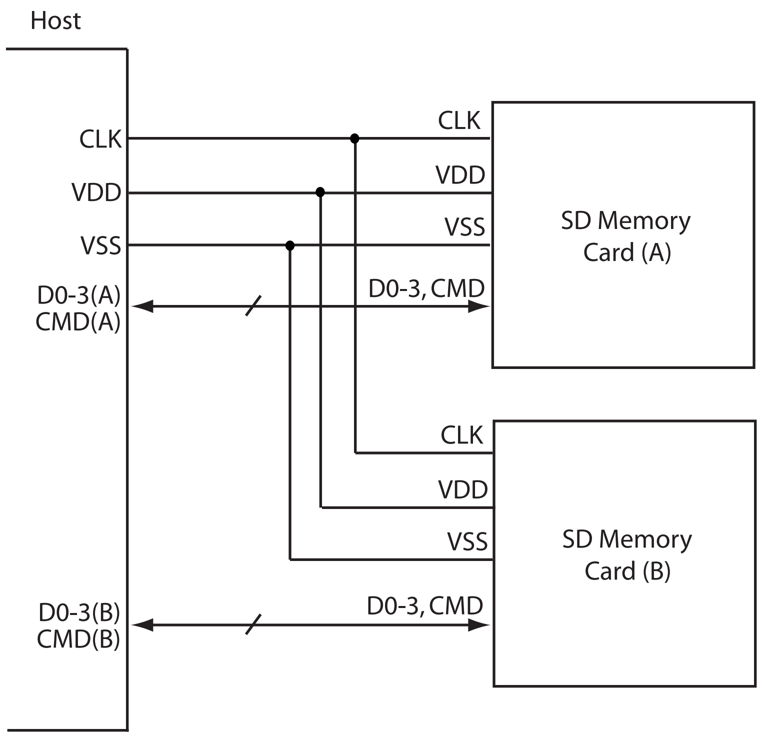

Figure 2: **SD System Bus Topology**

This reference design is fully compliant with the SDA Specification. The following section will show you how to satisfy the above requirements while supporting any number of SD devices using a controller with a single bus.

CPLD Design

Block Diagram

A block diagram showing typical use of this design for two SD devices sharing the same SD host interface can be seen in Figure 3. Conceptually, the design can be viewed and used as a bidirectional multiplexer. The host device controls the CoolRunner-II CPLD via the 'Select' signals, thereby dictating which SD device to communicate with. Once an SD device has been selected, the logic in the CoolRunner-II device automatically detects the direction of data flow, and allows data to flow accordingly (either from the host to the SD card or from the SD card to the host). A directional control pin is not required, thereby making this design easy to use.

