The Linley Group

MICROPROCESSOR report

Insightful Analysis of Processor Technology

TACHYUM TRIES FOR HYPERSCALE SERVERS

VLIW Prodigy Core Pairs With Mesh Fabric, Standard I/O

By David Kanter (October 29, 2018)

Tachyum is developing a 64-core server processor in 7nm technology for hyperscale data centers, targeting tape-out late next year. The design implements a VLIW instruction set with custom vector and matrix-multiplication instructions as well as a custom fabric.

The Prodigy core is a four-bundle eight-wide design with a short 9-stage integer pipeline and 14-stage floating-point pipeline. It packs four integer units, two vector multiply-accumulate units that are 512 bits wide, a vector permute unit, and three load/store pipelines. It can sustain 192 bytes per clock from the 16KB L1 data cache, which is backed by a 512KB L2 cache. Using 8x8 and 4x4 matrix-multiplication instructions, the core can deliver 1,024 and 512 operations per clock, respectively, for machine learning and HPC.

Although the pipeline is largely in order, the microarchitecture has some limited reordering capabilities around load misses; Tachyum claims they'll provide some benefits of out-of-order execution. The startup claims Prodigy should be more efficient than out-of-order designs while delivering similar or better performance.

As Figure 1 shows, the Tachyum server processor comprises a mesh fabric that connects 64 compute tiles, including a Prodigy core and 512KB of configurable L3 cache. The L3 operates as either a private victim cache for better locality or up to a 32MB slice of distributed shared L3 cache. For external memory, the processor has eight DDR4/5 channels or four HBM3 interfaces; the latter are intended for HPC-specific models. The 72 multimode-serdes lanes are typically configured as 64 PCIe 5.0 lanes and two 400G Ethernet links.

Tachyum claims the power-efficient Prodigy core and a simple mesh fabric will reach 4.0GHz in 7nm at 0.825V for a total chip power of 180W, even when the vector units are active. Given that the chip has yet to tape out and has specifications similar to those of other processors, these claims may prove optimistic.

Based in Silicon Valley and Slovakia, the company is looking to raise $30 million to enable tape out in late 2019. It then plans to raise another round and offer products in late 2020. If it can deliver on the technical claims and execute to plan, it may appeal to open-minded hyperscalers.

Packing a VLIW Full of Matrices

Earlier this year, Tachyum broke out of stealth mode with grand performance and power-efficiency claims. Using a new and more efficient instruction set, it claims Prodigy will deliver better performance than Intel x86 processors on

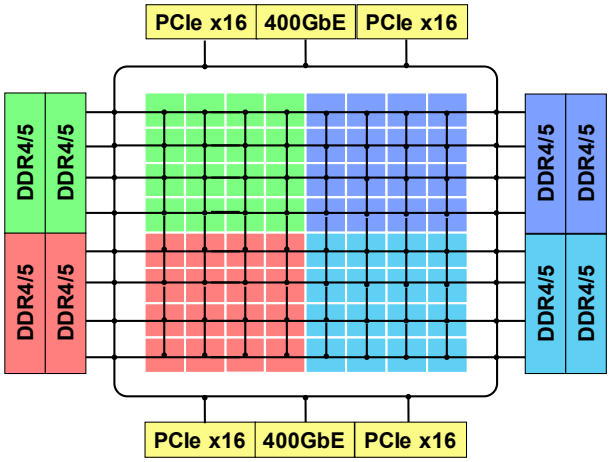

Figure 1. Tachyum T864 server processor. Each of the 64 tiles contains a Prodigy CPU core and a 512KB L3 cache slice. The tiles connect through a mesh-like fabric, and each quadrant of 16 tiles connects to a dual-channel DRAM controller. The memory and I/O are on a separate ring linked to the mesh.

SPECint_rate and SPECfp_rate as well as compelling performance for a variety of workloads such as machine learning and HPC (see *MPR 6/11/18,* "Tachyum Targets Data Centers").

Tachyum's architecture uses VLIW instruction bundles to reduce processor-core complexity by shifting instruction scheduling to the compiler. Each 32-bit instruction word can contain up to two dependent RISC-style operations; bundles contain one, two, or four words. The architectural state includes 32 integer registers and supports predication. But the microarchitecture is designed to enable limited reordering, which the company claims will provide many benefits of out-of-order execution.

To boost throughput, Tachyum's ISA emphasizes packed data. It offers 32 vector registers that are 512 bits wide and can hold packed vectors of double-, single-, and half-precision floating-point (FP) data as well as packed 8-, 16-, and 32-bit integer data. For vector control flow, the ISA has seven mask registers as well. In theory, it brings the company to par with Intel's AVX-512 extensions. For machine learning, Tachyum also developed an unsigned 8-bit FP data type with a 5-bit mantissa and 3-bit exponent. More promising is a set of matrix-multiplication instructions that are conceptually similar to those in Nvidia's Volta and Turing architectures (see *MPR 6/12/17,* "Nvidia's Volta Upgrades HPC, Training").

