MIPS R10000 Uses Decoupled Architecture

High-Performance Core Will Drive MIPS High-End for Years

by Linley Gwennap

MICROPROCESSOR FORUM 1994

Not to be left out in the move to the next generation of RISC, MIPS Technologies (MTI) unveiled the design of the R10000, also known as T5. As the spiritual successor to the R4000, the new design will be the basis of high-end MIPS processors for some time, at least until 1997. By swapping superpipelining for an aggressively out-of-order superscalar design, the R10000 has the potential to deliver high performance throughout that period.

The new processor uses deep queues decouple the instruction fetch logic from the execution units. Instructions that are ready to execute can jump ahead of those waiting for operands, increasing the utilization of the execution units. This technique, known as out-of-order execution, has been used in PowerPC processors for some time *(see **081402.PDF**)*, but the new MIPS design is the most aggressive implementation yet, allowing more instructions to be queued than any of its competitors.

Taking advantage of its experience with the 200-MHz R4400, MTI was able to streamline the design and expects it to run at a high clock rate. Speaking at the Microprocessor Forum, MTI's Chris Rowen said that the first R10000 processors will reach a speed of 200 MHz, 50% faster than the PowerPC 620. At this speed, he expects performance in excess of 300 SPECint92 and 600 SPECfp92, challenging Digital's 21164 for the performance lead. Due to schedule slips, however, the R10000 has not yet taped out; we do not expect volume shipments until 4Q95, by which time Digital may enhance the performance of its processor.

Speculative Execution Beyond Branches

The front end of the processor is responsible for maintaining a continuous flow of instructions into the queues, despite problems caused by branches and cache misses. As Figure 1 shows, the chip uses a two-way set-associative instruction cache of 32K. Like other highly superscalar designs, the R10000 predecodes instructions as they are loaded into this cache, which holds four extra bits per instruction. These bits reduce the time needed to determine the appropriate queue for each instruction.

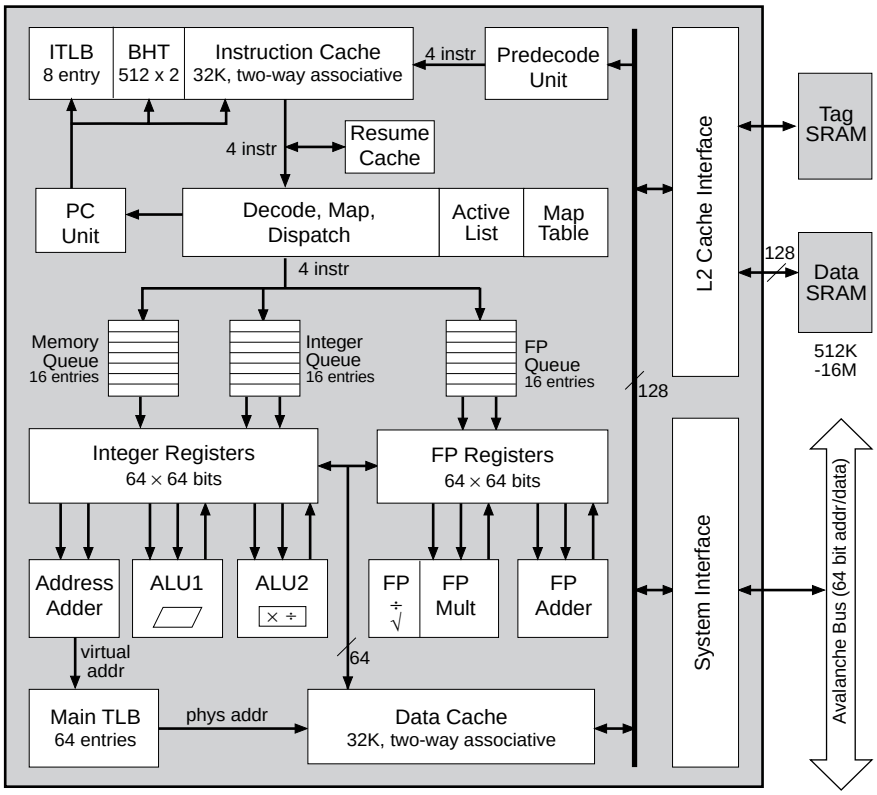

Figure 1. The R10000 uses deep instruction queues to decouple the instruction fetch logic from the five function units.

