

Hitachi SH-4 Gets Graphically Superscalar

New Superscalar Core, FP Unit Boost 3D Rendering Speeds to 1.4 GFLOPS



by Jim Turley

Hitachi continues to press its advantage in low-cost, high-volume microprocessors, this time adding two-way superscalar execution and acceleration for floating-point 3D geometry. The one-two punch, not expected to land for another 18 months, should belt SuperH 3D performance far ahead of chips with multimedia or signal-processing extensions.

Coming just weeks after Hitachi's SuperH was anointed one of two architectures for running Windows CE (see [1012MSB.PDF](#)), the announcement of the new core should boost Hitachi's position in the consumer-electronics and games markets.

During his presentation at last week's Microprocessor Forum, Hitachi's Jim Slager provided a glimpse of the SH-4, the fourth generation of Hitachi's popular RISC architecture. It is the first SuperH chip to offer two-way superscalar execution and the first to have a striking new floating-point unit optimized for geometry operations common in 3D rendering for games.

Samples of SH-4 chips are still 6 months away, with production 12 months behind that; pricing has not been disclosed. When it ships, the new part is expected to reach 200 MHz and deliver in excess of 300 MIPS. A new 0.25-micron process will hold power consumption to less than 2 watts.

New Registers Support 3D Geometry Code

The SuperH's modest register set was extended in various ways to accommodate the enhanced floating-point unit and to better support the parallel MAC hardware. Figure 1 shows how the basic set of sixteen 32-bit registers has been joined with two additional banks of sixteen 32-bit registers.

Registers f0-f15 are 32-bit floating-point registers, which first appeared in the FP-equipped SH-3E (see [091603.PDF](#)). Registers b0-b15 are new to the SH-4 and are used in only a handful of new 3D geometry instructions (described later).

The most significant new instructions are for 3D geometry. Given Hitachi's success in the game-console market with the Sega Saturn, this course of action is hardly surprising. The geometry instructions vastly improve the SuperH family's performance in the setup portion of 3D graphics applications, which is not aided by MMX, VIS, and similar instruction-level extensions.

Surprisingly, Hitachi was able to add these significant new features while staying within the architecture's fixed 16-bit instruction word. The SuperH's impoverished opcode map leaves precious little room for enhancement; as it is, SuperH has only one condition code, and conditional-branch offsets are limited to just 256 bytes. Adding FP instructions to the SH-3E ate up nearly all permutations of the F-line opcodes; the SH-4 uses the remaining two.

The first new instruction, FIPR (floating-point inner product), multiplies and accumulates two four-element vectors, a common operation for rendering 3D objects. The second new instruction, FTRV (floating-point transform vector), is even more ambitious, multiplying a 4x4 matrix with a four-element vector to produce a new (transformed) vector, all in seven clock cycles. At 200 MHz, the SH-4 can initiate a new FTRV every 35 ns for a whopping 1.4 GFLOPS.

3D Geometry Tasks Are Complex

In conventional 3D software, complex surfaces are broken up into a series of polygons, usually triangles. The boundaries of these polygons must be represented by a minimum of three (for a triangle) sets of x, y, and z coordinates, each of which is often stored as a single-precision floating-point value. Although this information is sufficient to determine the polygon's position, its orientation is still ambiguous.

