Reed Electronics Group

MICROPROCESSOR REPORT

www.MPRonline.com

THE INSIDER'S GUIDE TO MICROPROCESSOR HARDWARE

Tabula's Time Machine

Rapidly Reconfigurable Chips Will Challenge Conventional FPGAs

By Tom R. Halfhill {3/29/2010-01}

For years, chip designers have yearned for the day when three-dimensional chips with stacked layers of logic will be practical. Stacked chips could dramatically reduce the number of parts in a system while improving performance. Unfortunately, the obstacles of manufacturing costs and design complexity are too high today, except for very limited applications.

Now, a Silicon Valley startup is taking a radically different approach. On March 15, Santa Clara-based Tabula announced new programmable-logic devices that emulate three-dimensional stacked chips by rapidly reconfiguring their two-dimensional fabrics. With these devices, the third spatial dimension exists for only a split-second slice of time.

By storing multiple gate configurations on chip, Tabula's devices can completely reconfigure their fabrics up to 1.6 billion times per second. That's about one million times faster than conventional FPGAs that load their gate configurations from off-chip memory. Rapid reconfiguration makes the physical fabric seem much larger than it really is. Tabula's first-generation chips can reuse the same physical gates for as many as eight different functions. In this way, a Tabula chip can match the capacity of an FPGA that's larger and more expensive.

The basic concept of rapidly reconfiguring logic isn't new. Other FPGA companies have experimented with similar technology for 20 years. However, Tabula is the first to make a bet-the-company gamble on bringing the technology to market. In addition, Tabula claims to have solved the problem of efficiently placing and routing a circuit design in three dimensions while hiding the details from developers.

As Figure 1 shows, a Tabula device appears to be an FPGA/PLD with up to eight stacked layers of physical logic, memory, and interconnects—a real three-dimensional chip.

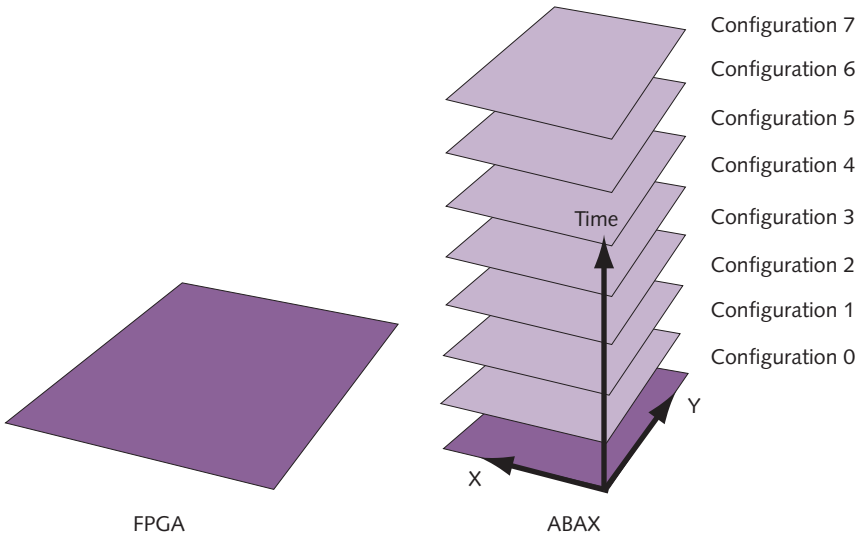

Figure 1. Like a conventional FPGA/PLD, a Tabula 3PLD has only one physical fabric. By rapidly reconfiguring the fabric, each physical gate can perform up to eight different functions. Tabula's development tools strive to hide these details from developers, making the device appear to be a three-dimensional chip with eight layers of logic stacked in a single package.

But there's really only one physical layer. The 3D abstraction extends from gate-level programming all the way down to final place-and-route. Rapid reconfiguration simply gives developers more "gates" to play with—gates that exist for only a moment of time.

In fact, the chips could reconfigure themselves even faster than 1.6 billion times per second, but Tabula is leaving enough time between transitions to emulate a two-dimensional FPGA running at 200MHz. If an application demands higher performance, the chip can reconfigure itself fewer times per clock cycle, trading capacity for speed.

Without reconfiguration, the fabric would run at the chip's actual maximum clock frequency of 1.6GHz. Some hard-wired function blocks on the device actually do run at that speed. Although the high clock frequency raises questions about excessive power consumption, Tabula says its smaller devices use about the same dynamic power as conventional FPGAs, while reducing static current leakage.

Tools Hide the Details

In addition to increasing the perceived size of the programmable fabric, Tabula says its devices offer performance benefits. They have shorter signal paths, less interconnect overhead, greater memory density, place local memory closer to logic, and allow hardware multipliers to run at full chip speed—up to 1.6GHz in the first implementations.

Perhaps the real breakthrough is the abstract programming model. Tabula says its proprietary development tools completely hide the time slicing from developers. The temporal dimension is modeled as a third spatial dimension. From a developer's view, the fabric isn't being reconfigured at all. The development flow is like using a conventional FPGA.

Tabula calls its patented technology "Spacetime"—homage to Albert Einstein's "space-time continuum" and Hermann Minkowski's related concept of "spacetime." Einstein proposed his theory of special relativity in 1905. Three years later, Minkowski derived from special relativity a new geometry that adds a time dimension to the conventional spatial dimensions of classical Euclidian geometry.

For Tabula, this history isn't merely academic. Einstein's and Minkowski's spacetime theorems are crucial to the company's technology. Tabula says its place-and-route tools rely, in part, on 100-year-old equations to solve the problem of efficiently laying out a circuit design in three dimensions—two spatial dimensions (Euclidian) and one temporal dimension (per Minkowski).

The same equations allow Tabula's tools to model signal propagation through the virtual 3D fabric. This modeling is essential to reconfiguring the logic gates and to optimizing a circuit design's critical paths. In modern engineering terms, Spacetime technology uses time-division multiplexing to virtualize a third spatial dimension in a two-dimensional physical fabric of reprogrammable logic.

Faster Than Partial Reconfiguration

Don't confuse Tabula's Spacetime technology with another trick called partial reconfiguration. Some conventional FPGAs can reprogram parts of their fabrics with different blocks of a user's design, on the fly. But partial reconfiguration is more like swapping overlays in a memory-starved software program. Rarely, if ever, is the FPGA's entire fabric reconfigured. Developers must manage the swaps under software control and each swap is six orders of magnitude slower than is possible with Tabula's first-generation Spacetime technology.

To highlight these differences, Tabula refers to its chips not as FPGAs or PLDs, but as 3PLDs—a generic (not trademarked) term for a three-dimensional programmable-logic device. (Tabula derives its company name from the Latin term *tabula rasa*: "blank slate.") Virtually, they *are* 3D PLDs.

Tabula is eager to distinguish 3PLDs from two-dimensional PLDs because the company is challenging all the existing players in the market, including the industry stalwarts, Altera and Xilinx. It's a bold ambition for a startup with only 105 employees, but Tabula has the potential to disrupt the status quo.

Over the years, numerous FPGA startups have failed because they tried to make better conventional devices than Altera and Xilinx offer. That's why Tabula is taking a different approach. Another FPGA startup trying something different is Tier Logic, which emerged from stealth mode only days before Tabula's product announcement on March 15. (See the sidebar, "Another Three-Dimensional FPGA Debuts.")

Ordinary but Extraordinary

First, some background on Tabula. Founder Steve Teig, CTO, was formerly CTO of Cadence Design Systems and four previous startups, including Tangent Systems. He pioneered place-and-route tools for sea-of-gates ASICs and has been working on Spacetime technology for six years. Tabula's CEO is Dennis Segers, formerly CEO of Matrix Semiconductor, which pioneered 3D memory chips. As a senior vice president at Xilinx, Segers launched the Virtex line of high-end FPGAs.

Tabula is salted with engineers from companies like Altera, Xilinx, Cadence, and Matrix Memory, so they have experience with ICs, programmable logic, and FPGA development tools. So far, Tabula has applied for about 150 patents; about 80 have been issued. The company has raised $106 million from eight venture-capital firms. (Full disclosure: Dave Epstein, a longtime member of the *Microprocessor Report* editorial board, works for one of those firms, Crosslink Capital. Epstein did not participate in the preparation of this article.)

