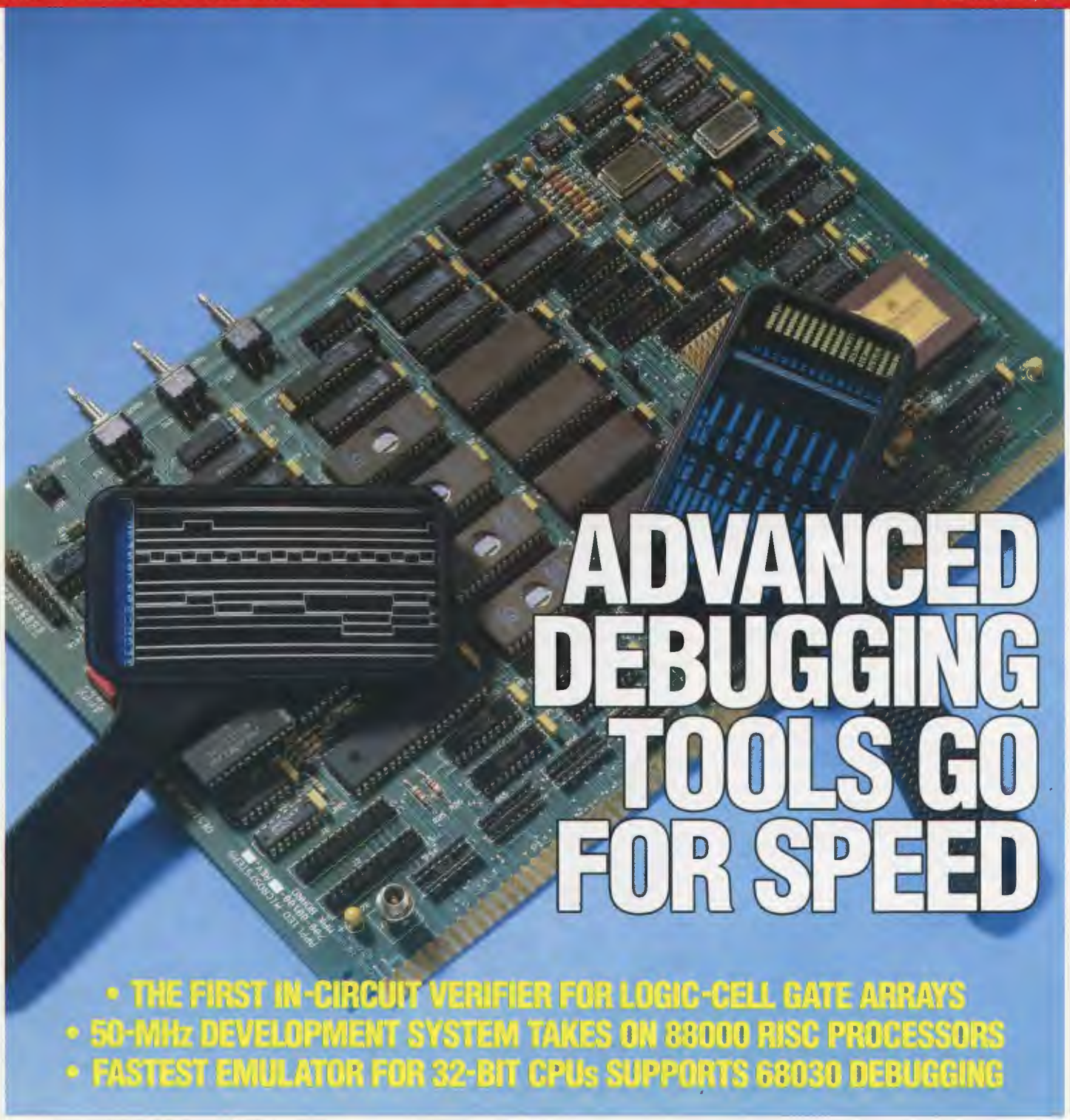
HIGH-RES DISPLAYS COME TO LAPTOP COMPUTERS
HOT NEW WORKSTATION RUNS SUN-4 SOFTWARE

ELECTRONIC DESIGN

A VNU BUSINESS PUBLICATION IN U.S. $5.00

JANUARY 26, 1989



ADVANCED DEBUGGING TOOLS GO FOR SPEED

- THE FIRST IN-CIRCUIT VERIFIER FOR LOGIC-CELL GATE ARRAYS
- 50-MHz DEVELOPMENT SYSTEM TAKES ON 88000 RISC PROCESSORS
- FASTEST EMULATOR FOR 32-BIT CPUs SUPPORTS 68030 DEBUGGING

DEVELOPMENT TOOL TROUBLE-SHOOTS PGAS IN THE TARGET SYSTEM

JOHN NOVELLINO

In a move that might be dubbed "Back to the Future," Data I/O Corp. has reinvented the breadboard. The 1990s version of the circuit designer's old friend does away with the pins and sockets its predecessor needed to prototype circuits with discrete components. Instead, the MESA I in-circuit verifier targets the task of prototyping, verifying, and debugging a special type of ASIC called the programmable gate array (PGA).