The processor fetches four instructions per cycle from the cache and decodes them. If a branch is discovered, it is immediately predicted; if it is predicted taken, the target address is sent to the instruction cache, redirecting the fetch stream. Because of the one cycle needed to decode the branch, taken branches create a "bubble" in the fetch stream; the deep queues, however, generally prevent this bubble from delaying the execution pipeline.

The sequential instructions that are loaded during this extra cycle are not discarded but are saved in a "resume" cache. If the branch is later determined to have been mispredicted, the sequential instructions are reloaded from the resume cache, reducing the mispredicted branch penalty by one cycle. The resume cache has four entries of four instructions each, allowing speculative execution beyond four branches.

The R10000 design uses the standard two-bit Smith method to predict

branches. A 512-entry table tracks branch history. To help predict subroutine returns, which have variable target addresses, the design includes a single-entry return-address buffer: when a subroutine returns to the caller, the address is taken from this buffer instead of being read from the register file, again saving a cycle. This technique is similar to Alpha's four-entry return stack but, with only a single entry, it is effective only for leaf routines (those that do not call other subroutines).

Register renaming *(see **081102.PDF**)* allows the R10000 to reduce the number of register dependencies in the instruction stream. The chip implements 64 physical registers each for integer and floating-point values, twice the number of logical registers. As instructions are decoded, their registers are remapped before the instructions are queued. The target register is assigned a new physical register from the list of those that are currently unused; the source registers are mapped according to the current mapping table.

The mapping table has 12 read ports and 4 write ports for the integer register map, and 16 read ports and 4 write ports for the floating-point register map. These ports support an issue rate of four instructions per cycle. Fully decoded and remapped instructions are written to one of three queues: integer ALU, floating-point ALU, or load/store. All four instructions can be written to a single queue if necessary. Because register dependencies do not cause stalls at this point, about the only situation that will prevent the chip from dispatching four instructions is if one or more of the queues are full. This situation rarely arises, as all three queues have 16 entries.

One minor exception is that integer multiply and divide instructions can stall the dispatch unit because they have two destination registers instead of one, causing a port conflict in the mapping logic. In a given cycle, no instructions after an integer multiply or divide will be queued, and if one of these instructions is the last in a group of four, it will be delayed until the next cycle.

When instructions are queued, they are also entered into the active list. The active list tracks up to 32 consecutive instructions; the dispatch unit will stall if there are no available entries in the active list. This structure is similar to AMD's reorder buffer *(see **081401.PDF**)* but avoids time-consuming associative look-ups by holding the renamed register values in the large physical register file. Note that there are enough physical registers to hold the 32 logical registers plus one result for each instruction in the active list.

Dynamic Instruction Issue

The instruction queues are somewhat misnamed, as instructions can be dispatched from any part of the queue, cutting in front of other instructions, unlike a true FIFO, through which instructions march in lockstep. These dynamic queues are best viewed as holding

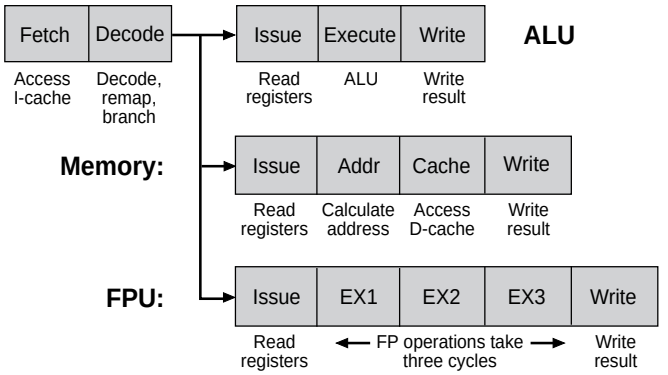

Figure 2. The decoupled design of the R10000 allows function units to have pipelines of different lengths.

tanks (or "reservation stations" in PowerPC terminology). At any given time, some instructions in the queue are executable, that is, their operands are available; other instructions are waiting for one or more operands to be calculated or loaded from memory.