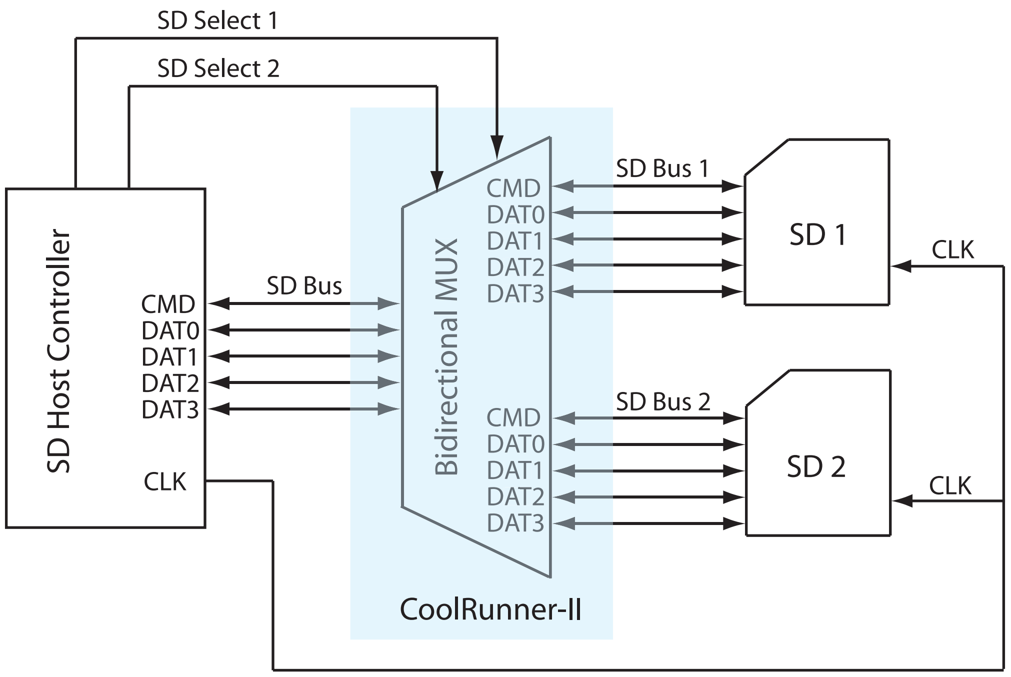

Figure 3: **Block Level Diagram: A Bidirectional Multiplexer**

The host can access each SD device individually without affecting the state of the other when the multiplexer is switched accordingly. If neither the host nor the SD is driving data, the CoolRunner-II CPLD allows the system to be in the default high impedance with weak pull-up state. The primary purpose of this circuit is to provide additional SD capability to the host, but this circuit can also be used to provide level translation and/or logic isolation.

Implementation Details

Figure 4 shows the actual logic circuit for a 1:2 bidirectional multiplexer design, which can be found described using VHDL (see "VHDL Download"). In the initial condition or idle state, the Host and SD Cards should be high impedance with a weak pull-up. Hence, the circuit in Figure 4 is designed to tristate the CoolRunner-II device's output buffers, thereby allowing the external pull-up resistors to take effect. Register A (A_REG) and Register B (B_REG) are both designed to be initialized to logic '0' upon power-up.

The SD Cards are selected via the 'Select' inputs to the CPLD. When 'Select' is logic '0', SD1 is chosen and when 'Select' is logic '1', the SD1 device is chosen. For simplicity while describing this circuit, let us assume in the following discussion that the Host is only choosing to communicate with SD1.

The autodirectional control aspect of this design is implemented in the following manner -- A transaction is initiated when either the host or SD1 drives Low. For example, if the host wants to send data to the SD1 device, the host would begin by driving the A side Low. Upon driving Low, the logic in the circuit detects the Low going edge and responds by enabling the 'B' output buffer, but continues to keep the 'A' output buffer disabled. Specifically, when A is driven Low, a rising edge is delivered to the clock input of A_REG. After clocking, A_REG's Q output becomes logic '1' and therefore prevents B_REG from receiving a clocking event. In parallel with the A_REG clocking and triggering, gate B1 outputs a logic '1' when A goes Low. This enables the 'B' Output Buffer and, ultimately, B will follow A and drive Low.

Conversely, when it is driven from Low to High, gate B1 outputs a Low and tristates the B output buffer. This forces B to go High (via the external pull-up resistor). Once the A and B sides are both High, A_REG and B_REG are reset to 0. This process is repeated indefinitely. The reverse happens when SD1 attempts to drive data toward the host. Additionally, if the host wishes to communicate with the SD2 device, the 'Select' inputs to the circuit are set to a logic '1' and the sequence of events are similar to the above.