The basic semiconductor technology underlying Spacetime technology is surprisingly ordinary. Tabula's foundry is TSMC, which manufactures the chips in its normal 40nm

bulk-CMOS process—the same process Altera uses for Stratix-IV FPGAs. TSMC began making test chips for Altera in early 2006 and for Tabula in September 2007. Stratix-IV chips began full production in early 2009. Tabula's first production runs are scheduled for the fourth quarter of this year.

In other words, there's nothing special about the fabrication or system-level operation of Tabula's chips. They are manufactured, programmed, and designed into systems like conventional FPGAs. Tabula and a lead customer have been working with silicon for a few months. The product line, formally introduced on March 15, is branded Abax, derived from "abacus."

Fundamentally, Abax 3PLDs are SRAM-based FPGAs. That is, the programmable-logic gates and interconnects in the fabric are configured by standard six-transistor SRAM cells that store the gate-level layout of the circuit design. Unlike flash-based FPGAs from companies like Actel, all SRAM-based FPGAs are volatile—the fabric forgets the configuration when powered down.

Tabula's Abax 3PLDs are no exception. Like conventional SRAM-based FPGAs, they must load their initial configuration from external memory, such as flash, ROM, or DRAM. Initialization requires approximately the same amount of time as with a conventional FPGA. However, Abax 3PLDs can reconfigure themselves much more quickly because they load all eight configurations into SRAM at initialization. When a conventional FPGA does partial reconfiguration, it must load the new configuration from slower off-chip memory.

At the system level, an Abax 3PLD works like a conventional FPGA. Developers can program the fabric in Verilog, VHDL, or the various C-like languages for FPGAs. There are no special design requirements, except that developers must use Tabula's back-end tools—the same kind of tools that an FPGA from another vendor would require. Abax devices are suitable for any system that would normally use FPGAs. Because they cost less, they are also suitable for some systems that would use ASICs.

Initially, Tabula is focusing on the communications market. That's where the company hopes to find the best opportunity for growth. FPGAs are commonly found in the infrastructure equipment of communications networks, such as cellular base stations. The relatively low volumes of this equipment may not amortize the development of an ASIC, and evolving communications standards favor a field-programmable solution.

Folding Time Into Space

If three-dimensional stacked chips were practical today, Tabula might be in that business instead of using Spacetime technology to emulate the third dimension. Although stacked ICs exist, usually they are ASICs that overlay a logic chip with a memory chip in the same package. Stacked ASICs conserve space in small embedded systems, such as cellphones. Stacked FPGAs can be built, but they would be too costly to compete with conventional FPGAs in all but the most expensive systems.

Tabula doesn't claim to have invented the idea of rapidly reconfiguring a PLD to emulate stacking. Xilinx has experimented with similar technology since 1991. In 1997, Xilinx published an IEEE paper titled "A Time-Multiplexed FPGA." Indeed, the figures in the Xilinx paper look much like the illustrations Tabula uses to explain Spacetime technology. Other researchers published similar papers in the 1990s. Two recent textbooks also describe the concept; coincidentally, both books are titled *Reconfigurable Computing*. (For references to these sources, see the "For More Information" box.)

Conceptually, rapid reconfiguration works like server virtualization, which can make one physical server appear to be, say, eight virtual servers. Another rough analogy is time sharing on a mainframe. To users, a single computer with eight terminals seems like eight computers because the mainframe is fast enough to switch among eight programs running simultaneously. Figure 2 is another illustration of the concept.

An Abax 3PLD can rapidly switch among eight different configurations of the programmable-logic fabric, repeating these transitions in an endless round-robin loop. Tabula calls each configuration a "fold," because it folds time into space. The number of folds can vary to suit the capacity and performance requirements of the target application. First-generation Abax chips support a maximum of eight folds; future chips will support more. (Additional details later.) If

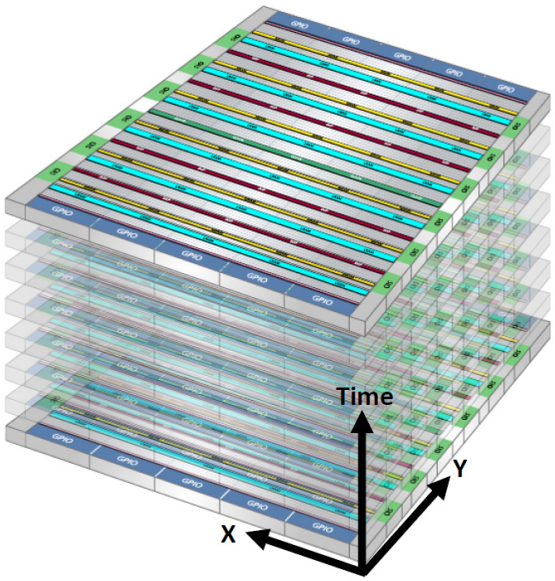

Figure 2. Tabula uses time to emulate the third spatial dimension, making one fabric seem like eight fabrics stacked together. Each configuration is called a "fold" because it folds time into space. Tabula has designed its development tools to present the folds as physical layers of logic. The goal is make reconfiguration transparent to developers.

a stacked FPGA were practical, Tabula says its development tools could be adapted to program it.

Passing State Through Time

In a true stacked chip, thousands of vertical interconnects, called vias, would connect each die to the ones above and below it, passing signals from layer to layer. Tabula does the same thing with "time vias." A time via passes the state of each wire to the next fold in the round robin by using a transparent latch (a standard logic element). Almost every wire in an Abax 3PLD has a transparent latch, so almost every wire has a time via to the next virtual layer.

When open, a transparent latch works like a FIFO buffer—signals pass through. When closed, it holds the signal. In an Abax 3PLD, an open latch propagates signals within a fold. A closed latch becomes a state element, passing the signal to the next fold. Some time vias may pass signals through multiple folds. In a stacked chip, physical vias would pass these signals vertically through the stack. In an Abax chip, time vias pass the signals forward in time through the virtual stack. In this way, each fold passes the active states of all its circuits to the next fold. Figure 3 illustrates the function of a time via.

One limitation of time vias is that time flows in only one direction—forward. Interconnect vias can pass signals up and down through the stack of chips, but time vias can pass them only "upward" through the stack of folds. A fold can't pass a signal directly to the previous fold, because Tabula hasn't found a way to make time run backwards.

However, Spacetime technology offers a compensation: *time wraps around.* The last fold in the round robin passes signals directly to the first fold. Moreover, this transfer happens just as quickly as any fold-to-fold transfer does. In a physically stacked chip, sending a signal from one end of the stack to the other would take more time, because the signal must pass through all the layers.

Wrap-around time is a mind-bending concept. In Tabula's universe, time isn't linear—it's an endless loop. As Figure 4 shows, the virtual stacked chip isn't really shaped like a pile of chips—it's a torus. Time vias stitch the slices of the torus together. Signals propagate through the time vias, flowing around and around, from one fold to another. Meanwhile, other signals are flowing in the conventional manner through the two physical dimensions of each fold in the fabric.

Carrying this analysis to the next level, yet another shape emerges. As signals propagate in the "vertical" or "circular" direction through the folds and in the "horizontal" directions within a fold, they can reach more and more gates over time. Their potential paths resemble a cone—another geometrical aspect of Einstein's and Minkowski's work. The torus in Figure 4 would gradually grow wider, like the flared end of a tuba. But unlike a tuba, the cone would wrap around itself, as a torus does. We can't draw this shape in classical geometry. However, as we'll explain later, the cones have additional implications for Tabula's devices.