The MESA I solves a problem inherent to highly integrated ASICs: the unobservability of internal circuitry. With this unique tool, designers can view the contents of every node in the circuit. As a result, verification time drops by 50%. Equally as important, it shortens the design cycle by reducing the need for rigorous simulation.

The Redmond, Wash.-based company designed the MESA I (for modeling, emulation, simulation, and analysis) specifically for the Logic Cell Array. Introduced by Xilinx in 1985, the LCA was the first PGA on the market and is still the most prevalent in use. The array differs from other programmable logic devices (PLDs) in that its configuration is held in an internal RAM rather than in a fuse pattern. Consequently, it's easily and quickly reprogrammable.

Designers can perform four basic tasks with the in-circuit verifier:

- Check that the configuration was properly downloaded to the PGA.
- Observe the register states within the PGA.
- Isolate specific signals from the target system and force states to isolate problems.
- Try out design alternatives with the array in the target system.

As a relatively new technology, PGAs still have limited capability. The maximum gate number is 9000, compared with 100,000 for other types of gate arrays. And at 70 ns, PGA speeds are slower than either PLDs (10 to 25 ns) or ECL gate arrays (2 to 3 ns).

REDUCE DESIGN COST

But in suitable applications, the PGA's reprogrammability offers a tremendous advantage over gate arrays. It takes four to six weeks to get gate-array prototypes, at $20,000 to $50,000. And commonly quoted statistics show that although 85 to 90% of ASICs returned from the foundry perform as designed, fully 50% don't work as planned in the target system. PGAs eliminate the high cost and delay of design changes. Using PGAs, engineers can quickly and inexpensively go through several design iterations.

Combined with the advantages of the PGA, the in-circuit verifier offers a design process in which errors can be corrected quickly and inexpensively, according to F. David Kohlmeier, a section engineering manager in Data I/O's Design Automation Division "As long as you can reiterate quickly and see exactly what was wrong, there's no pain in this process," he says. Consequently, the need to do heavy-duty simulation probably disappears. "Some people will still want to do it," Kohlmeier adds. "We're not knocking simulation. We're just saying that because of the LCA and its environment, the need for simulation is much less, provided that you have a tool that lets you debug your part in the circuit."

Mesa I consists of a chassis or pod containing the pin-logic circuitry, a cable assembly and probe for the target system, software that runs under Microsoft Windows (which is bundled with the product), and a controller board that plugs into a PC AT and supports up to four pods *(Fig. 1)*. Because the unit employs the same computer needed by Xilinx's LCA design tools, designers need not buy additional equipment. The verifier's

1. THE MESA I IN-CIRCUIT VERIFIER facilitates the prototyping, verification, and debugging of programmable gate-array designs. A probe connected to the chassis by a flex-circuit cable plugs into the array's socket in the target system and, in turn, holds the chip being debugged.

low power consumption makes it possible for it to run off the computer's power supply.

The Mesa software is also compatible with the Xilinx design software, so users can extract symbolic information from Xilinx design files. As a result, designers can refer to logic blocks by their symbolic names (for example, "counter" or "decoder") rather than by generic designations. Drop-down menus, dialogue boxes, and other prompts guide users through the system's setup and operation *(Fig. 2)*.

Shadow LCA Used

The probe, which is connected to the pod by a flex-circuit cable, plugs into the LCA socket on the target system. A socket on the probe holds the target LCA. The chassis has another PLCC socket into which the user plugs a shadow LCA that can mirror the states of the target LCA.

Square pins surrounding the shadow LCA socket connect to that LCA's external pins. Consequently, designers can examine the contents of the shadow's nodes and thus the target's nodes. If a signal goes to an external pin, it can be viewed directly. If not, it can be routed out to one of the pins.

Data I/O's own designers learned the ins and outs of working with PGAs by experience. They incorporated 13 LCAs into Mesa I: one on the controller board and 12 in the chassis. Of the latter group, 10 control the signals from the target system; one receives configuration data for downloading; and one serves as a manager for interrupts, communication to the PC, and other miscellaneous duties. At power-up, the Mesa I software automatically configures the unit's LCAs, so Data I/O can update the system simply by sending out new diskettes.

The designer can operate MESA I in several modes that can be broadly classified as configuration, in-circuit verification, and out-of-circuit verification. In the configuration mode, users download a configuration file to the shadow or target LCA, then read it back to verify proper loading. MESA's software always configures the shadow LCA, while the target can be configured by either the software or the target system.