Each cycle, the processor updates the queues to mark instructions as executable if their operands have become available. Each physical register has a "busy" bit that is initially set when it is assigned to a result. When a result is stored into that register, the busy bit is cleared, indicating that other instructions may use that register as a source. Because the renaming process ensures that physical registers are not written by more than one instruction in the active list, the busy bit is an unambiguous signal that the register data is valid.

Executable instructions flow from the queues to the register file, picking up their operands, and then into the appropriate function unit. Only one instruction can be issued to each function unit per cycle; if there are two or more executable instructions for a particular function unit, one is chosen pseudo-randomly.

The R10000 designers kept the instruction prioritization rules to a minimum, as these rules can create complex logic that reduces the processor cycle time. For ALU1, conditional branches are given priority to minimize the mispredicted branch penalty. ALU2 and the FP multiplier give priority to nonspeculative instructions that were ready the previous cycle but were not issued; this rule generally gives priority to older instructions. The address queue prioritizes instructions strictly by age, issuing instructions out of order only when older instructions are stalled.

Function Units Proceed at Own Pace

Each function unit has its own pipeline, as Figure 2 shows. Integer ALU operations execute in a single cycle (stage 4) and store their results in the following stage. Loads and stores calculate addresses in stage 4 and access the cache and TLB in parallel during stage 5. Data is ready for use by the end of stage 5, creating a single-

Price & Availability

The MIPS R10000 processor is not yet announced as a product. MIPS Technologies expects that the part will sample in 3Q95, with volume shipments in 4Q95. NEC and Toshiba will market the R10000. For more information, contact Steve Proffitt at MIPS Technologies (Mt. View, Calif.) at 415.390.4125; fax 415.390.6170.

cycle load-use penalty. Simple floating-point calculations are completed in three cycles, but operations such as divide and square root take longer, as Table 1 shows.

Both integer units contain a full ALU, but only ALU1 contains a shifter, while only ALU2 accepts integer multiply and divide instructions. All conditional branch instructions are handled by ALU1. Both integer units are fed by the same instruction queue.

There are two floating-point units: one for addition and similar functions, the other for multiplies and long-latency operations such as divide and square root. The latter unit can execute a multiply, a divide, and a square-root operation in parallel but has a single set of register ports. Both FP add and FP multiply instructions have a latency of just two cycles, helping to produce a high SPECfp92 rating. The R10000 implements the MIPS IV instruction set *(see **071102.PDF**)*, which includes a multiply-add combination; this operation hops from one unit to the other with a total latency of four cycles.

Floating-point conditional branches are not issued to either FPU, as they require only the checking of a bit in the FP condition-code register. These instructions are handled by a small block of logic that can be individually accessed. Thus, the R10000 can actually issue six instructions in a single cycle if one is an FP branch.

The pipelines can be viewed as a six-cycle pipeline for integer instructions and a seven-cycle FP pipeline. Unlike the bloated R4000 "super" pipeline, the new processor keeps the load-use penalty to a single cycle and the mispredicted branch penalty to two cycles if the instructions are in the resume cache, or three if not. In the optimum case, a conditional branch instruction flows quickly through the pipeline, but some branches must wait for operands. If such a branch eventually is discovered to be mispredicted, the mispredicted branch penalty can be many cycles.