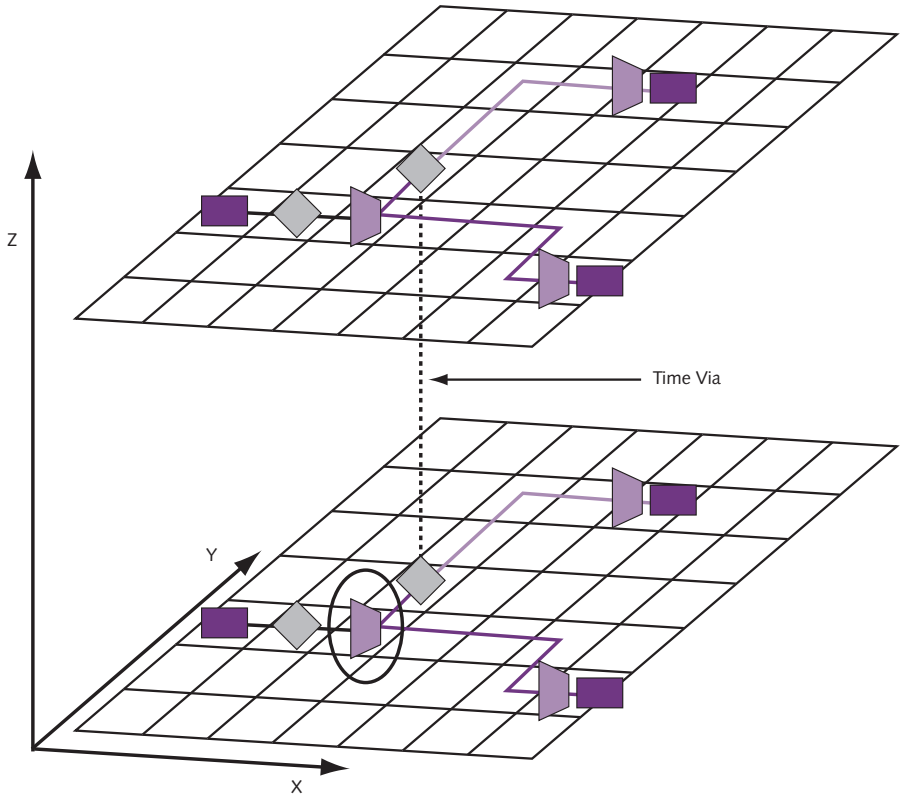

Figure 3. Tabula uses transparent latches as "time vias" to pass signals forward in time from one fold (fabric configuration) to another. Time vias emulate the vertical interconnect vias in a stacked chip. When the latch is open, signals pass through and remain in the same fold. When closed, the latch becomes a state element, holding the signal until the next fold is ready.

Proprietary Tools Handle Layout

Tabula says the latency of a time via is approximately the same as the latency of an interconnect via in a stacked chip. In first-generation Abax devices, the latency is about 80 picoseconds. So, for the purposes of layout and timing closure, it doesn't matter if signals are traveling in the two spatial dimensions of the fabric (x and y) or in the temporal dimension (virtual z). Signal propagation is the same, except for the limitation described above—time vias can propagate a signal in one direction only.

Tabula's place-and-route tools model the signal-propagation characteristics of the fabric and try to calculate the optimal placement of logic gates and interconnects, just as all place-and-route tools do. When laying out the gates and wires of a critical path, the tools try to

pack those gates and wires close together in the same fold, or in adjacent folds, to achieve the shortest possible path with the least signal delay. Gates and wires in less-critical paths will be further separated, but within the limits of their timing requirements.

According to Tabula, the abstraction is so complete that the place-and-route tools think they are laying out a real three-dimensional chip. If stacked FPGAs ever become practical, Tabula says its tools could do the physical layout.

At every step, the tools strive to hide these details from developers. To someone programming an Abax 3PLD, time vias are as invisible as wire vias. The programmable fabric simply appears to be larger than its physical size. Just as users of time-share terminals aren't aware that their programs are running for only a slice of time on the mainframe, users of an Abax 3PLD aren't aware that the programmable gates in their design are running for only a slice of time in the fabric. Developers don't know if any particular gates in their design are placed in the same fold or in different folds. If Tabula's tools can't achieve timing closure, they generate the same error messages that other place-and-route tools do, and the solutions are the same.

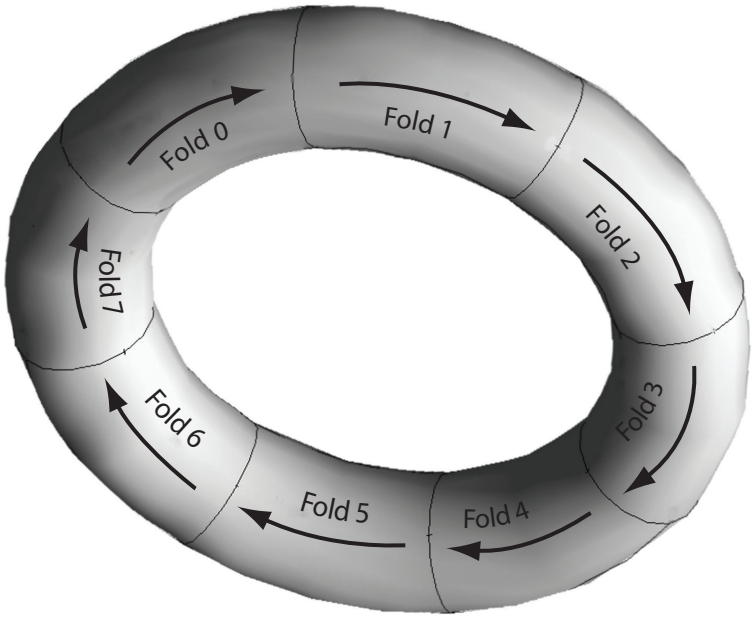

Figure 4. The virtual stack of fabrics in Figures 1 and 2 illustrate the concept of emulating a three-dimensional chip in time, but the technology has more profound implications. Time isn't linear. It's an endless loop, because the last fold wraps around to the first fold. The virtual stack is really a torus.

There is one circumstance in which developers become somewhat aware of folds. Earlier, we mentioned that the number of folds can vary to suit the capacity and performance requirements of the target application. First-generation Abax chips support a maximum of eight folds. If an application demands higher performance, developers can trade fabric capacity for clock speed by reducing the number of folds. Even in this circumstance, however, developers don't work directly with folds.

Fewer Folds, Faster Logic

The maximum number of folds is more or less arbitrary and depends on the maximum clock frequency of the chip. Tabula is getting acceptable yields from TSMC's 40nm process at 1.6GHz, so that clock speed was established as the baseline for the initial devices. Future devices could run at higher or lower clock rates.

With eight folds, the chip appears to run at one-eighth the base clock frequency, or 200MHz. Tabula refers to this frequency as the "user clock speed." It accounts for the fact that each logic gate is performing a particular function for only one-eighth the base clock rate of the chip. Within each fold, however, logic gates and signals are actually running at the base clock speed. Hard-wired blocks, such as the DSP multipliers found on Tabula's highest-end device, also run at the base clock speed.

Some competing FPGAs from Altera and Xilinx run as fast as 600MHz. However, functions implemented in their fabrics generally run at 200MHz to 350MHz. Xilinx says next-generation wireless designs need to run their hard-wired DSP blocks and much of the fabric at 368MHz to perform their required duties.

If a target application needs more than 200MHz of performance, an Abax 3PLD can use fewer folds, trading capacity for speed. With four folds instead of eight, each configuration of the fabric appears to run at one-fourth the base clock speed—400MHz for a 1.6GHz Abax chip—but the fabric capacity is halved. With two folds, the user clock speed soars to 800MHz, but the fabric capacity is halved again. (To scale performance more finely, odd numbers of folds are possible.) Figure 5 illustrates the relationship between folds and user clock speeds.

Using only one fold would deliver the chip's full clock speed of 1.6GHz. In practice, that extreme wouldn't make sense, because it throws away the advantages of Spacetime technology. At 1.6GHz, the 3PLD fabric would appear to be no larger than the fabric of a two-dimensional PLD.

As fabrication technology continues to advance and clock speeds accelerate, Tabula's chips can use more folds or faster folds. Tabula leans toward more folds. If a chip could run at 8.0GHz, Tabula's models suggest that 40 folds running at a user clock speed of 200MHz would use the fabric more efficiently than, say, 10 folds running at 800MHz. The requirements of the target application will determine the best trade-off between fabric capacity and performance.

Getting More From Moore's Law

Another implication of Tabula's modeling: as time goes by, Abax 3PLDs should reap greater benefits from Moore's

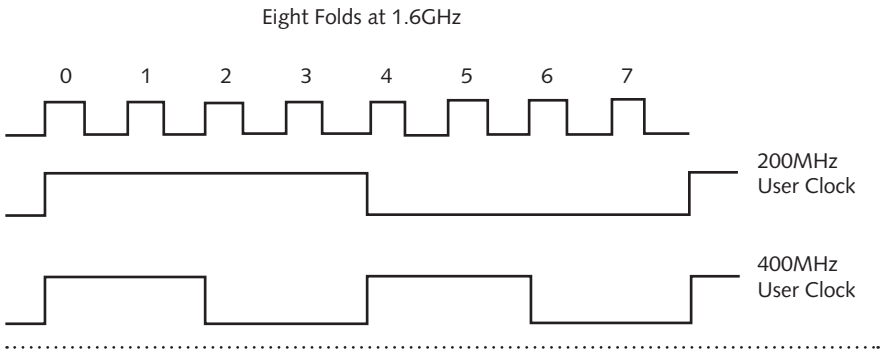

Figure 5. Tabula's first-generation devices run at 1.6GHz, but the perceived user clock speed depends on the number of folds. With eight folds—the maximum in these first devices—the "user clock rate" is 200MHz. That's the effective speed of the programmable logic, although signals within each fold run at the base clock rate. If an application needs more performance, developers can trade capacity for speed. With four folds, the user clock rate is 400MHz, but only half as many gates are available.

law than other chips do. Moore's law predicts that transistor densities will double every 24 months. But higher clock speeds allow Tabula to add more folds, so the capacity of its chips can grow faster than the rate predicted by Moore's law. In theory, Tabula should be able to widen its advantage over competitors with every process shrink. This advantage could prove strategic when the Moore's-law curve flattens out, as someday it must.

