

M I C R O P R O C E S S O R

www.MPRonline.com

THE INSIDER'S GUIDE TO MICROPROCESSOR HARDWARE

BUSY BEES AT SILICON HIVE

New Processor Cores Target Pixel Processing and Communications

By Tom R. Halfhill {6/20/05-01}

Silicon Hive's mission is to replace DSPs and hard-wired application-specific logic with programmable processors based on its unbelievably long instruction word (ULIW) architecture. OK, we're exaggerating—ULIW actually stands for *ultralong* instruction word. But

with instruction words stretching as long as 918 bits, this architecture does seem almost unbelievable.

Nevertheless, *Microprocessor Report* believed Silicon Hive's technology was so innovative that we awarded it our MPR Analysts' Choice Award for Best Soft-IP Processor Core of 2003. (See MPR 2/9/04-18, "Avispa+ Buzzes With Innovation.") Our award was based on Silicon Hive's debut at Microprocessor Forum 2003, when the company introduced the first two ULIW processor cores—Avispa and Avispa+—intended for wireless-communications chips. (See MPR 12/1/03-02, "Silicon Hive Breaks Out.")

Last month, at **Spring Processor Forum 2005**, Silicon Hive followed up with two more ULIW processor cores, the Avispa-IM1 and Avispa-CH1. This time, the company is targeting pixel processing as well as wireless communications. Like all Silicon Hive processors introduced so far, the Avispa-IM1 and Avispa-CH1 are synthesizable cores offered as licensable intellectual property (IP).

In some ways, the new processors are simpler than Silicon Hive's first two cores. They have fewer function units and issue fewer instructions in parallel. However, both rely on new types of function units optimized for digital-signal processing and can deliver higher throughput for those tasks. In addition, the Avispa-IM1 can execute longer instruction words than any previous Silicon Hive core, and the Avispa-CH1 reaches a higher clock frequency than other Avispa cores. Their functional blocks and local memories are configurable at design time, so Silicon Hive can further

optimize the processors for specific applications and silicon budgets.

Overall, the Avispa-IM1 and Avispa-CH1 represent a logical evolution of ULIW technology. Both new cores are available now from Silicon Hive, a Netherlands-based subsidiary of Royal Philips Electronics. The Avispa-CH1 supersedes the earlier Avispa and Avispa+ cores for wireless applications.

Avispa-IM1 Crunches Pixels

The Avispa-IM1 is a pixel-processing engine that can replace a DSP core on an SoC or a discrete DSP in a less integrated design. It's intended for multifunction printers, digital still imaging, and digital video, and is optimized for such tasks as error diffusion, RGB-to-CMYK conversions, Bayer-to-RGB conversions, gamma correction, YUV color-space transformations, image cropping, image scaling, and JPEG encoding and decoding.

In a multifunction printer, Silicon Hive envisions the Avispa-IM1 serving as an image post-processing engine for converting color spaces and for error-diffusion dithering. In a digital camera, the Avispa-IM1 could perform various transformations on RAW data from the image sensor and compress the data in JPEG format.

All those tasks run best on a DSP or a processor with DSP extensions, so Silicon Hive has designed a new DSP block especially for the Avispa-IM1. (Actually, Silicon Hive has designed several new DSP blocks, and a different one is found in the Avispa-CH1, described later.) Silicon Hive refers to these and other functional blocks in the ULIW architecture as

processing and storage elements (PSE). Avispa processors have several PSEs for handling different tasks. PSEs are virtual processor cores in their own right, having their own function units, register files, ULIW issue slots, interfaces to the system, and, sometimes, their own data memories. They share program memory with other PSEs in the same Avispa core.

There are nine PSEs in the Avispa-IM1: one Control PSE and eight of the new DSP PSEs, which Silicon Hive refers to as a Generic DSP. The Control PSE executes branches, updates status registers, and transfers data to and from the FIFO buffers on the I/O bus. This PSE is a relatively simple block, and it merits only one issue slot in the Avispa-IM1's 918-bit-long instruction words. (The Avispa-IM1's program memory is organized as 1,024-bit words, but the actual instruction words are an odd-size 918 bits long.)