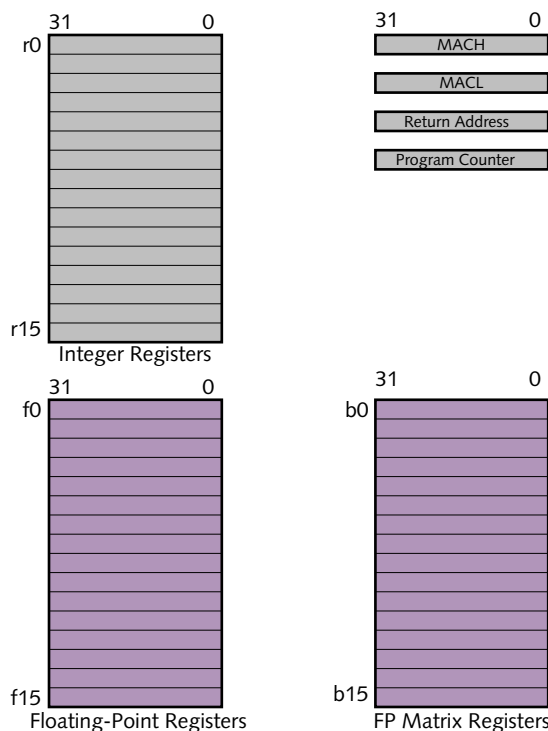


Figure 1. Hitachi's new SH-4 processor core includes three sets of 16 registers. The FP registers first appeared in the SH-3E; the FP matrix registers are used only by the SH-4's matrix transformation instructions.

For rendering purposes, each polygon must have a front and a back. The front of a surface is frequently determined by selecting one point on the surface and assigning it a direction (surface normal), expressed in degrees of rotation from some reference. A triangular surface is always planar (hence the preference for triangles when deconstructing complex objects), so a single four-element vector consisting of x, y, and z coordinates plus the surface angle, w, can act as a representative sample for the entire surface.

To render a surface, the application must calculate its brightness, reflectivity, or visibility based on the incident angle of the light source (or sources), which is also conveniently represented by a four-element vector. Thus, vector multiplication, shown in Figure 2, is a critical inner-loop function for most graphics software. Hitachi's designers created a large parallel floating-point multiply-accumulate block in the SH-4 specifically for this purpose.

The FIPR instruction uses two sets of four consecutive FP registers from f0 to f15. As Figure 3 shows, the four single-precision FP multiply operations are carried out in parallel, using four identical FP multiply units, in a single cycle. In the cycle that follows, the results of these four operations are then added together, with the single-precision result written back in place of the last element of the vector, w₂, in the last cycle.

This operation would typically be repeated once for every polygon to be rendered and again for each additional light source, if any. The result of each operation is treated as the incident angle of the light source, a value that determines a polygon's brightness or visibility. For rapidly moving 3D objects, the speed of this operation determines the complexity of the objects that can be rendered in one screen-refresh period. In the world of video games, this complexity translates into more convincing graphics.

Coordinate Transform Function Crunches 288 Bits

Just as important as calculating the brightness of a polygon is determining that polygon's location. For that, Hitachi added the FTRV instruction, an impressively single-minded operation whose purpose is to speed translations and rotations.

Where FIPR multiplies two vectors, FTRV multiplies and adds a vector and a 16-element matrix. As before, the four-element vector illustrated in Figure 4 holds the x, y, z, and w (surface normal) coordinates of a surface. The matrix, a₁₁-a₄₄, represents a group of x, y, z, and w transformations on each of those coordinates.

The vector elements are taken from four consecutive FP registers, as with the FIPR instruction. To hold the entire matrix, however, the SH-4 core must use the whole register set, b0-b15. Again, the four multiply operations are carried out in parallel, with the four results added together. The coordinate transformation is an iterative process, which the FTRV instruction repeats four times, once for each row of the matrix. The first single-precision result is deposited in the first FP register, replacing vector element x₁. The next

$$\begin{bmatrix} x_1 & y_1 & z_1 & w_1 \end{bmatrix} \times \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ w_2 \end{bmatrix} \rightarrow w_2$$

$$w_2 = (x_1 \times x_2) + (y_1 \times y_2) + (z_1 \times z_2) + (w_1 \times w_2)$$

Figure 2. The FIPR (floating-point inner product) instruction multiplies two four-element vectors, adds them, and returns a single-precision result. The instruction is useful in calculating the angle at which two polygons (or a polygon and a light source) intersect.

result replaces y₁ during the cycle that follows, and so on. At the conclusion of the entire operation, seven clocks later, all elements of the matrix have been replaced by their transformed coordinates.