The entire fabric of an Abax 3PLD needn't run at the same user clock speed. Developers can divide the fabric into multiple clock domains and run them at different clock speeds, up to the maximum frequency of 1.6GHz in first-generation devices. Faster clock domains have fewer folds. Clock granularity is extremely fine—a domain can have as few as two gates.

A reasonable question is whether Tabula's chips can find enough time to do their work if the fabric is reconfiguring itself 1.6 billion times per second. The duration of each fold depends on the user clock speed and number of folds:

duration = 1,000,000 / (freq x folds) ps

So, a 3PLD with a user clock speed of 200MHz and eight folds will spend 625 picoseconds in each fold. To put this number in perspective, light travels about 5/8 of an inch in 625 picoseconds. Electrons traveling through a wire are slower than photons in a vacuum.

Those numbers would seem to leave almost no time for signals to propagate through the gates and wires of the chip between configurations. And what about the time required for each reconfiguration?

Hiding Reconfiguration Latency

Signals that don't reach their destination within a fold are forwarded through a time via to the next fold. Some signals will pass through two or more folds before reaching their destination. As a rule, Tabula's place-and-route tools try to minimize the forwarding, but this is one aspect of the technology that will depend on the efficiency of Tabula's tools.

Although each reconfiguration takes about 80 picoseconds, Tabula says that latency is almost completely hidden by the transparent latches that implement the time vias. In effect, reconfiguration happens in parallel with signal propagation.

Assume a signal is moving from a programmable-gate lookup table (LUT) in fold 0 to another LUT in fold 1. The wire connecting the two LUTs has a transparent latch that closes when the signal moves through it. The closed latch tells fold 1 that a signal from fold 0 is moving through that wire. The signal continues traveling through the wire as the fabric reconfigures itself for fold 1 and conjures the destination LUT into existence.

If the destination LUT materializes before the signal arrives, the wire latency completely hides the reconfiguration latency. No time is lost at all. If the signal arrives *before* the LUT appears, some time is lost, because the latch must regenerate the signal. The important point is that developers needn't worry about these latencies. Tabula says its place-and-route tools automatically account for the latencies of gate reconfiguration and wire delays.

For instance, assume that the signal that would arrive before the LUT appears in fold 1 happens to be in a time-critical path. The place-and-route tools try to move the LUTs closer together, perhaps by shortening their interconnect or by relocating them to the same fold. If the signal isn't time-critical, its arrival time at the destination LUT doesn't matter, for the purposes of timing closure.

Place-and-route tools for conventional FPGAs work on the same principles, but they arrange LUTs and wires for the optimal path in a two-dimensional fabric. Tabula's tools arrange LUTs and wires in a three-dimensional fabric that emulates a stacked chip with up to eight layers. Spacetime technology gives the place-and-route tools more options for placement. If the latency of a time via is approximately the same as the latency of a wire via, an Abax 3PLD really functions as a three-dimensional fabric would.

Turning Wire Latency to Advantage

In conventional FPGAs, wire latency is always bad. LUTs that could be switching signals are idly waiting for their signals to arrive. Wire latency is usually bad in a 3PLD, too, but there's an exception.

Because Tabula's place-and-route tools are aware of the latencies for wires, gates, and folds, they can take advantage of those latencies. The 3PLD can reuse a LUT that otherwise would be idly waiting for a signal to arrive. The LUT can perform a completely different function while the first signal is traveling through a wire. By the time the signal arrives, the LUT will be reprogrammed to perform the function needed by that signal.

Analogy: In football, a tight end is a pass receiver who specializes in blocking, too. He may block a linebacker for a moment, then disengage and catch a pass. The quarterback may throw the ball while the tight end hits the linebacker. By the time the ball arrives at the predetermined destination, the tight end has left the linebacker behind and arrived at the target location on his pass route. So, in one play—perhaps while the ball is in the air—the tight end has performed two different functions: blocking and receiving. In the same way, a LUT in a 3PLD can perform two different functions in the time required for a signal to traverse a wire.

Tabula says this stunt is invisible to developers. The place-and-route tools simply recognize an opportunity to reuse an idle LUT for a different function, depending on the latency of signal propagation. This technique increases the number of LUTs available for the developer's design, even beyond the reconfiguration of LUTs from one fold to another.

Although place-and-route tools for conventional FPGAs have the same foreknowledge of signal propagation through their fabrics, they can't reconfigure the LUTs quickly enough to reuse them for different purposes. Whereas an Abax 3PLD reconfigures its LUTs using on-chip SRAM, a conventional FPGA must load a new gate configuration from off-chip memory. That's much too slow to keep up with the rapidly moving signals.

Rewriting Rent's Rule

Another advantage of its 3PLDs, says Tabula, is a simpler interconnect network. Although it's common to think of FPGAs as vast seas of gates, they are more like vast networks of interconnects. Indeed, the wires connecting the LUTs typically occupy most of the silicon. To implement complex designs, the fabric needs enough wiring for the place-and-route tools to stitch the gates together in an almost infinite number of configurations.

Most muxes in a conventional FPGA interconnect fabric aren't used when the final design is routed. Whereas gate utilization often reaches 75% to 80%, mux utilization can be less than 5%. That's a lot of wasted wiring.

Tabula says 3PLDs have less interconnect wiring in the two spatial dimensions of the chip. Less wiring is needed because the time vias provide toroidal paths through the folds, allowing the place-and-route tools to pack the logic closer together in Spacetime. Remember, each fold has another virtual fabric immediately "above" and "below" it, and the structure wraps around itself. Tabula says the gate utilization of a 3PLD is about the same as conventional FPGAs (75% to 80%), and mux utilization is better. (Tabula doesn't specify how much better; it depends on the design.)

If Tabula's interconnects are truly more efficient, they could rewrite Rent's rule. This rule—first described by IBM's E.F. Rent in 1960—is really an observation that the complexity of the interconnects in a computer expands faster than the complexity of the logic. Interconnect complexity is a growing problem in advanced SoC and FPGA designs. The wiring is occupying more and more room on the chip, wasting silicon that could be used for logic. And, as the interconnects lengthen, timing closure becomes more difficult, owing to signal delays.

In any 3PLD—whether the third dimension is spatial or temporal—gates are closer together and wires are shorter. Tabula says the interconnects in an Abax 3PLD are, on average, 78.5% shorter than the interconnects in a two-dimensional chip. Thanks to these shorter wires, a 3PLD can reach about 3.2 times more LUTs per clock cycle than a two-dimensional PLD with the same number of LUTs.

In theory, 3PLDs should derive more benefit from Moore's law than conventional FPGAs do. As we mentioned before, Tabula can use higher clock speeds to add more folds, increasing the density of the fabric. But it's not just transistor density that matters; interconnect density is important as well. Other FPGAs can expand in only two dimensions, so their interconnects (and wire delays) will keep lengthening at a faster rate than Tabula's interconnects do. Figure 6 illustrates this concept.