Initially, the data goes to the pod's RAM, from which it can go directly to the designated LCA or wait until a request from the target triggers downloading. Holding the configuration in RAM eliminates delays in responding to download requests. For a system with multiple LCAs, the design can download one file with data for every array, and each will take its proper data and pass the rest down the chain.

With in-circuit verification, engineers can observe the LCA's internal states and isolate faults. The first technique available is random-sampled readback, in which MESA I randomly samples and stores the contents of either the shadow or target array's internal registers. In this mode, the designer checks to see whether the registers can change states. Because it's asynchronous, though, it can't offer a simultaneous look at all registers.

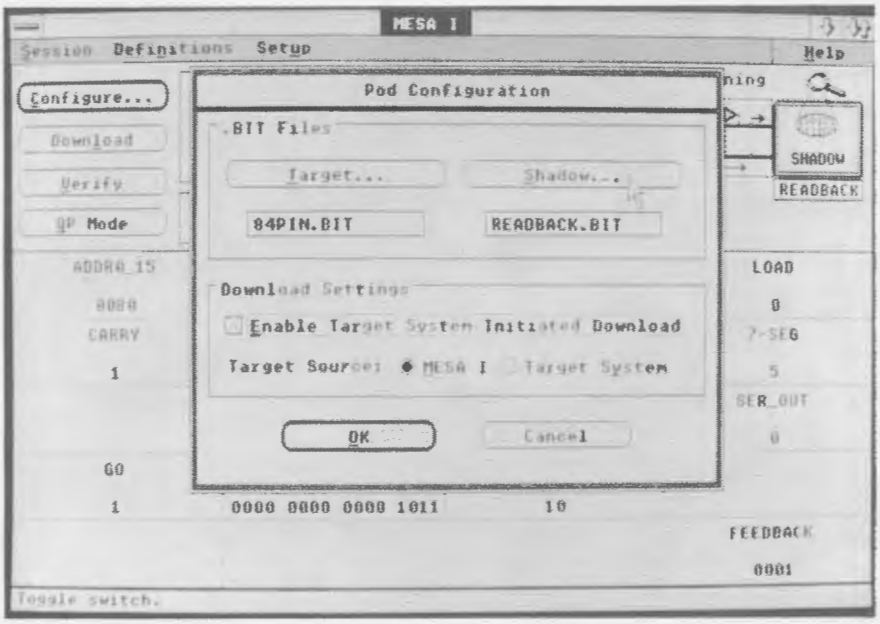
Synchronous readback, on the other hand, takes a chip-wide "snapshot" of the shadow LCA, so users can look at the relationship between the contents of all registers. A trigger signal, either from the MESA software or an external instrument, stops the transmission of signals to the shadow LCA, freezing it for viewing. If a logic analyzer sends the trigger, it can be synchronized to a specific event.

A Register History

To get a history of selected registers, including the timing relationships between states, engineers use real-time dynamic viewing. First, the selected internal signals must be routed out to the shadow LCA's external pins. If the design is too large to allow room for this routing, the Xilinx chip editor can help remove some of the circuitry.

In this mode, the shadow LCA becomes important. Because it has no effect on the target, the shadow's circuitry can be changed, and the target can continue running in the target system, supplying stimulus to the shadow LCA. Engineers can quickly restore the full design after the problem is solved.

To isolate problems, users disconnect suspect signals from the target system, so that they no longer receive stimulus. Then these signals are forced to the desired state with the verifier. One way is to disconnect selected pins on the shadow LCA and



2. THE SOFTWARE FOR THE MESA I runs under Microsoft Windows, which is bundled with the product. Drop-down menus, dialogue boxes, and other prompts make the user interface easy to use.

PRICE AND AVAILABILITY

The MESA I in-circuit verifier costs $9390. Delivery is eight weeks after receipt of order.

Data I/O Corp., 10525 Willows Rd. NE, P.O. Box 97046, Redmond, WA 98073-9746; (206) 881-6444.

toggle them to the desired state. If the system operates properly, the problem is deemed to be in the forced signals. Forced inputs also can simulate a function the target isn't yet able to supply, or verify that another component in the target system is working.

Furthermore, users can force vectors by disconnecting all pins from the target stimulus and controlling the LCA with the MESA software. Functional verification vectors, or sequences of signals, are then applied to force desired sequences of states onto the pins. These vectors typically need not be the large set needed for gate-array simulation, but rather 10 or 15 vectors aimed at a specific function.

Finally, engineers can isolate faults using a partial set of force vectors, along with a real-time stimulus from external instruments, such as function generators or a stimulus from the target system. This combination lets designers troubleshoot faults that show up only with a real-time stimulus.

Out-of-circuit verification is where MESA I really becomes a breadboard. Engineers can verify an LCA design before the system board is ready or can experiment with different design alternatives. Only the pod socket is used. After downloading a trial configuration file from the development system into the shadow LCA, engineers can force inputs as desired and observe the internal states and outputs that result. They can use either full vectors or partial vectors with external stimuli.□