Obviously, the geometry operations place huge demands on the datapath of the SH-4 core. The FTRV instruction, for example, transfers 288 bits every cycle. Most of the new core is devoted simply to the datapath and to the tripleported floating-point registers, according to Hitachi.

Geometry Functions Are Seriously Data-Hungry

Because the geometry operations are often repetitive, Hitachi designed the FP datapath to forward its results. The results of the FIPR (or any FP) instruction are available to the subse-

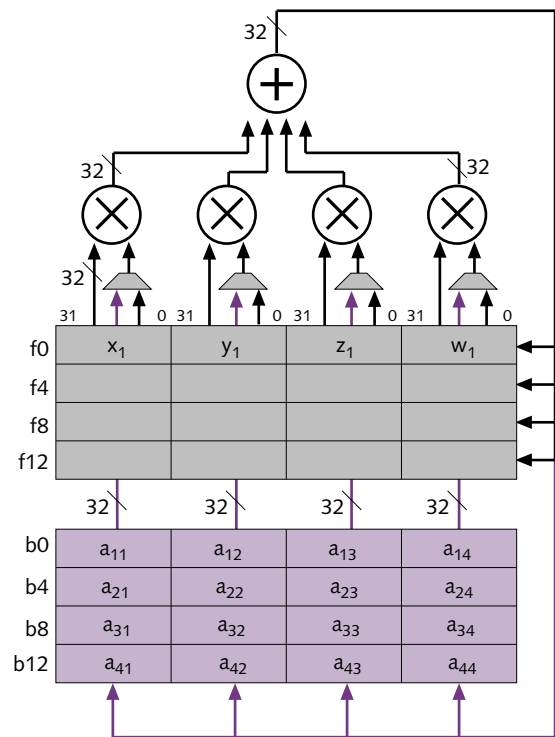


Figure 3. SH-4's single-precision SIMD FPU can carry out four 32x32-bit multiplications and a 32-bit addition in a single cycle. Two new instructions sequence several of these operations to create powerful graphics primitives.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \times \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ w_1 \end{bmatrix} \longrightarrow \begin{bmatrix} x'_1 \\ y'_1 \\ z'_1 \\ w'_1 \end{bmatrix}$$

$$\begin{aligned}
 x'_1 &= (a_{11} \times x_1) + (a_{21} \times y_1) + (a_{31} \times z_1) + (a_{41} \times w_1) \\
 y'_1 &= (a_{12} \times x_1) + (a_{22} \times y_1) + (a_{32} \times z_1) + (a_{42} \times w_1) \\
 z'_1 &= (a_{13} \times x_1) + (a_{23} \times y_1) + (a_{33} \times z_1) + (a_{43} \times w_1) \\
 w'_1 &= (a_{14} \times x_1) + (a_{24} \times y_1) + (a_{34} \times z_1) + (a_{44} \times w_1)
 \end{aligned}$$

Figure 4. The FTRV (floating-point transpose vector) instruction multiplies each element of a four-element vector with its corresponding element in a 4x4 matrix, adds the results, and returns a new vector. This operation is used to translate or rotate a point in 3D space.

quent FP operation after three cycles, but they are not written to the register file until the following cycle. This allows the SH-4 to maintain a three-clock repeat rate on the most performance-critical geometry functions.

The repetitious nature of the geometry functions, particularly FTRV, also presents a serious potential bottleneck. A coordinate transformation accesses eight 32-bit registers (four FP vector registers and four FP matrix registers) for four cycles. Without a means of emptying and refilling the four vector registers, FTRV instructions could not be executed consecutively, seriously hampering performance.

Here the superscalar nature of the SH-4 is especially valuable. Because the geometry datapath is separate from both the integer datapath and the conventional floating-point datapath, new integer and FP instructions can execute in parallel with the geometry calculations. The SH-4 has just enough time to increment the address pointers (an integer operation) and load new vector elements (FP ops) before the next FTRV instruction needs the data. Assuming all goes well, the SH-4 can theoretically maintain an unbroken sequence of coordinate transformations at the rate of one every four cycles.