	Single-Precision		Double-Precision	
	Issue Rate	Latency	Issue Rate	Latency
Integer Multiply	6 cycles	6 cycles	10 cycles	10 cycles
Integer Divide	35 cycles	35 cycles	67 cycles	67 cycles
FP Add	1 cycle	2 cycles	1 cycle	2 cycles
FP Multiply	1 cycle	2 cycles	1 cycle	2 cycles
FP Mult-Add	1 cycle	4 cycles	1 cycle	4 cycles
FP Divide*	11 cycles	11 cycles	18 cycles	18 cycles
FP Sq Root*	17 cycles	17 cycles	32 cycles	32 cycles
FP Recip Sq Rt*	17 cycles	28 cycles	32 cycles	50 cycles
Integer Load	1 cycle	2 cycles	1 cycle	2 cycles
FP Load	1 cycle	3 cycles	1 cycle	3 cycles

Table 1. The R10000 executes the most common floating-point operations in two cycles. Note that, once started, divide and square root functions can execute in parallel with other FP operations. *preliminary values; final results may be better

Active List Maintains Order

The active list tracks which instructions have been executed. It ensures that, as instructions execute speculatively and out of order, the CPU presents an in-order model to software and to the rest of the system. This feat requires undoing instructions when an exception or incorrect speculation (mispredicted branch) occurs.

An instruction is retired when that instruction and all preceding instructions have been completed. Up to four instructions can be retired per cycle, with a maximum of one store. To retire an instruction, the processor simply frees the old copy of that register in the mapping table, allowing it to be reused. This action commits the new value of the logical register.

If an instruction causes an exception, the active list is used to undo the results of any successive instructions that have been executed out of order; results are undone by simply unmapping the physical register that received the result. Because the out-of-order instructions have not yet been retired, the register file contains the old data in other physical registers. Instructions can be undone at a rate of four per cycle, for a maximum exception latency of eight cycles.

To speed the handling of mispredicted branches, the processor saves the mapping table—and other important state—in shadow registers whenever a branch is speculatively issued. If the branch is determined to be mispredicted, the processor can resume execution with the correct state. Without this feature, the active list would have to be repaired four entries at a time, as if an exception had occurred. Shadow storage is provided for speculation of four conditional branches.

Associative Caches at All Levels

Both the instruction and data caches are 32K and two-way set associative, twice the size and twice the associativity of the R4400's caches. The data cache has a 32-byte line size, while the instruction cache uses 64-byte lines. The data cache is nonblocking; it will continue to respond to accesses even while a miss is being handled.

Unlike all R4x00 processors, the new design uses six-transistor (6T) cells for the on-chip caches. This technique, also used by most other microprocessor vendors, eliminates an extra poly layer from the manufacturing process but can increase the size of the cell. MTI is making the switch because 4T cells do not scale well at

geometries below 0.5 microns, where the R10000 is clearly headed in the future.

The TLB uses the traditional MIPS design in which each pair of entries maps adjacent pages; the R10000 has 64 pairs of entries in the unified TLB, 16 more than its predecessor. The instruction micro-TLB is expanded to eight single-page entries.

The secondary cache uses a 128-bit interface (plus ECC) to the data array, which can range from 512K to 16M in size, along with a 26-bit connection to the external tags. Including tags, the cache can be implemented with ten ×18 parts. The supported ratios of CPU-to-cache speeds are 1, 2/3, 1/2, 2/5, and 1/3. With a 200-MHz processor, systems will typically use a 100- or 133-MHz cache. To simplify design at these clock speeds, the external cache uses synchronous SRAMs.

The external cache is two-way set associative, increasing the hit rate over the direct-mapped cache used by the R4400. The R10000 includes an 8K × 1 way-prediction table that logs the most recently used way (set) for each line in the external cache. On each access to the external cache, the processor reads the data from the way predicted by the table. At the same time, the tags for both ways are read through the separate tag interface. (Since it takes at least two cycles to refill an internal cache line, there is always adequate time to read both tags.)

CLARENCE TOWERS

Chris Rowen of MIPS Technologies elaborates on the decoupled architecture of the R10000.

If the tag comparison indicates a hit in the predicted way, the data read from the array is valid. If the requested data is in the other way, a second access of the data array must be performed. If neither tag matches the requested address, a refill from main memory is initiated.