More interesting are the new Generic DSP PSEs, which are essentially fixed-point programmable DSP cores. The configurable datapaths can be 8, 16, or 32 bits wide, and they are 32 bits wide in the standard Avispa-IM1. Each Generic DSP has four function units: an ALU for general-purpose RISC operations; a barrel shifter and modulo add/subtract unit for bit-manipulation operations; a multiply-accumulate (MAC) unit for DSP operations; and a load/store unit that can automatically increment memory addresses. Supporting these function units are four 32-bit register files, totaling 34 registers. The MAC unit can execute a 16×16-bit multiply in one clock cycle or a 32-bit MAC in two cycles. The load/store unit, which also can fire on every clock cycle, connects to local data memory (configurable in size, up to 1GB) and a slave interface to the system. Figure 1 shows a block diagram of a Generic DSP PSE.

High Parallelism in Tight Loops

Each Generic DSP has three issue slots in the Avispa-IM1's lengthy instruction words. With eight Generic DSPs in the Avispa-IM1, a single instruction word can issue up

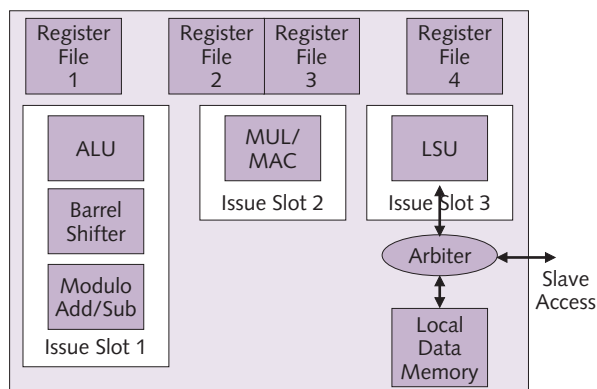


Figure 1. Silicon Hive's Avispa-IM1 processor core includes eight of the new Generic DSP processing and storage elements (PSE). Each programmable Generic DSP can execute up to three instructions per clock cycle, using four function units, providing roughly the same signal-processing capability as a 32-bit DSP core. The load/store unit accesses local data memory and a slave interface to a system bus or host CPU.

to 25 operations for parallel execution—24 DSP operations and one operation for the Control PSE. Therefore, under ideal conditions, the Avispa-IM1 can execute 25 operations per clock cycle, and 8 of those operations can be MACs. Peak performance is 1,200 million MACs per second (MMACS) at the processor's nominal clock frequency of 150MHz (assuming a 0.13-micron fabrication process, worst case).

No independent benchmarks are available for the Avispa-IM1. That's too bad, because we think EEMBC's new Digital Entertainment suite would be a particularly revealing test. (See *MPR 2/22/05-01*, "EEMBC Expands Benchmarks.") The DSP benchmarks by Berkeley Design Technology Inc. (BDTI) could also reveal useful insights. Silicon Hive says it is evaluating industry-standard benchmark suites for future performance testing. Meanwhile, the company offers some hypothetical performance examples.

According to Silicon Hive, JPEG baseline encoding requires about nine clock cycles per pixel, so the Avispa-IM1 could compress a five-megapixel image in about a third of a second, if memory can keep up. Trilinear interpolation requires about 12 cycles per pixel, and, on average, the processor issues 20.6 instructions in parallel when executing the inner loop. For post-filtering after a resolution conversion, the processor requires about 40 cycles per pixel and issues an average of 9.8 instructions in parallel during the inner loop.

One advantage of the Avispa-IM1 is worth noting: unlike conventional DSPs, it doesn't require intense assembly-language programming. Indeed, for practical purposes, Avispa processors aren't programmable in assembly language. Instead, Silicon Hive provides its own C compiler, *HiveCC*. It's a proprietary spatial compiler that creates a mathematical constraint model of the program's dataflow graph and matches it with a similar constraint model of the processor. Dr. Lex Augustejn, Silicon Hive's chief compiler architect, described *HiveCC* at Embedded Processor Forum 2004. To find links to more information about this tool, see the "Price & Availability" box.