Tabula makes this rough analogy: the Willis Tower (formerly Sears Tower) in Chicago is like a 3PLD, because its floors are stacked vertically. In contrast, the Pentagon in Arlington, Virginia, is more like a two-dimensional FPGA. The worst-case "wire latency" between any two offices in a skyscraper is the time required to walk to the nearest

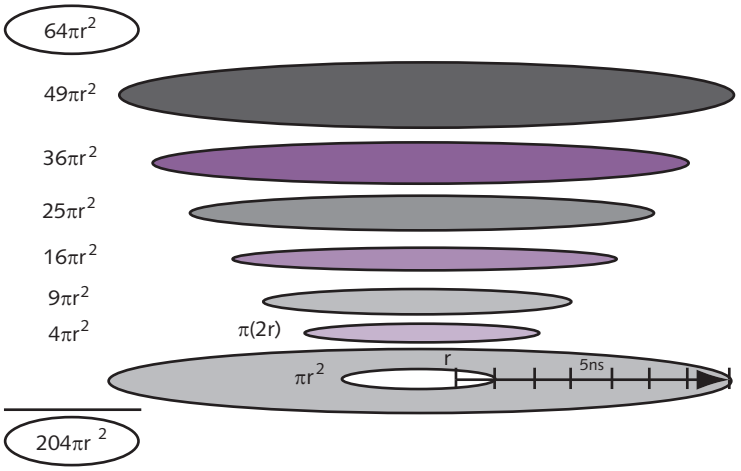

Figure 6. As chip-fabrication technology improves, Tabula's 3PLDs may derive greater benefits from Moore's law. Faster clock speeds allow Tabula to add more folds to its virtual 3D fabric. This has implications for interconnects, as well as for gate density. Conventional FPGA fabrics can expand in only two dimensions, lengthening their interconnects and signal paths across the chip. Tabula's fabrics can expand in three dimensions, allowing shorter "vertical" paths. The cone-shaped stack of circles in this figure shows that signals can reach more gates as they propagate forward through the time vias and virtual stacked layers of logic. The expanding cone is characteristic of Hermann Minkowski's spacetime geometry, derived from Einstein's special theory of relativity. Tabula says Minkowski's 100-year-old equations were the key to solving the problems of 3D circuit layout.

elevator, ride to the desired floor, and then walk a short distance to the destination. The Pentagon's sprawling floor plan requires a traveler to hike around the long rings of corridors or traverse the building's large central plaza. Enlarging the Pentagon would only worsen the disparity; adding floors to the skyscraper wouldn't significantly lengthen the worst-case path.

Memory is Faster, Too

Another theoretical benefit of a 3PLD (whether the third dimension is spatial or temporal) is that local memory can be placed closer to logic and accessed more quickly. Memory isn't limited to hugging the periphery of the logic on a two-dimensional plane, like suburbs surrounding a city. Instead, memory can be tightly coupled to logic through vias—be they wire vias, in a physical 3PLD, or time vias, in Tabula's virtual 3PLDs.

Conventional FPGAs have dual-ported SRAM for user memory. Tabula uses single-ported SRAM. Although this difference allows Tabula to pack more SRAM into the same space, single-ported memory is accessible only once per clock cycle. Under normal conditions, it's slower than dual-ported memory.

However, as Figure 7 shows, a different function block of logic in an Abax 3PLD can access the same memory during each fold. In effect, the single-ported memory works like eight-ported memory. Internally, each virtual port can even be mapped to a different address space. (Tabula's synthesis tools handle the mapping, so it's transparent to developers.) Result: single-ported 1.6GHz SRAM appears to be eight-ported 200MHz SRAM and has shorter paths to logic.

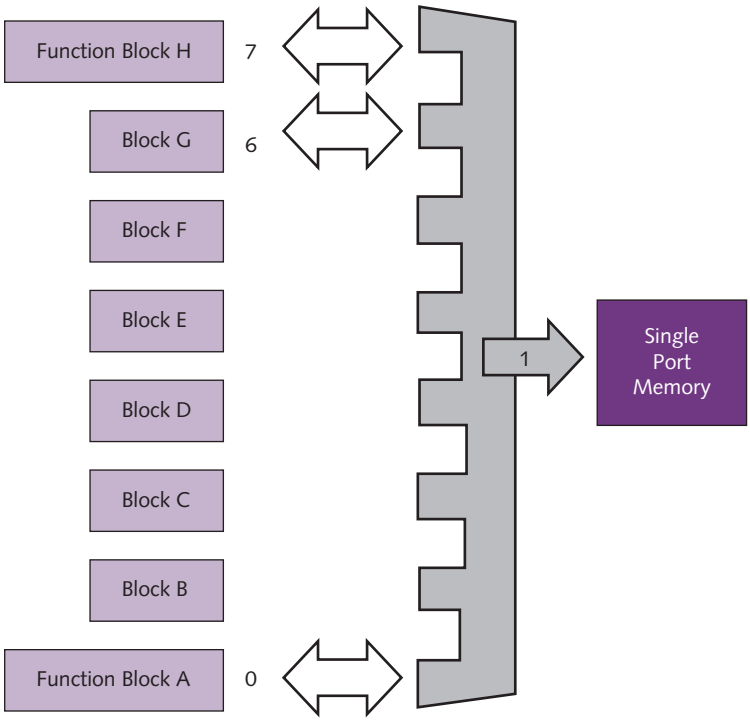

Figure 7. Memory access in a Tabula Abax 3PLD. Although Tabula's chips use single-ported SRAM instead of dual-ported SRAM for user memory, different function blocks can independently access the same memory during each fold. The effect is that each fold has a virtual port to memory. Therefore, in a 1.6GHz device, the memory appears to be eight-ported SRAM running at 200MHz. Single-ported SRAM cells are about half the size of dual-ported cells, so Tabula's memory is denser than the memory in conventional FPGAs.

Tabula's virtual porting will be especially valuable in high-performance designs. Assume that a design requires two or more function blocks to have dual-ported access to the same local memory. With conventional FPGAs, developers must cobble together their own multiported local-memory interface, using muxes and arbitration logic. With Tabula's eight-ported memory, as many as four different function blocks can have dual-ported access to the same SRAM.

The Overhead of Spacetime

The physical implementation of an Abax 3PLD is fairly conventional. The SRAMs are standard six-transistor cells compiled from a TSMC library. The programmable fabric is a tiled network of identical logic/memory blocks. Each tile has 16 physical LUTs, which become 128 virtual LUTs in an eight-fold design.

Tabula's devices use a LUT structure that's slightly more flexible than a basic four-input LUT. For comparison purposes, Tabula talks about "four-input LUT equivalents," because Altera, Xilinx, and other FPGA vendors describe their logic elements in different ways, making comparisons difficult. Some Xilinx Virtex FPGAs have six-input LUTs; to compare them with Tabula's, Xilinx multiplies the number of LUTs by 1.6. (Tabula's chips have several register files that can also be used as six-input LUTs.)

Each physical LUT in an Abax 3PLD has 512 bits of SRAM to store its eight possible gate configurations. That's 64 configuration bits per virtual LUT (or "user LUT"). In comparison, a Xilinx Virtex-6 FPGA has about 310 configuration bits per LUT. Tabula's LUTs need less configuration memory because their interconnects are simpler. A stacked chip with eight physical fabrics would need approximately the same amount of configuration memory as an Abax 3PLD.

As mentioned before, an Abax 3PLD stores all eight gate configurations on chip during initialization. Storing all the configurations in SRAM allows an Abax 3PLD to rapidly switch among its virtual fabrics without loading each configuration as a bitstream from slow off-chip memory. However, rapid reconfiguration does incur a power-consumption penalty, which Tabula calls the "reconfiguration tax." This tax partially offsets the power savings of having fewer physical gates, fewer transistors, and shorter interconnects on a smaller die. If future Abax devices add more folds, they will need more configuration memory.

As Figure 8 shows, the LUTs and configuration memory in each tile are surrounded by interconnects,

routing muxes, and the control logic required to manage Spacetime folding. Tabula has announced four Abax 3PLDs, with 220,000 to 630,000 LUTs. Each chip has 5.5MB of user memory and a base clock speed of 1.6GHz—fast enough for eight folds at a user clock rate of 200MHz.

Tabula hasn't expressed the overhead of Spacetime control logic as a percentage of the die. Judging from the layout in Figure 8, the control logic in each tile is as large as the configuration memory or mux routing. In other words, it's not insignificant. The overhead is a partial trade-off for the denser logic and memory enabled by Spacetime technology.