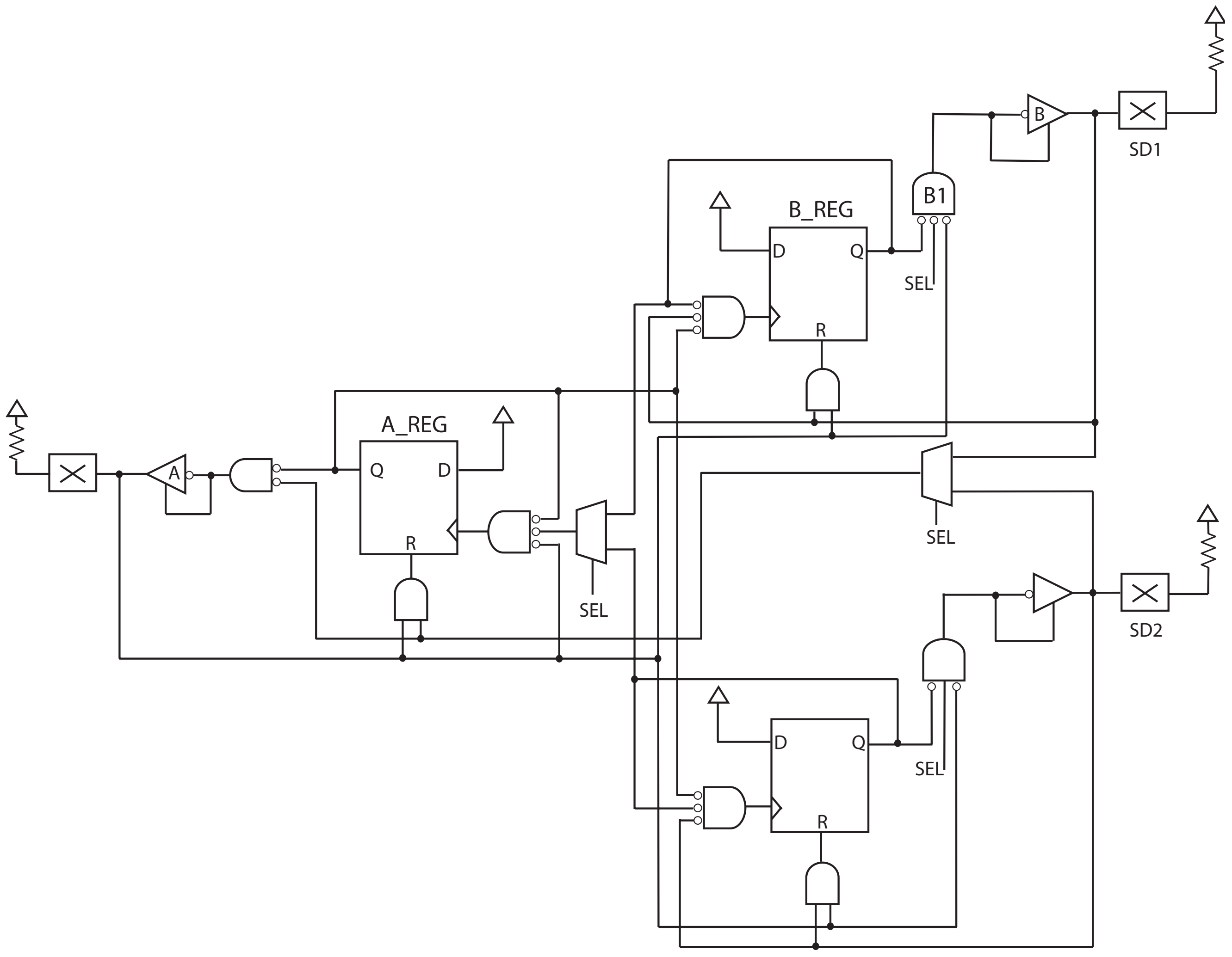

Figure 4: **SD Multiplexer Circuit for Two SD Devices**

Design Verification

Simulation Results

Functional and timing simulations have been extensively performed on this circuit using ModelSim, and test stimuli have been included with the VHDL download. Figure 5 shows some simulation results.