This algorithm maintains the hit rate of a two-way cache without external logic to select the proper data from the array or the high pin count needed to incorporate this logic on chip. It also avoids the time needed to multiplex the data. The downside is the extended latency if the way prediction is incorrect. With a half-speed cache, the load-use penalty for an external cache hit is 6 cycles for the predicted way and 12 cycles for the mispredicted way. MTI estimates that the way prediction will be correct about 90% of the time.

The system bus has been completely redesigned from the R4x00's SysAD bus. The new Avalanche bus is also a 64-bit multiplexed address/data bus, but it supports split transactions (up to eight pending transactions) and full cache consistency, allowing multiple processors to connect directly to the bus. The bus timing is programmable independently of the cache speed, with divisors down to 1/4 of the CPU clock. For practical purposes, the bus is limited to about 100 MHz in a uniprocessor design and about 80 MHz with multiple processors.

Because of the multiplexed design, three cycles are required for each address: one to turn the bus around, one to send the address, and one to turn the bus back. A single transaction can return 32 words in 16 cycles. Thus, at 80 MHz, the maximum sustainable bandwidth is 539 Mbytes/s. MTI's Rowen says that this bandwidth is sufficient for four R10000 processors to operate without data starvation. The 620 system bus, however, offers about twice the sustainable bandwidth (1,067 Mbytes/s) using a 128-bit nonmultiplexed interface.

MIPS CPU Vendors Dwindle to Two

Although the design is not yet complete, Rowen estimates that the die will be 298 mm^2 in area. This is slightly more compact than the other next-generation RISC processors shown in Table 2, but still twice the size of the R4400 in the same 0.5-micron process. The new chip, in fact, takes advantage of four metal layers, while the R4400 makes do with only two.

Of the six original MIPS processor vendors, only giants NEC and Toshiba have signed up to produce the R10000. IDT is a third source for R4400 processors but is currently running its 0.5-micron process on 150-mm wafers, which are not efficient for such a large die. While IDT may choose to market the R10000 at a later date, it appears that there will initially be only two sources, just as in the PowerPC world.

This announcement covers only the CPU design; there are no specific details on pricing and availability from either of the vendors. Volume shipments are expected at the end of 1995. The processor initially will be sold in a 527-pin ceramic PGA; the vendors are investigating a 339-pin multichip module that would contain the CPU along with 1M of cache.

Next-Generation CPUs Are Similar

Table 2 shows how the R10000 compares with the other announced next-generation processors. (HP disclosed its PA-8000 design at the recent Microprocessor Forum but did not provide enough detail for this chart; the PA-8000 will be described in our next issue.) These designs have converged on several common points but differ in a few key areas.

All dispatch four instructions per cycle. This rate is a convenient power of two, with eight too large a value for this generation. Most cache predecoded instructions, giving up die area to speed the pipeline. All use dynamic branch prediction with two bits of history per branch.

Register renaming is becoming popular as a way to support speculative and out-of-order execution. All include control logic for the external cache on chip to tighten the coupling between the CPU and external cache. While the R10000 and PowerPC 620 have a dedicated interface to this cache, the 21164 and UltraSparc share the external cache interface with the system bus.

In the physical realm, all use comparable 0.5-micron CMOS processes with four metal layers. The die sizes are nearly identical, within 5%, and are very large. With high transistor counts and fast clock speeds, the power dissipation of these chips also breaks records at 30 watts or more. For the system interface, glueless MP is a must and synchronous SRAMs are gaining in popularity.

The biggest differences are in the cache organizations and the tradeoffs between clock speed and complexity. Both the R10000 and the 620 take a relatively straightforward approach to cache design, simply doubling the size and associativity of the on-chip caches from the previous generation. UltraSparc chose to maintain the cache size at 32K total but added a single-cycle external cache to make up the difference. Digital was forced into a radical two-level on-chip cache by its extremely high clock rate.

Digital's designers push the clock frequency as high as possible, then add complexity without impacting the frequency. The 21164 is perhaps the least complex of this group—although still far beyond relatively simple superscalar designs—as evidenced by the number of logic transistors. The R10000 is at the other end of the complexity scale, with the most renaming and largest amount of speculative and out-of-order execution. All else being similar, the more complex a processor, the lower the clock rate.