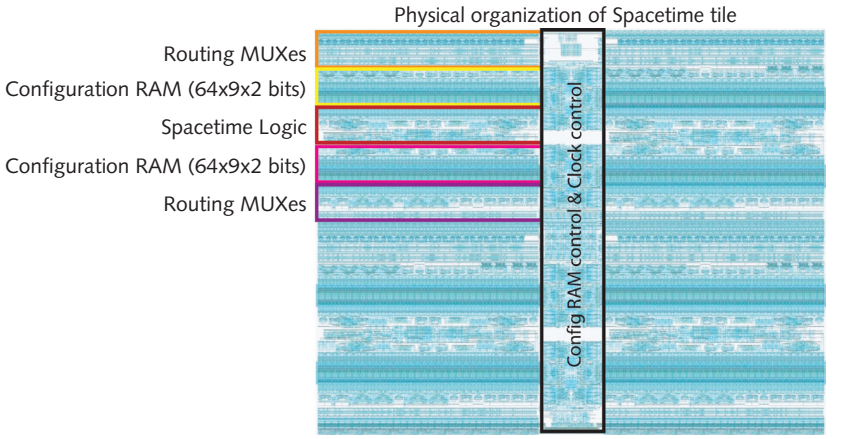

Figure 8. Physical layout of logic tiles in a Tabula 3PLD. The highlighted section shows one tile. It contains two blocks of multiplexers for routing the interconnects, two blocks of configuration memory (totaling 2,048 bits of parity-protected SRAM), and the Spacetime control logic required to switch configurations for every fold.

Keep in mind that a developer's design requires a certain number of logic gates to perform its intended functions. That gate count remains substantially the same, whether the programmable-logic fabric has one, two, three, or more layers, and whether additional layers are physical or virtual. Tabula's time-slicing technology reduces the number of physical gates in the fabric but doesn't change the gate count of the developer's design. Any space on the chip dedicated to managing the rapid reconfiguration of the fabric is overhead, because it occupies silicon that could have been used for more logic gates or memory.

Evaluating the Trade-Offs

The overhead of Spacetime technology reduces the fabric's density well below the theoretical maximum. In theory, an eight-fold Abax 3PLD is almost eight times denser than a two-dimensional PLD, because it can use each LUT for eight different functions per user clock cycle. And, in theory, there's more room for local memory, and the memory appears to have four times as many ports.

In practice, resource utilization isn't 100%. Not every LUT is used eight times. (In two-dimensional FPGAs, some LUTs aren't used at all.) The overhead of Spacetime control logic reduces the theoretical maximum even further. For a typical customer's design, Tabula estimates that an Abax 3PLD will offer about 2.8 times more logic (in LUTs), 3.2 times more user memory, and 3.0 times more memory ports than a two-dimensional device with a similar die size.

If those estimates are close to accurate, Tabula will have an exploitable advantage over vendors of conventional FPGAs. Tabula's chips will be smaller than competing devices with similar capacity, so they will cost less to manufacture. (On the other hand, Tabula's sales volumes will be lower, which increases the manufacturing cost.) For customers, an Abax 3PLD has about three times more capacity on a chip that's approximately the same size as a competing part. Or, trading off in the other direction, customers can get the same capacity on a chip that's much smaller than a competing device.

Unfortunately, it's nearly impossible to pin down the price/performance advantage of an Abax 3PLD. One problem is that FPGA vendors describe the amount of logic on their devices in different ways that defy easy comparisons. (It's probably deliberate.) Altera refers to "logic elements"; Xilinx refers to "logic cells"; Tabula tries to bridge the gap by referring to "four-input LUT equivalents." Fundamentally, all these cells are based on LUTs, but they have different numbers and configurations of LUTs, and those LUTs translate into different numbers of usable logic gates.

For years, the marketing departments of Altera and Xilinx have battled each other with PowerPoint presentations and white papers, arguing the relative merits of their logic fabrics. In blogs and online support forums, engineers debate conversion factors from one vendor's gates to another's.

Variables Confuse Comparisons

Even those conversion factors are inadequate, because numerous other variables cloud the picture. Today's FPGAs, especially toward the high end of the product lines, are much more than seas of programmable gates. They also integrate memory, transceivers, hardware multipliers, configurable I/O ports, and other features. Apples-to-apples comparisons are hard to draw.

Additional variables are the relative efficiencies of different synthesis compilers and place-and-route tools. All netlists are not created equal. Perhaps the most straightforward (though tedious) way to compare FPGAs is to buy samples from each vendor, compile a representative design, and measure the gate utilization, power consumption, and performance.

One hint of Tabula's position is pricing. The largest 3PLD that Tabula announced on March 15 (the Abax A1EC06) has 630,000 LUTs and will cost $200 in 2,000-unit volumes when it debuts later this year. Initially, Tabula is aiming for

Another Three-Dimensional FPGA Debuts

As we were finishing this article, another FPGA startup emerged from stealth mode and announced a three-dimensional programmable-logic technology. Tier Logic—which, like Tabula, is based in Santa Clara, California—unveiled its TierFPGA and TierASIC devices on March 10. However, Tier Logic's chips work on a completely different principle than Tabula's 3PLDs.

Tier Logic doesn't use time-division multiplexing to emulate a third dimension of stacked logic. Instead, a Tier-Logic FPGA removes the configuration memory from the base silicon layer and relocates it to an additional silicon layer that implements the SRAM in thin-film transistors. Separating the configuration memory from the programmable fabric allows Tier Logic to move the logic blocks closer together. The goal is to reduce the size, cost, and power consumption of the chip.

One disadvantage of this approach is that the additional memory layer complicates manufacturing. Tier Logic has patented some aspects of the manufacturing process. In contrast, Tabula manufactures its chips in a standard process at TSMC. (Tier Logic is fabless, so the company still relies on foundry manufacturing.)

Perhaps the biggest claimed advantage of Tier Logic's technology is that developers can migrate their designs from an FPGA to an ASIC more easily than before. If a customer's volumes rise to the point where an ASIC would be more economical than an FPGA, it's possible to turn a TierFPGA into a TierASIC. Tier Logic converts the reprogrammable thin-film transistor layer into a single-mask, hard-wired layer with the same bit configuration. Timing doesn't change, so the FPGA design doubles as an ASIC design, without further development work.

Altera offers an FPGA-to-ASIC program called HardCopy, but the conversion requires more engineering effort than Tier Logic claims is required for its technology. Tabula doesn't offer a similar FPGA-to-ASIC option.

For more information about Tier Logic, visit *www.tierlogic.com*.

the upper end of the programmable-logic market. In that segment, dominated by Altera's Stratix-IV and the Xilinx Virtex-6, $200 is a steal for a large-capacity device.

Tabula compares the 630,000-LUT Abax A1EC06 with Altera's Stratix-IV EP4SGX530, an FPGA with 531,200 logic elements. Both devices are fabricated in TSMC's 40nm process. Roughly speaking, the Abax device has 1.2 times more LUTs and 2.2 times more user memory. Pricing for the EP4SGX530 is hard to peg, because Altera lists 32 different versions of the part and doesn't quote volume pricing.

For one chip, Altera's online store charges $9,200 to $11,400. Arrow, a leading distributor, quotes volume pricing as high as $12,490 per unit. Tabula's one-unit price for the Abax A1EC06 is $500 and the volume price is $200. Although Altera and Tabula manufacture their chips in the same fabrication process (40nm CMOS) at the same foundry (TSMC), Tabula's chip is about 25% smaller while offering more gates and memory. Tabula insists it's not sacrificing profit margin to buy market share. Instead, says Tabula, its devices are much cheaper because they're much smaller, so the yields are much better.

As TSMC improves yields, even Tabula admits that Altera's manufacturing costs and retail prices will plunge. (Of course, Tabula will benefit from better yields, too.) Steep discounts for large volumes are common in the FPGA business, especially for favored customers. By next year, Tabula expects competing Stratix-IV devices to be selling in volume for $1,000 to $2,000.

Even then, Tabula's $200 volume-priced part will have an enormous price/performance advantage—if the Abax device proves to be truly competitive with the Stratix-IV device. When Tabula begins sampling in the third quarter, engineers will have to pay only $500 to buy a single part and find out.

Power-Consumption Questions

Another vital consideration is power. For now, Tabula isn't specifying the maximum power consumption of its devices. Power will vary, depending on the customer's design, and Tabula doesn't have many customer designs yet. Tabula says some designs implemented in an Abax 3PLD will use more power than they would in a conventional FPGA, while some designs will use less.

All other things being equal, a smaller die usually implies lower power, and an Abax 3PLD is smaller than an FPGA with a similar number of gates. Fewer transistors tend to use and leak less power. However, Tabula is driving its chips at 1.6GHz to deliver 200MHz of perceived performance. To reconfigure the fabric 1.6 billion times per second, the chip must repeatedly switch among the eight different gate configurations in SRAM—part of the reconfiguration tax.

Power has a linear relationship with clock frequency and rises at a squared rate as the voltage increases. Tabula says first-generation Abax devices operate at only 1.0V. That's by no means excessive, especially for a device clocked at 1.6GHz.