Avispa-CH1 Optimized for Communications

The second new processor core, the Avispa-CH1, is designed for wireless communications standards based on orthogonal frequency division multiplexing (OFDM). In that respect, it's like the earlier Avispa and Avispa+ processors, which it replaces. It runs at a higher clock frequency—200MHz instead of 150MHz (0.13-micron CMOS, worst case)—but has shorter instruction words than the earlier processors. Whereas Avispa packed 42 operations into each 486-bit instruction word, and Avispa+ packed 60 operations into each 768-bit instruction word, the Avispa-CH1 has only 15 issue slots per 256-bit instruction word. Actual performance is scalable, because Silicon Hive will configure the core with different numbers of PSEs and different amounts of local memory, depending on the customer's needs.

Silicon Hive envisions the Avispa-CH1 as displacing DSPs or hard-wired logic in the inner receiver section of a

multistandard wireless receiver. Applications include digital audio broadcasting (DAB), terrestrial digital video broadcasting (DVB-T), handheld DVB (DVB-H), terrestrial digital multimedia broadcasting (DMB-T), and terrestrial integrated services digital broadcasting (ISDB-T). Higher-end configurations are suitable for 802.11 Wi-Fi and 802.16 WiMax receivers. The Avispa-CH1 can accelerate fast-Fourier transforms (FFT), inverse FFTs, and various algorithms used for channel estimation, equalization, Doppler compensation, data demodulation, and time/frequency synchronization.

For this processor, Silicon Hive designed an enhanced version of the Generic DSP PSE found in the Avispa-IM1, plus a new Complex DSP PSE for manipulating 32-bit complex numbers (16-bit reals + 16-bit imaginaries). The enhanced Generic DSP has a second ALU and supports four issue slots, instead of only one ALU and three issue slots like the Generic DSP in the Avispa-IM1. In contrast, the Complex DSP supports five issue slots, each with its own function unit and register file. Function units include an ALU/MAC unit, two load/store units for local data memory, a load/store unit for local lookup-table (LUT) memory, and an address-generation unit for the LUTs. There are 28 registers, all 32 bits wide. Figure 2 shows a block diagram of a Complex DSP PSE.

Silicon Hive will configure the Avispa-CH1 with one, two, or four Complex DSPs. One or two are sufficient for digital audio or video receivers; four are advisable for Wi-Fi and WiMax receivers. A typical configuration with two Complex DSPs, one enhanced Generic DSP, and one Control PSE would provide 15 issue slots within 256-bit instruction words and require about 250,000 gates, not including local memories. This typical configuration could execute two complex MACs per clock cycle, or 400 complex MMACS at 200MHz. One complex MAC consists of four real multiplies and five adds, so it's roughly equivalent to four real MACs, or 1,600 MMACS at 200MHz. The enhanced Generic DSP PSE can contribute another 200 MMACS at that clock speed.

Sizes of the 32-bit data memories and LUTs are design-time configurable; each can address up to 1GB. In real-world applications, few tasks would demand so much memory. Silicon Hive says a Complex DSP needs only about 256KB of local memory and 4KB of LUTs in an Avispa-CH1 processor configured for a DVB-T or DVB-H receiver. The more PSEs in the processor, the less memory each PSE requires.

The small LUTs supplement the local data memories. LUTs hold frequently used data—such as complex filter coefficients and FFT twiddle factors—and are paired with address-generation units to speed up the complicated addressing schemes associated with these tables.

According to Silicon Hive, an Avispa-CH1 with two Complex DSPs can execute two FFT butterflies per clock cycle, so an 8,192-point FFT on 2x16-bit complex data words would require about 3.5 cycles per sample. This is a typical requirement for demodulating OFDM signals. In a DVB-T application, this task would use

Price & Availability

Silicon Hive's new Avispa-IM1 and Avispa-CH1 processor cores are available for licensing now, along with the *HiveCC* compiler and other development tools. Both cores are available in synthesizable VHDL; Verilog will be available in 4Q05. Silicon Hive doesn't disclose licensing fees. For more information about the Avispa family, visit www.silicon-hive.com/t.php?assetname=text&id=23. For more information about *HiveCC*, visit www.silicon-hive.com/t.php?assetname=text&id=28.

about 15% of the processor's bandwidth. Offering another example, Silicon Hive says an equally configured Avispa-CH1 can execute four finite impulse response (FIR) taps per clock cycle, so an 11-tap FIR filter on 2x16-bit complex data would require about three cycles. In a DVB-T application, this task would use about 13.5% of the processor's bandwidth.