The R10000 doesn't quite fit this mold, as its clock speed is purportedly higher than those of the less complex 620 and UltraSparc. We have yet to see if these processors meet their rated clock speeds, however, so it may be premature to make this analysis. Interestingly, the estimated performance numbers show the R10000 at just 1.5 SPECint92/MHz, while the 620 and UltraSparc deliver about 1.7 SPECint92/MHz; the more complex design should be more efficient in work per clock cycle.

The R10000 also appears to be slightly less expensive to manufacture than its competitors. It crams more logic transistors and more cache transistors than UltraSparc into a smaller die and avoids the higher wafer cost of the 21164 process. Again, it is premature to make precise comparisons until we see the final die for these chips; until then, we can consider them all to be comparable in cost.

	R10000	PPC 620	UltraSparc	21164
Clock Speed	200 MHz	133 MHz	167 MHz	300 MHz
Cache Size	32K/32K	32K/32K	16K/16K	8K/8K/96K
Dispatch Rate	four	four	four	four
Function Units	five	six	nine	four
Predecode Bits	4 bits	7 bits	4 bits	none
Rename Regs	32 int, 32 fp	8 int, 8 fp	none	none
Branch History	512 x 2	2K x 2	512 x 2	2K x 2
Out of Order	32 instr	16 instr	none	6 loads
Ext Cache Cntl	on chip	on chip	on chip	on chip
Synch SRAM	yes	yes	yes	optional
Glueless MP	yes	yes	yes	yes
Power Usage	30 W	30 W	30 W	50 W
IC Process	0.5-micron	0.5-micron	0.5-micron	0.5-micron
Metal Layers	four	four	four	four
Logic Transistors	2.3 million	2.2 million	2.0 million	1.8 million
Total Transistors	5.9 million	6.9 million	3.8 million	9.3 million
Package Type	527-pin CPGA	625-pad CBGA	521-pin CPGA	499-pin CPGA
Die Size	298 mm^2	311 mm^2	315 mm^2	298 mm^2
Est Mfg Cost	$320	$380	$420	$430
First Silicon	4Q94 (est)	7/94	10/94	2/94
Volume Parts	4Q95 (est)	3Q95 (est)	3Q95 (est)	1Q95 (est)
SPECint92 (est)	>300 int	225 int	275 int	330 int
SPECfp92 (est)	>600 fp	300 fp	305 fp	500 fp

Table 2. Next-generation processors in the MIPS, PowerPC, SPARC, and Alpha families, all due in 1995, share many common features but differ in clock speed, complexity, and performance.

The Future of MIPS

Historically, MIPS Technologies has produced a new mainstream processor core every three years. The company does not plan to release a major redesign of the R10000 for quite some time. In the interim, the R10000 will go through process shrinks and design tweaks, much as the R4000 did, to further increase performance, but the long-term competitiveness of the MIPS high end rests entirely with the R10000.

If the team can meet its design goals, the product line will be in good shape. At 200 MHz, the 300 SPECint92 goal appears conservative for such a complex processor; final performance numbers should approach those of the 300-MHz 21164, although Digital may have faster parts available by the time the R10000 ships. The MIPS chip has a good shot at outrunning the PowerPC 620 and UltraSparc.

The clock-speed goal appears aggressive, however. The company hopes to achieve the same clock rate as the R4400 in the same IC process, but the new chip has a shorter pipeline and more complex logic in critical timing paths. Without a high clock frequency, the R10000's performance advantage will quickly disappear.

The R10000 sets new standards for instruction-level parallelism, allowing the function units to take advantage of the parallelism inherent in the code rather than enforcing some arbitrary program ordering. Techniques such as register renaming and out-of-order execution will be used widely in future processor designs. The new CPU promises to restore vitality to the high end of the MIPS processor line. To fulfill this promise, however, working parts must be fabricated. ♦