In its 1997 paper on time-multiplexed FPGAs, Xilinx identified power consumption as a drawback of rapid-reconfiguration technology. Xilinx estimated that a small device with a 20- × 20-bit array of configuration SRAM running at only 40MHz would consume "tens of watts."

Of course, that estimate was based on fabrication technology available 13 years ago. But power consumption could be one reason why Xilinx, Altera, and other FPGA companies haven't brought rapid-reconfiguration technology to market, despite research going back 20 years. Another reason could be the vexing problem of placing and routing a circuit design in three dimensions—a problem Tabula claims to have cracked (with help from Einstein and Minkowski).

Reducing Static Leakage

Tabula says its Abax 3PLDs save power in two major ways: they need fewer physical interconnects, because they emulate stacked logic; and they are smaller chips with millions fewer transistors, so they leak less static current.

Static leakage wasn't a significant factor when Xilinx published its 1997 paper. After fabrication technology reached the 90nm node, static leakage became as important, or even more important, than dynamic power. At 40nm, it's very important. Going forward, leakage will become an even larger term in the power-consumption equation. This trend is especially true for fabless companies, like Tabula and its FPGA competitors, because they have less control over process technology than an integrated device manufacturer like Intel does. (TSMC is only now readying a process with low-*k* metal-gate transistors.)

When all factors are considered, Tabula says it has no remarkable power-consumption advantage or disadvantage relative to conventional FPGAs. Abax chips are smaller and have less wiring and shorter wires, but their clock speeds are higher and they must constantly reconfigure their gates. Tabula's tests indicate that some designs will use less power and others will use more power. Customers must decide whether it's worth trading power for lower prices, more memory, and potentially higher performance.

Initially, Tabula is aiming for the high end of the programmable-logic market—and, particularly, for the communications-infrastructure market. That segment is somewhat less sensitive to power consumption and will welcome the large amount of user memory on Abax chips. At the system level, Tabula could have a power advantage if the greater logic and memory capacity of an Abax 3PLD reduce the total chip count. This calculation depends greatly on the system design.

First Chips Target Communications

Communications is a high-growth market with a special hunger for local memory on FPGAs. Tabula delivered preproduction silicon to its first customer (unnamed) a few months ago. The customer is using an Abax 3PLD to replace a Xilinx FPGA and two external memory chips (QDR SRAMs) in a packet-processor design, reducing the overall chip count.

Integration is a key factor with programmable-logic devices. Today's high-end FPGAs offer much more than a fabric of programmable gates. As Figure 9 shows, an Abax 3PLD also integrates memory, I/O controllers, physical-level interfaces (PHYs), and hardware function blocks. Note that these blocks are not implemented in the programmable-logic fabric and, therefore, have none of the advantages or disadvantages of Spacetime technology. Indeed, the more hard blocks integrated in an Abax 3PLD, the less different it seems from a conventional FPGA.

Each of the four Abax devices announced so far has 5.5MB of user memory, two 24-channel serializers/deserializers (SerDes) configurable I/O controllers, and 920 general-purpose I/O (GPIO) ports for parallel I/O. The SerDes controllers support data rates from 55Mb/s to 6.5Gb/s and several popular standards, including PCI Express, Gigabit Ethernet, SGMII, XAUI, Serial RapidIO, Sonet, DisplayPort, Fibre Channel, and SATA. The parallel I/O controllers support external DRAM at DDR, DDR2, and DDR3 speeds, with effective data rates up to 800MHz. Developers can also use the parallel I/O for flat-panel displays and clock interfaces.

The highest-end Abax 3PLD that Tabula announced on March 15 also has 1,280 DSP blocks. These are standard-cell 18- × 18-bit multipliers with 44-bit accumulators. They can perform 18-bit multiply-accumulate (MAC) operations at the chip's base clock frequency of 1.6GHz, producing results in every fold. In contrast, soft multipliers implemented in an eight-fold fabric would run at the user clock

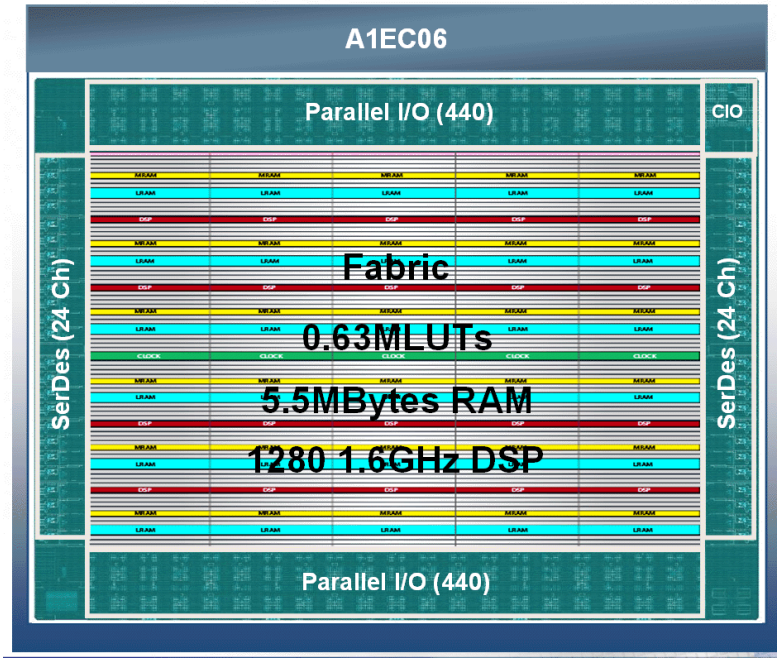

Figure 9. Floor plan of Tabula's Abax A1EC06—the highest-end 3PLD that Tabula has announced so far. It has 630,000 LUTs, 5.5MB of user memory, two 24-channel SerDes controllers, and 920 GPIO ports for parallel I/O. At $200 in volume, it's low priced for a high-end, programmable-logic device with these capabilities.

rate of 200MHz. Hardware multipliers in some competing FPGAs—such as the Altera Stratix-IV EP4SGX530 mentioned above—can run at 600MHz, but they rarely exceed 400MHz in real-world applications.

All four of Tabula's first Abax chips are packaged in a flip-chip ball-grid array (FC-BGA) with 1,936 pins. Obviously, the generous I/O interfaces of these devices contribute to their high pin count. Going forward—perhaps even today—Tabula's chips may be pad limited. Spacetime technology can put more programmable logic on a smaller die, but the die may not have enough room for all the I/O interfaces that customers expect on a high-end device.

This problem could force Tabula to add more programmable logic, memory, or hard-wired function blocks, just to make room for more I/O pads. As fabrication technology continues to shrink transistors, all FPGA vendors face the same problem. However, high-end ASICs have much more logic than even the largest FPGAs, so it's unlikely that customers will complain if FPGA vendors fill their white space with more programmable logic.

The First Four Chips

Table 1 compares the four Abax 3PLDs that Tabula has announced so far. The only significant differences are the number of LUTs (220,000 to 630,000) and the MAC units, which are found in only the highest-end device, the Abax A1EC06.

Tabula has lined up some soft intellectual property (IP), too. Partners are offering DDR2 and DDR3 memory controllers, a PCI Express controller, Ethernet controllers (1GB/s and 10GB/s), and a 32-bit embedded-processor core. That's a sparse catalog of verified IP when compared with the offerings for Tabula's competitors, but all startups find themselves at a similar disadvantage. Any soft IP should work with Abax devices.

The 32-bit embedded-processor core that Tabula has lined up is Freescale's ColdFire V1, an offshoot of Motorola's venerable 68K CPU architecture. IPextreme handles the IP licensing for Freescale. (See *MPR 2/11/08-01*, "Buy SoC IP Like MP3s.")

In contrast, Altera offers its own Nios II processor core for its FPGAs, and Xilinx offers its own MicroBlaze processor core. ColdFire V1 requires about twice as many gates as those 32-bit cores, but it's a more common CPU architecture and isn't bound to any particular FPGA vendor. (See *MPR 6/28/04-02*, "Altera's New CPU for FPGAs," and *MPR 11/13/07-01*, "MicroBlaze v7 Gets an MMU.")

Tabula says a three-sector baseband processor for a Long-Term Evolution (LTE) cellular base station would need only three Abax A1EC06 chips, not including the Gigabit Ethernet switch and host processor. In contrast, a conventional design might need eight DSPs, each with its own external DRAM, to perform the same baseband functions. And Tabula's design would be reconfigurable, providing more flexibility as communications standards evolve.