Bundling Up the Front End

Prodigy is the first implementation of the Tachyum architecture. As Figure 2 illustrates, this in-order core has a short 9-stage integer pipeline, whereas many deeply out-of-order

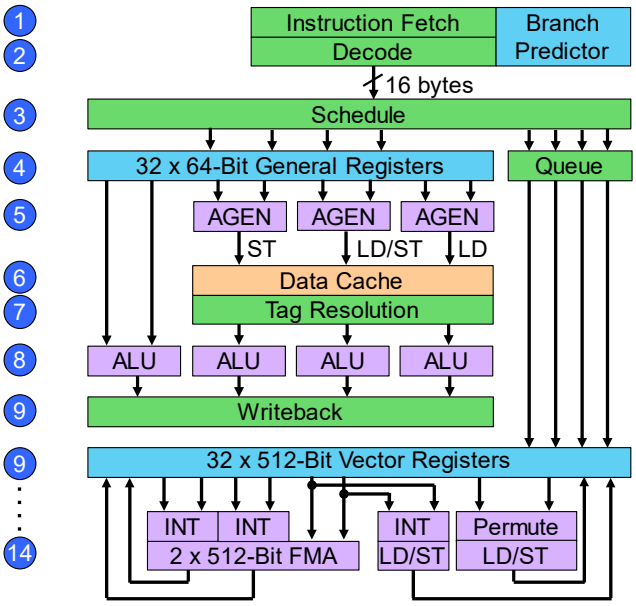

Figure 2. Prodigy core. The design fetches and decodes bundles of four instruction words and executes up to eight micro-ops per cycle, mostly in order. It has 512-bit SIMD and load/store units.

alternatives have 15–25 stages. Multiply-accumulate instructions through the vector pipeline add another five stages. The core is generally designed to sustain a full bundle every clock, dictating the parameters for many portions of the pipeline.

The front end features modern branch handling with a full panoply of predictors that operate in parallel with fetch and decode. It can predict up to two conditional branches every cycle. The branch predictor is tightly coupled to the small 16KB L1 cache, which is two-way set associative and surprisingly provides ECC protection. Instruction-cache lines are 128 bytes, and a miss will initiate hardware prefetching from outer cache levels; instruction misses are filled using a 64-byte bus.

A skewed-Gshare-like algorithm predicts conditional branches using a 12-entry global history buffer. The L1 instruction cache integrates both direct and indirect branch targets, and each of the 1,024 cache lines has fields for up to two target branches. A small 16-entry branch target cache accelerates many predictions and also serves as a loop buffer. A dedicated stack predictor with 16 entries resolves function calls and returns.

Most predicted branches have a maximum penalty of one cycle, whereas statically predicted branches have a two-cycle penalty. The minimum branch-misprediction penalty is eight cycles, and the predictors use checkpoints to reduce state corruption caused by branch mispredictions. Once a fetch address is determined, the fetched instruction bundles go into a 12-entry queue, helping to decouple the rest of the pipeline from front-end stalls.

Mostly Static, With a Side of Reordering

The Tachyum architecture favors moderately complex instruction words that mimic actual code. For example, load and operate is a common idiom that translates into a single word, as does compare and branch. The decoding, however, is straightforward and consumes a single clock, with the micro-ops passing to a scheduler.

Although the ISA is clearly VLIW, the microarchitecture supports limited reordering through various low-cost techniques. The scheduler receives bundles from the front end and can hold up to two sequential bundles for a total of eight instruction words. It reads register inputs from the 32-entry integer register file and issues micro-ops to the execution pipelines as their operands become ready.

Unlike out-of-order designs, the load/store units feed directly into the pipeline ALUs to optimize for common instruction words by reducing the load-to-ALU latency. Any instruction word that leaves either the load/store unit or the ALU unused will have a padding NOP for that unit. To enable some reordering, loads don't stall on a miss; instead, the dependent operation that consumes the load will stall. The scheduler comprises a small replay buffer to allow this stall-on-use behavior. The scheduler can continue to issue from subsequent independent bundles to unstalled

pipelines, supporting modest parallelism around some cache misses.

In a scheduled in-order architecture, memory-access latency is the biggest impediment to good performance, and Prodigy's memory hierarchy is tuned to deliver low latency and high bandwidth. The core includes three address-generation units: one for loads, one for stores, and one for either loads or stores.

To ensure four-cycle load-to-load latency, Prodigy uses a tiny 16KB L1 data cache that's two-way set associative and comprises 64-byte cache lines. This writeback design protects lines with ECC. It's heavily banked and can sustain three full cache-line accesses per clock. To reduce latency, hardware prefetchers track virtual addresses to detect sequential or strided access patterns and to bring in lines early. A small store buffer holds six lines with two entries each and can forward data with a small one- or two-cycle penalty.

