CONSIDERATIONS FOR THE DESIGN OF A REUSABLE SOC HARDWARE/SOFTWARE DEVELOPMENT BOARD

Authors: Jonah Probell and Andy Young, design engineers, Lexra, Inc.

A Hardware/Software Development board is useful throughout the product design cycle. The design cycle of modern SoC products typically includes an architectural design phase, a hardware design phase, and a software development phase. Design teams no longer treat these as consecutive, but execute them in parallel in order to meet time-to-market goals.

Hardware designers can often create rough RTL models in a matter of days. While verification and debug of the hardware may take weeks or months, once an RTL model is written, it can be implemented almost immediately in PLDs on a Development Board. This allows software developers to begin running code on a system, which may be connected to real-world external stimuli. Rapidly prototyping a working system can uncover architectural flaws and unexpected data flow bottlenecks.

While simulation is a valuable tool for design verification, it may be difficult or time consuming for a verification engineer to create an accurate transactor model to simulate the external stimuli that the SoC will encounter. A Hardware/Software Development Board allows the SoC prototype to be connected to real-world stimuli in the lab. For example, a networking device prototype could be connected to an Ethernet link or a DSP device prototype could be connected to a digital to analog or analog to digital converter.

The most useful configuration for a reusable Hardware/Software Development Board has exactly 2 PLDs. This allows reused, pre-verified, or IP (such as a CPU) design blocks to be implemented in one device while the hardware under development is implemented and may be frequently updated in the second device during debug. Separating the two eliminates the risk of breaking stable pieces of the design as the volatile pieces are recompiled. This ensures a stable CPU configuration, which is important for software engineers, running code under development. Separating the infrequently and frequently compiled pieces of the system also allows for faster compile times for the pieces being debugged.

With the large (and increasing) amount of logic available in today's high-end PLDs, two can hold most SoC designs. While having three or more PLDs on a Development Board can increase total connectivity on the board, it is not usually worthwhile. The greater concern is interconnectivity between any two PLDs, which is limited by available pin count. Partitioning a design into an increasing number of pieces becomes more complex, and any fixed interconnection of three or more PLDs makes the development board in general less flexible.

Let's call the set of interconnections between the two PLDs the ProtoBus. Each ProtoBus net should have a header pin to which a scope, logic analyzer, or external device may connect. The ProtoBus nets may be used to implement a System bus or a CPU coprocessor interface between the two PLDs, or may be used for connecting to external input or output devices. The ProtoBus should also be connected to the I/O pins of a socket for a packaged ASIC. This can be used for testing

hardware within an ASIC test chip. Furthermore, the PLDs should be socketed for upward size migration as logically larger parts become available from the manufacturer.

<ProtoBus_diagram_here>

If time to market pressure and mask costs are not prohibitive, an ASIC test chip can be laid out prior to tape-out of the final SoC product. The choice of an ASIC test chip package and corresponding socket may be a difficult one. The package should be one that is readily available and a common industry standard. This will ensure that the same package can be reused for other designs. It should also have enough pins for all of the signals that will be brought in and out of the test chip. This is usually a greater number of signals than will exist in the final SoC package. The test chip package need not be the same as that planned for the final SoC, unless thermal or package performance effects are to be prototyped.

An ASIC test chip, containing the reusable piece of an SoC, can be valuable for a software development system. It allows software to be run at speed within the test chip since a PLD implementation typically cannot run at the speeds required of the final product. Flexible board design, which is important for a reusable development board, often requires decisions that cost speed on the board. For example, the connections to two PLD sockets, a test chip socket, and a header pin for each ProtoBus net each look like a stub, which can cause signal interference from signal reflections at high-speed. Though it may not be possible to run the test-chip external devices at the speed of the final system, the ability to run code at speed on the CPU within the test chip is valuable.

On-chip-debug, which is built into many embedded CPUs, is an invaluable tool for software development. A Hardware/Software Development Board should have an on-chip debug/trace connector attached to the CPU PLD. Also important for the bring-up of many modern SoC designs is a serial port, indicator LEDs, a PCI connector, DRAM, and ROM. Adding other memories or external interfaces for a particular SoC design will cost PLD pins and make the Development Board less useful for other designs.

For standard interfaces such as PCI or DRAM, it is advisable to have a direct connection to a PLD without sharing the interface signals over the ProtoBus. Though this takes PLD pins away from the ProtoBus these interfaces require such connections. Dedicated pins on one PLD for interfaces other than the ProtoBus mean non-ProtoBus pins will be available on the other PLD. Those pins, like the ProtoBus nets, can be connected to headers and used as an interface to an external device or for probing internal signals with a scope or logic analyzer.

Having a PCI card-edge connector on the board is particularly important because it allows for connections to numerous off-the-shelf interfaces. Even if it is not a part of the final design, an ethernet connection is invaluable for uploading large amounts of code or data under development. Furthermore, many SoC prototypes can benefit from a USB, FireWire, disk controller, video card, or any number of other such PCI devices.

While the ability to bring signals out of the chip is useful for hardware debug, it can also be a helpful tool for software characterization and optimization. A simple memory mapped device on the CPU's system bus can be used to toggle signals to probe points and/or LEDs on the board. By adding a single system bus write instruction, one can use the LED to determine if/when the loop is executed.

Turning the LED on at the beginning of a loop and off at the end can be used as an inexpensive, non-intrusive means of code profiling on the fly. One can use the LED's average intensity, or measure the duty cycle more accurately with an oscilloscope, to determine what percentage of time is spent in the loop. Multiple LEDs can be used to measure and compare time spent in various loops.

Early in the SoC design cycle a CPU implemented in the first PLD may be used by software developers to bring up an operating system while hardware engineers design devices such as system bus peripherals or coprocessors. Hardware engineers can use the ProtoBus headers as probe points for debugging their design while software engineers use the on-chip debug/trace connections to debug their code.

Once the operating system is booting and the peripheral hardware can interact properly with the CPU, software developers can begin writing application specific code that uses the peripheral hardware while the hardware designers debug hardware/software interaction problems with the peripheral blocks.

After the working system has been prototyped, the CPU and as much as possible of the peripherals can be implemented in an ASIC test chip. This can be plugged into the test chip socket on the development board and can be run at speed. The test chip may be designed to rely on peripherals implemented in the PLD. This is especially useful if the SoC prototype relies on peripherals such as memory controllers or a PCI bus interface, which may not exist in the final SoC.

Lexra, a leading provider of microprocessor IP, has designed a reusable Hardware/Software development board, which conforms to the guidelines described here. An SoC design team interested in purchasing such a board should contact their local Lexra sales office.

Waltham, Massachusetts: 781 899-5799 x600
San Jose, California: 408 573-1890 x601
e-mail: sales@lexra.com