However, all might not go well. Hiding the loads and stores under the geometry operations depends on single-cycle access to memory—in other words, a cache hit. Should a single cache miss occur for either an instruction or for data, the entire process grinds to a halt until the miss is served. Given the relative size of complex 3D databases versus a normal data cache, some misses seem inevitable.

			Dispatched together
CMP/GT	r1, r2	; compare r1<r2	
BT	label	; branch if true	
ADD	#4, r3	; add two registers if compare=false	
label: MOV	r4, r5	; copy register	
			Dispatched together

Figure 5. The SH-4's superscalar core can execute two instructions at once. When a comparison and a conditional branch are paired, the instruction immediately following the branch is executed conditionally; the entire construct requires only two cycles to execute, regardless of the outcome of the comparison.

Two factors work in Hitachi's favor here. First, for rendering relatively small objects with, say, fewer than a dozen polygons, data-cache misses are probably unlikely. Second, the transformation matrix, a_{11} - a_{44} , is generally constant and gets reused over several related transformations. Once those 16 registers are loaded, they don't need to be changed until a different object is to be rendered.

Superscalar, Conditional Execution a SuperH First

The SH-4's other big enhancement is its two-way superscalar execution. The chip can retire two instructions per clock cycle, with certain restrictions. The SH-4 cannot dispatch two similar instructions (ADD with ADD, float with float, etc.), nor can it mix certain complex or multicycle instructions with any others. Within these confines, however, the chip performs happily, mixing integer and floating-point code with no register or datapath conflicts.

A more subtle improvement to the SH-4 prevents pipeline disruptions in some cases. Similar to conditional execution, the feature works by recognizing a special case where a comparison instruction is dispatched alongside a conditional branch with an offset of zero.

Because the SuperH ISA has only two conditional-branch instructions (branch if true/false), decoding a conditional branch with an offset of zero is trivial: it amounts to nothing more than matching the opcodes 8900 or 8B00.

In the example in Figure 5, the first instruction compares two registers and sets the condition flag. Based on the condition of this flag, the second instruction may branch to the fourth instruction, an offset of zero according to the SuperH definition. If the condition is not true, the ADD instruction executes normally.

The first two instructions generate the condition flag and include the "magical" conditional-branch-with-zero-offset opcode. In second pair of instructions, the ADD is possibly treated as a NOP if the condition flag is not set.

Regardless of the outcome of the comparison, the entire construct takes two clock cycles to execute, effectively making the ADD into a predicated-execution instruction. In this instance, the branch does not disrupt the fetch stream or alter execution time. Although this construct will work for any pair of instructions, it's valuable only when the third and fourth instruction can be paired. If the fourth instruction were an ADD, for instance, it could not be executed in parallel with the ADD, negating any advantages.

First SH-4 Chip Already Planned

The first member of the SH-4 family will operate at up to 200 MHz and include separate caches: 8K for instructions and 16K for data. The data cache has 16-byte lines with a selectable write-back/write-through update policy. The caches are not lockable.

The part's MMU is nearly identical to that on the existing SH7702 and SH7708 chips, an obvious move that ensures Windows CE will run on the new chip with only minimal

tweaking. A 64-entry unified TLB is complemented with a four-entry micro-TLB for instructions, minimizing the disruptive effects of code fetches on data translation.

A set of peripherals rounds out the chip, including a four-channel DMA controller, two serial channels, three timers, a real-time clock, interrupt logic, and a programmable breakpoint unit. Also included is a DRAM interface that directly controls SGRAM, SDRAM, or EDO DRAM.

The system interface is described as compatible with the SH-3 parts, with two pinout options. In a 256-lead PQFP package, the chip will have a 64-bit data bus multiplexed with its 32-bit address bus. A less expensive 32-bit option fits in a smaller PQFP-208 package.