Feature	Tabula Abax A1EC02	Tabula Abax A1EC03	Tabula Abax A1EC04	Tabula Abax A1EC06
Base Clock Freq	1.6GHz	1.6GHz	1.6GHz	1.6GHz
User Clock Freq (Eight Folds)	200MHz	200MHz	200MHz	200MHz
LUTs	220,000	300,000	390,000	630,000
Register Files*	960 blocks 64 x 9 bits 8 write ports	960 blocks 64 x 9 bits 8 write ports	960 blocks 64 x 9 bits 8 write ports	960 blocks 64 x 9 bits 8 write ports
Large RAM*	480 blocks x 72Kbits Up to 8 ports	480 blocks x 72Kbits Up to 8 ports	480 blocks x 72Kbits Up to 8 ports	480 blocks x 72Kbits Up to 8 ports
Medium RAM*	240 blocks x 36Kbits Up to 16 ports	240 blocks x 36Kbits Up to 16 ports	240 blocks x 36Kbits Up to 16 ports	240 blocks x 36Kbits Up to 16 ports
Total User SRAM	5.5MB	5.5MB	5.5MB	5.5MB
MAC Units (1.6GHz)	—	—	—	1,280 18 x 18 bits
Parallel I/O Ports	920	920	920	920
SerDes 55Mb/s–6.5Gb/s	48	48	48	48
PLLs	44	44	44	44
Package	1,936-pin FC-BGA	1,936-pin FC-BGA	1,936-pin FC-BGA	1,936-pin FC-BGA
Temp Range	-40° to 125°C	-40° to 125°C	-40° to 125°C	-40° to 125°C
Samples	n/a	n/a	Q3-2010	n/a
Production	n/a	n/a	Q4-2010	n/a
Price (2K Units)	$105	$135	$150	$200

Table 1. Feature comparison of Tabula's first four Abax 3PLDs. All these chips are fabricated in TSMC's 40nm CMOS process and run at base clock frequency of 1.6GHz. With eight "folds" or gate configurations, the user clock rate is 200MHz. Distinguishing features are the number of lookup tables (LUTs) and the multiply-accumulate (MAC) hardware blocks. The LUTs in this table are virtual LUTs, assuming eight folds per user clock cycle. To obtain the number of physical LUTs, divide by eight. *Register files, Large RAM blocks, and Medium RAM blocks are part of the total user SRAM. (n/a: data not available.)

Fighting Altera and Xilinx

Together, Altera and Xilinx command almost 90% of the FPGA market—a virtual duopoly. Actel and Lattice Semiconductor have most of the remaining market share, followed by even smaller players like Achronix, Atmel, SiliconBlue, and QuickLogic. Discouraged by these solidly entrenched forces, dozens of semiconductor companies have dropped their programmable-logic devices over the years, preferring to compete in other markets. Several startups have met their doom in the last decade, including Chameleon Systems, CSwitch, MathStar, and Velogix. (See *MPR 7/24/06-02*, "MathStar Challenges FPGAs.")

Tabula's pitch is that Spacetime technology brings advantages in logic density, memory, and performance that will overcome the dominance of Altera and Xilinx. Every FPGA startup makes a similar promise. The problem is that innovations in programmable-logic design usually don't create a big enough advantage to woo customers away from the tried-and-true products. Some FPGA startups have made their debut with a seemingly impressive 2× advantage in some aspect of performance, only to see their lead wiped out when the established companies move to the next process generation. Startups often lack the funding and sales volumes required to stay in the Moore's-law race for long.

Tabula is making its debut in TSMC's 40nm process, the same one Altera uses for its high-end Stratix-IV FPGAs. Xilinx currently manufactures its high-end Virtex-6 FPGAs in UMC's 40nm process and its midrange Spartan-6 devices in Samsung Electronics' 45nm process. In February, Xilinx announced it will move to a new 28nm high-*k* metal-gate process at TSMC and Samsung for future FPGAs, without specifying a target date. Volume production at 28mm is probably at least a year away. Altera, as always, will be compelled to follow.

With Tabula's first Abax chips not scheduled to reach production volumes until the fourth quarter of this year, the window of opportunity is narrowing. By the time Tabula hits its stride, Xilinx may be announcing 28nm parts. Tabula's approximate 3× advantage in logic density and memory density will diminish. A slimmer advantage may not be enough to prevail against Altera and Xilinx.

On the other hand, Tabula's Spacetime technology promises greater benefits from process shrinks than conventional technologies will enjoy. If Tabula can afford to make a timely migration to 28nm, the additional folds enabled by higher clock speeds could widen Tabula's advantage. If Tabula can demonstrate, say, a 4× advantage, Spacetime would have a better claim to being the programmable-logic technology with the strongest forward momentum. It would also reassure prospective customers that Tabula has the financial and technical resources to make a process transition without running out of steam.

Spacetime Makes the Difference

Ultimately, Tabula's survival depends on the effectiveness and efficiency of Spacetime technology. Unlike some previous FPGA startups, Tabula isn't promising to pack physical LUTs, SRAM cells, and function blocks onto a chip better than the leading FPGA vendors can. Against the experts at Altera and Xilinx, that's a losing battle. Tabula had to open another front.

Rapid reconfiguration is the potential game-changer. It's not a new idea, but Tabula has refined it to a higher level than ever before. If it works as advertised, developers needn't bother with the details—they simply perceive a fabric that's larger and denser than the chip's size and cost would suggest. By slicing time to make a two-dimensional chip work like a three-dimensional chip, Tabula is bringing its version of stacked logic to market years before the real thing is practical.

What if Altera, Xilinx, or another competitor does introduce a stacked FPGA? Tabula's Spacetime technology would seem to be obsolete. A real stacked chip wouldn't bear the overhead of Spacetime control logic and wouldn't pay the reconfiguration tax. Even if this evolution happens soon, Tabula says it isn't worried. For one thing, stacked chips would cost more to manufacture than Tabula's chips. For another, Tabula says Spacetime technology is applicable to stacking, too. Each physical layer could rapidly reconfigure itself, creating a *four*-dimensional device—a 4PLD.

A more immediate challenge for any FPGA startup is the quality of its development tools. Competitors have had many years to hone their tools and optimize their place-and-route algorithms. Tabula has had only a few years to duplicate those efforts and extend the scope to a third dimension—not a trivial task. Hiding the details of rapid reconfiguration from developers is an admirable accomplishment. But

Price & Availability

Tabula announced its first four Abax three-dimensional PLDs (3PLDs) on March 15. These devices have 220,000 to 630,000 lookup tables (LUTs), 5.5MB of user memory, two 24-channel SerDes controllers, and 920 general-purpose I/O (GPIO) ports for parallel I/O. One device has 1,280 18-bit hardware multipliers. All devices run at a base clock frequency of 1.6GHz, yielding a user clock rate of 200MHz. They are packaged in a 1,936-pin flip-chip ball grid array (FC-BGA) and are rated for an extended temperature range of –40° to +125°C.

Prices will range from $105 to $200 when devices are purchased in 2,000-unit quantities. The one-unit price is $500. Samples of the Abax A1EC04 chip are scheduled to be available in the third quarter, with volume production commencing in the fourth quarter. Tabula hasn't announced availability for the other three parts. For more information, visit *www.tabula.com*.

For More Information

To review some previous research on virtual three-dimensional FPGAs, see the IEEE paper presented by Xilinx in 1997, "A Time-Multiplexed FPGA":

www.computer.org/portal/web/csdl/doi/10.1109/FPGA.1997.624601

Two recent textbooks also discuss the basic concepts. Both books have similar titles:

Reconfigurable Computing: Accelerating Computation with Field-Programmable Gate Arrays, by Maya B. Gokhale and Paul S. Graham (Springer, 2005).

Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation, by Scott Hauck and André DeHon (Morgan Kaufmann, 2007).

if carried too far, it could impede debugging, especially if the bug is related to reconfiguration.

Tool efficiency is another critical factor. Tabula claims to have the same gate utilization and better interconnect utilization than existing tools, but few customer designs have been ported yet. Inefficient utilization could reduce Tabula's theoretical advantages to irrelevance.

Therefore, Tabula's success or failure depends largely on three questions. Are the performance and power consumption of Abax 3PLDs competitive with conventional FPGAs of similar capacity? Can Tabula successfully weather a process-node transition while proving that Spacetime technology gets better with age? And, are Tabula's development tools as solid as promised—especially for debugging and verifying a design? If Tabula can answer those questions in the positive, its new twist on time could reconfigure the market for programmable logic.