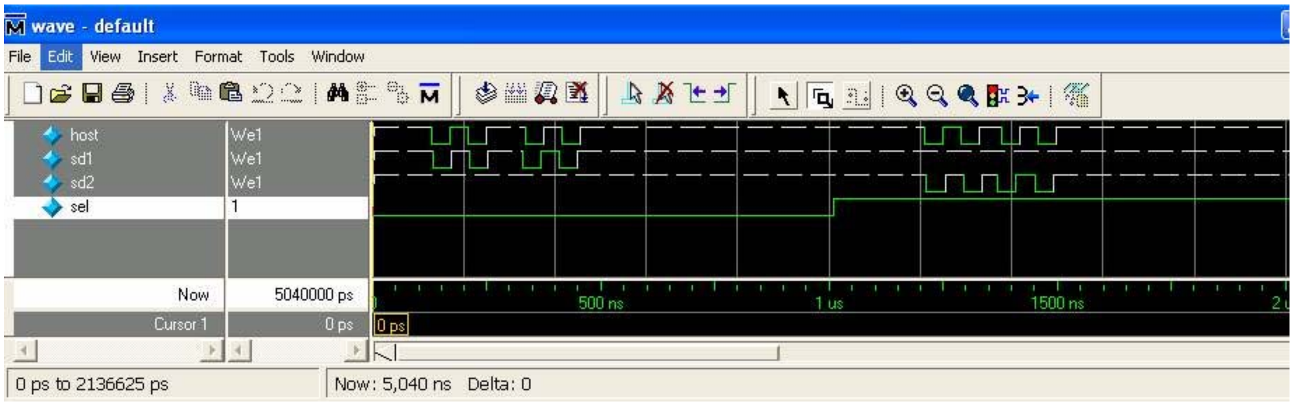

Figure 5: **Simulation Results**

In the first part of Figure 5, the Select input is held Low. A dotted white line denotes a "Weak 1" condition, or, in other words, represents a pulled up state. In the first transaction, the host attempts to drive data toward SD1, and SD1 follows accordingly. Immediately after, the SD1 device attempts to drive data toward the host, and the host follows. Similar events happen when the Select input is driven Low. The host drives data toward the SD2 device, then the SD2 device drives data toward the host.

Hardware Results

An SD Multiplexer demo board has also been created, and has been used to verify this design. Figure 6 shows the demo board, which features an XC2C32A located in the center. Two SD card sockets are located along the top edge of the board. The bottom-most portion of the board is designed to mimic the physical dimensions of an SD card. Figure 7 shows the demo board plugged into a USB SD Card Reader. As expected, the XC2C32A allows a PC to communicate with either the first or second SD card flawlessly.