Aggressive Process Pushes Out Production

The SH-4 is still in the final stages of simulation; tapeout is expected at the very end of this year. Initial samples of the part will be fabricated in Hitachi's new 0.3-micron logic process. Those samples are expected to hit 167 MHz and will be available in 2Q97.

A relatively straightforward shrink will reduce the gate length to 0.25 microns for production. In that process, the die is expected to measure just 42 mm², nearly the same size as the company's own SH7708 in a 0.5-micron process (see [090302.PDF](#)). Full production of the 0.25-micron SH-4 is planned for 2Q98, a full 18 months after announcement. Hitachi expects the chip to reach 200 MHz at 2.5 V, where it will dissipate 1.8 W (maximum).

As with Digital's StrongArm (see [100201.PDF](#)), a special low-power version will be available. That chip should reach as high as 50 MHz from a tiny 1.0–1.2-volt supply; power dissipation is expected to be just 100 mW. If Hitachi can achieve this goal, the SH-4 will have roughly the same integer performance as the SA-110 with just one-third the power dissipation. That comparison is still more than a year away, however, during which time Digital is certain to make improvements to its process technology as well.

The Road Ahead with SuperH

Hitachi's SuperH architecture gained notoriety when Sega made the SH7604 famous. Although the 7604 is still regarded as "the Sega processor," that company currently accounts for less than half the unit volume of SuperH chips, as Hitachi has diversified its customer base.

Recently, sales of the Sega Saturn game player have begun to slump, suffering from stiff competition with Sony's PlayStation plus anticipation of the Nintendo 64, both MIPS-based machines. Perhaps because of this, Sega, too, has diversified, spinning off software development into a separate company that promptly declared its platform independence.

Price & Availability

The initial implementation of the SH-4 design will tape out in December. Samples are expected to be available in 2Q97, with production commencing 2Q98, after a 0.25-micron shrink. Pricing has not been disclosed.


For information, contact Hitachi (Brisbane, Calif.) at 415.244.7317 or e-mail pelavin_d@halsp.hitachi.com.

Neither company is affirming that Hitachi chips will appear in new Sega hardware. But the design of the SH-4 clearly points in that direction—perhaps not for Sega specifically, but certainly for graphics-enabled consumer items: maybe games, maybe set-top boxes, maybe high-end PDAs.

With Hewlett-Packard, Casio, and LG now lining up behind SuperH and Windows CE, the future looks bright for Hitachi's RISC chips. These three companies, and a handful of others, will roll out their new HPC (handheld PC) systems in November, which should cause an uptick in SuperH sales through 1997 at least.

Hitachi asks rhetorically, "How long ago would the SH-4 have been the fastest microprocessor in the world?" In terms of basic integer performance, at least a few years back. But for graphics setup and rendering applications, the SH-4 may well hold the title—on paper. With chips more than a year away from shipping, there's plenty of time for that situation to change. Better to ask, "How long from now before the SH-4 is overtaken by another microprocessor?"

Still, the direction Hitachi is taking with the SH-4 is similar to what most of the microprocessor industry is doing. Integer arithmetic, flow control, and Boolean logic were mastered long ago; power consumption, heat dissipation, and clock frequency are all governed by process technology. The only way for one architecture to set itself apart from the others is through intelligent and forward-looking extensions to the instruction set, and those extensions increasingly have nothing to do with the traditional values of embedded computing. Embedded systems have moved from counting, calculating, and computing to communicating and entertaining. A whole new set of architectures will be best suited to these new types of systems.

Paradoxically, the more exotic microprocessors become, the more invisible they become. As they find their way into more and more nontraditional applications, some of the most unusual microprocessors will find themselves in the most usual of homes. With its SH-4 and the rest of the SuperH family, Hitachi is well on the way to making these chips invisible, indeed. 



MICHAEL MUSTAGHI

Jim Slager of Hitachi describes SH-4's impressive FP capabilities at the Microprocessor Forum.