A modest private L2 cache backs up and is inclusive of the small L1. This 512KB L2 is two-way associative with 128-byte lines, but the ECC granule is 64B to enable easy write-backs from the L1. An L2 hit incurs an extra six cycles of latency beyond the L1—or a 10-cycle load-to-load latency, which is slightly less than Intel's Skylake-SP. The L2 is banked and can sustain two accesses per clock. Prodigy also has a small 256-entry TLB that is two-way associative and can cache translations ranging from 4KB to 1GB. Each core can sustain several (fewer than eight) outstanding cache misses to the L3 and fabric.

Decoupled Vectors and Matrices

To reduce latency, the integer ALUs connect directly to the output of the memory pipelines. On the integer side, two branch units can compute the direction for conditional branches. There are three ALUs and an integer shifter. Although this organization reduces latency for common load and operate sequences, it has a downside: an instruction word that's only an ALU operation will consume the same nine clock cycles as a more complicated load hit and operate sequence.

The vector pipelines are decoupled from the core's integer and memory side by a 32-entry queue. The scheduler can continue to issue vector operations into this queue despite any stalls from memory accesses. Again, this capability can expose a fraction of the parallelism available to an out-of-order design, at a small cost. Loads require an extra two cycles to forward data to the vector units, largely owing to physical distances.

The vector unit comprises four 512-bit SIMD units and an interesting design to execute the Tachyum ISA's matrix-multiply instructions. Two of these units can perform FP multiply-add and integer vector ALU operations. The third contains an integer vector ALU and an access port for vector loads and stores. The fourth has another access port and a permute unit.

The vector units work with a variety of data formats including Bfloat16, IEEE FP32, IEEE FP64, INT8, INT16, INT32, and the proprietary unsigned 8-bit FP format. To support this wide variety at high throughput, Tachyum implemented the two 512-bit FPUs with 512x8-bit multipliers and adders. The multipliers can be chained together in a tree to operate on larger data types yet sustain full throughput for lower precision. Using SIMD instructions, each core delivers a maximum of 128flops per clock for FP16 data.

To accelerate machine learning, the Tachyum ISA includes 4x4 matrix-multiply instructions that operate on FP32 or FP64 data, as well as 8x8 instructions that operate on 16-bit or smaller data (both integer and floating point). Performing the matrix multiplication in a single instruction allows the input data to be reused so that both input matrices are readable through six 512-bit accesses to the register file. The extra multipliers and adders are specifically for matrix multiplication and can deliver up to 1,024 operations per clock—even faster than packed vectors using similar data types.

Prodigy Meshes Together the System

The Prodigy L3 cache is configurable at boot time to operate in either a private mode or a distributed and shared mode. Physically, it's 512KB per tile and four-way associative with 128-byte lines, and it generally acts as a victim cache for the L2. In private mode, only the local core can evict lines from the L3, whereas any core can read across the fabric. When the local core reads data from the L3, the latency is roughly 10 cycles greater than when it reads from the L2, and it can sustain 64 bytes per clock. The L3 is protected with stronger ECC that corrects double errors and detects triple errors (DECTED).

In distributed mode, addresses are hashed across the L3 instances in each 16-core quadrant on the basis of low-order address bits, and a distributed-directory scheme enforces coherence. Tachyum employs a mesh-like fabric to

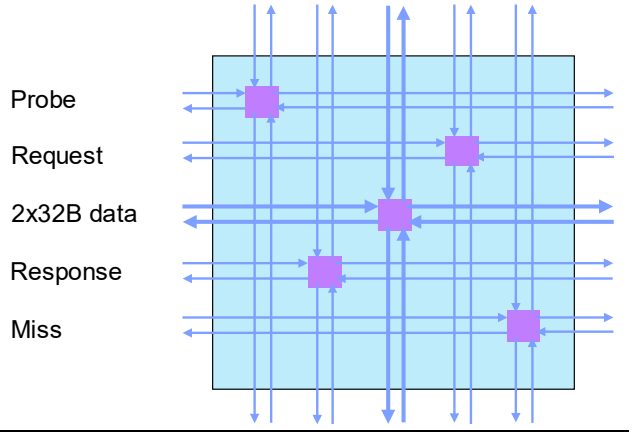

Figure 3. Tachyum fabric. This configuration has a request channel, a response channel, and a bidirectional 32-byte data channel. Requests for the nearest memory controller are routed over a separate fabric.

Price and Availability

Tachyum expects to tape out a 64-core Prodigy processor in late 2019, with production products arriving by late 2020. More online information is available at www.tachyum.com.

link together up to 64 Prodigy cores, the L3 cache, memory, and a variety of I/Os.