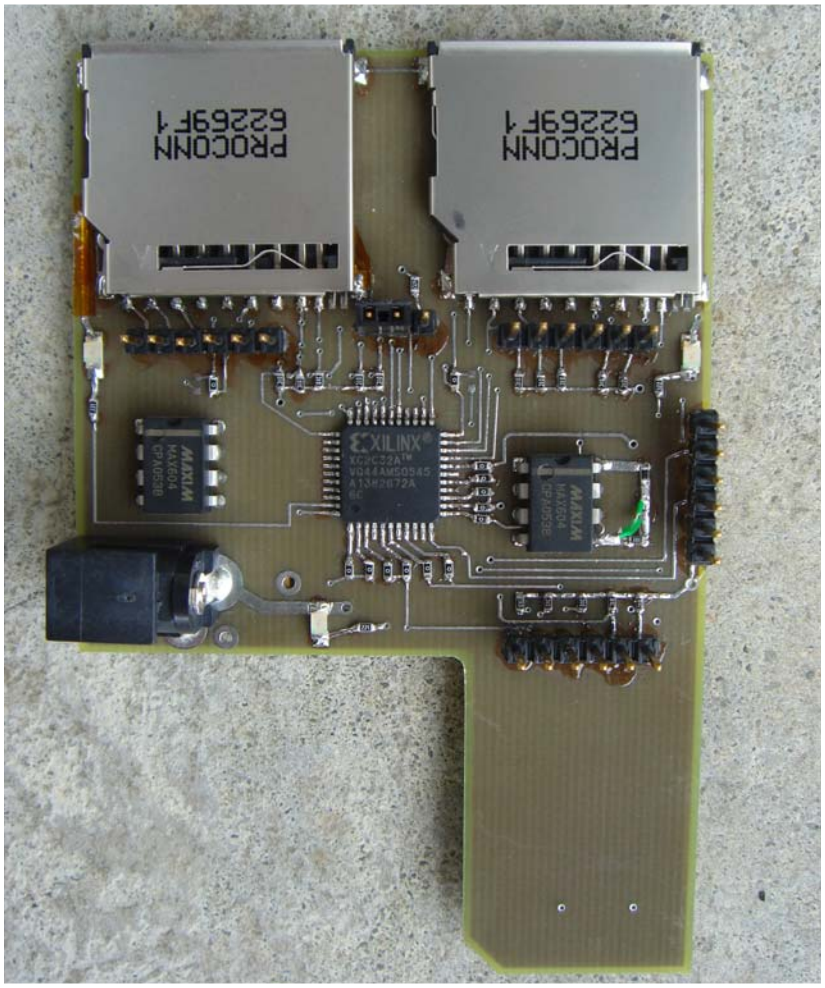

Figure 6: **SD Multiplexer Demo Board**

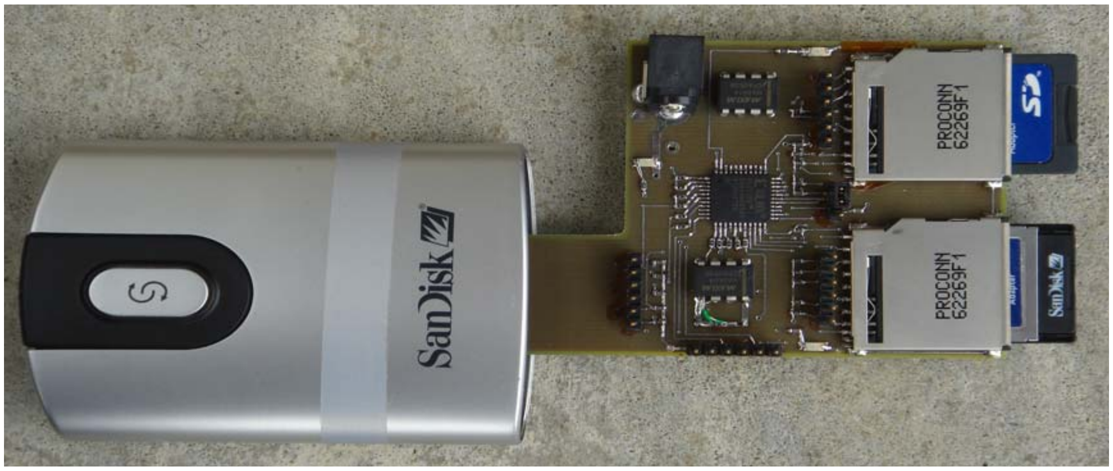

Figure 7: **Demo Board Plugged into USB SD Card Reader**

Device Utilization

Table 1 shows device utilization statistics for various implementations. As stated in the SDA specification, there are three signaling modes defined for SD cards: SPI Mode, 1-bit SD Data Transfer Mode, and 4-bit SD Data Transfer Mode. This design can be easily adapted for any chosen mode. The design can also accommodate any number of SD expansion ports.

Table 1: **Device Utilization Statistics for Various Implementations**

Number of SD Expansion Ports	Device	Macrocell Utilization (SPI or 1-Bit Data Mode)	Macrocell Utilization (4-Bit Data Transfer Mode)
1	XC2C32A	13 Macrocells	19 Macrocells
2	XC2C32A	21 Macrocells	30 Macrocells
3	XC2C64A	27 Macrocells	39 Macrocells

Voltage and Current Considerations

The SDA Specification contains stringent voltage and current requirements for SD cards. CoolRunner-II devices are ideal for this application because they are extremely low power and have features such as I/O Banking. The CoolRunner-II I/O’s can be configured as 1.5V, 1.8V, 2.5V or 3.3V allowing them to interface to any SD device. CoolRunner-II CPLDs also contain I/O Banks, which allow for voltage translation capabilities between the processor and SD card.

The extremely low power nature of the CoolRunner-II family allows for standby operation as low as 15 μA. The addition of a CoolRunner-II device in a system will minimally impact the current budget.

VHDL Download

The VHDL files to compile and simulate these designs are located at:

XAPP906_Design_Files

Conclusion

As SD devices gain in popularity, the need will increase for ways to support more than one SD device with host controllers. This application note provides a verified solution to the problem at hand. This solution will give designers the flexibility to implement two or more SD devices into a system.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
09/28/06	1.0	Initial Xilinx release.
09/14/07	1.1	Correct link to design files.