The Tachyum fabric comprises three meshes and a separate DRAM-request fabric, as Figure 3 shows. Each segment of the mesh comprises request and response channels as well as two 32-byte data channels, one in each direction. One hop on the mesh takes a single cycle, and the mesh operates at the core clock frequency. In addition, the DRAM-request fabric runs in each quadrant to the local memory controller, but it uses a topology that's simpler than a mesh. The mesh is simple: it has no virtual channels and only 1–2 buffer cycles for back pressure, with credit-based flow control.

The external interfaces are normal by server standards but amped up for next-generation standards that will emerge in 2020 and beyond. The memory interfaces are on the east and west sides of the die. The eight DDR4/5 channels are in pairs; each pair is associated with a quadrant of 16 cores to improve locality. Additionally, each side of the die includes an optional HBM3 memory controller for systems that need greater memory bandwidth. As in Knights Landing, the HBM3 is configurable as a cache or a separate address space. To avoid deadlock, all memory controllers are overprovisioned—each can have 512 outstanding transactions, well beyond the ability to generate requests.

The die's north and south edges are dedicated to a total of 72 PCIe 5.0 lanes. The PCIe PHYs are multimode, and eight lanes are reconfigurable to operate as two 400G Ethernet interfaces. The I/Os sit on a ring directly connected to the memory-interface mesh stops and indirectly connected to the Prodigy cores.

Tachyum plans at least three different server-processor models. The T864 will be fully featured with everything but the HBM3 enabled. The T432 and T216 will enable half of the cores and a quarter of the memory channels. They'll also provide half the PCIe lanes and less networking. These two models are designed to recover defective silicon by disabling functions. Last, the company plans to offer the TH24, an AI/HPC model with all cores and HBM3 enabled for maximum bandwidth.

Silicon and Software Challenges

Tachyum is targeting 7nm technology at TSMC and plans to tape out an 290mm^2 die. It claims it can achieve a 4.0GHz core and fabric frequency running on a 0.825V supply and dissipating 180W, all without custom design. But the design is too immature for activity-based power estimates, and the feasibility of these targets is questionable. First, the power draw from active 512-bit vector units is high and will probably reduce the frequency. For example, a 28-core Skylake-SP using the AVX-512 vector units must decrease the clock frequency by about 30% compared with a 128-bit SSE baseline—and that's using state-of-the-art power management. Second, an architecture's first incarnation rarely hits frequency and power targets. For example, both X-Gene and ThunderX missed theirs and took multiple generations (or an acquisition in the case of ThunderX) to become competitive.

In Prodigy, the caches' low associativity will pressure the memory system. Moreover, Tachyum's mesh fabric has an unusually high frequency target. Most processor companies favor an on-die fabric that operates at about 2GHz for efficiency. Even Intel, despite custom design and its vaunted frequency-optimized process technology, only runs the Skylake-SP mesh at about 2GHz to keep power low (see *MPR 7/17/17*, "Skylake-SP Scales Server Systems").

Tachyum also bears the burden of developing an entirely new software ecosystem from scratch. Since Prodigy is an in-order core, the company's compilers and tools are even more essential and will likely require extensive tuning to deliver high-performance code. History has been unkind to new ISAs running general-purpose workloads, even when companies such as Intel and HP are willing to pour money into the ecosystem. Although Tachyum will offer x86 compatibility via QEMU for user-mode applications, the performance overhead will probably exceed 50%.

Servers Remain the Golden Ticket

Tachyum's plan is to raise $30 million to fund tapeout and even more money to enable production shipments by the end of 2020. The schedule is loose enough that the company should be able to do a metal respin and still deliver in the same year. It may also require additional money to concurrently start on a second-generation design and invest in the software ecosystem.

Tachyum's target market is hyperscale-data-center customers such as Amazon and Facebook. The transition effort is large given the sheer quantity of software, but these companies have many internal workloads and will be tempted by a low total cost of ownership. Tachyum also has engagements with potential customers focusing on specialty workloads such as AI, HPC, and storage that could provide an initial entry point. Unlike large cloud companies, these verticals have smaller software stacks and can work with diverse instruction sets. These specialty applications may yield less revenue, but in the short term, they would help with cash flow.

Given the massive profits available in the server market, Tachyum is taking an educated risk. If it can stick to its plan and deliver compelling performance, hyperscale companies will strongly consider it as an alternative to Intel's Xeon. Customers that primarily run internal workloads (e.g.,

Facebook) are well positioned to adopt new platforms. Infrastructure-as-a-service applications (e.g., Amazon's AWS and Microsoft's Azure) are more difficult to adopt since the code is often unknown. The two challenges for Tachyum are finding investors willing to brave a crowded market and offering sufficient value relative to AMD, Intel, Marvell, and other data-center competitors. ♦