Avispa Processors Aren't Software Compatible

All Avispa processor cores are based on the same ULIW architecture, but, as Table 1 shows, their features vary widely (pun intended). While other companies have struggled to exploit the wide instruction-issue potential of VLIW, Silicon Hive has carried the concept to extremes. In Avispa processors, instruction words vary from 256 bits to 918 bits long, and their issue widths vary from 15 operations per clock cycle to 60. One consequence of these variations is that Avispa processors cannot run the same executables—customers must compile their programs for a specific implementation. This isn't a handicap in most embedded applications.

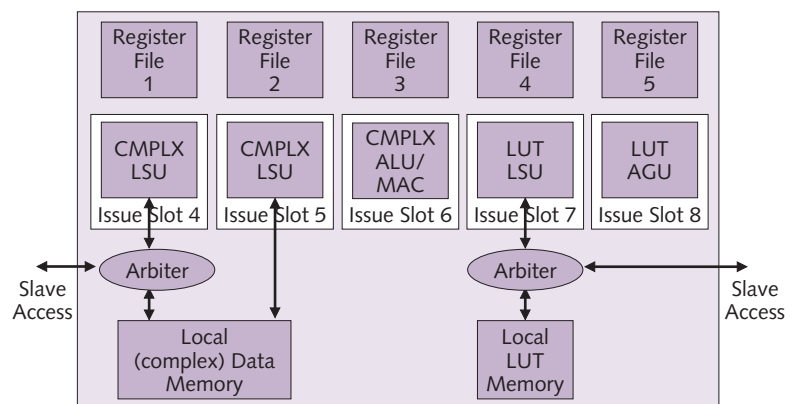


Figure 2. The Avispa-CH1 has a second new type of function block, known as a Complex DSP PSE. It's designed for crunching 32-bit complex numbers and can execute five operations per clock cycle if all five issue slots are filled with useful instructions. In addition to local 32-bit data memory, a Complex DSP has 32-bit local memory dedicated for lookup tables. The amount of memory and the number of PSEs are configurable at design time. A typical Avispa-CH1 configuration has two Complex DSPs, an enhanced version of the Generic DSP found in the Avispa-IM1, and a Control PSE.

Feature	Silicon Hive Avispa-IM1	Silicon Hive Avispa-CH1	Silicon Hive Avispa	Silicon Hive Avispa+
General Features				
Architecture	ULIW	ULIW	ULIW	ULIW
Application Domain	Pixel processing	OFDM radio	OFDM radio	OFDM radio
Instruction-Word Width	918 bits	256 bits	486 bits	768 bits
Issue Slots Per Word	25 operations	15 operations	42 operations	60 operations
Program Memory*	Up to 4.1Mwords x 1,024 bits	Up to 134Mwords x 256 bits	512 words x 512 bits	512 words x 768 bits
Arithmetic Processing & Storage Elements				
Arithmetic PSEs	—	—	4	4
Dual-Port Mini-Caches	—	—	—	4
Local Data Memory*	—	—	4x4K dual port	4x4K single port
Viterbi Registers	—	—	—	4 x 48 bits
Control and I/O Processing & Storage Elements				
Control & I/O PSEs	1	1	1	1
Issue Slots Per PSE	1	1	1	1
Generic DSP Processing & Storage Elements				
Generic DSP PSEs	8	1	—	—
Datapath Width*	32 bits	32 bits	—	—
Issue Slots Per PSE	3	4	—	—
Data Memory Per PSE*	Up to 1GB x 32 bits	Up to 1GB x 32 bits	—	—
Complex DSP Processing & Storage Elements				
Complex DSP PSEs	—	2	—	—
Issue Slots Per PSE	—	5	—	—
Data Memory Per PSE*	—	Up to 1GB x 32 bits	—	—
LUT Memory Per PSE*	—	Up to 1GB x 32 bits	—	—
Other Features				
Function Units (Total)	35	18	75	103
Register Files (Total)	33	15	95	130
Clock Freq (0.13 μ m)	150MHz	200MHz	150MHz	150MHz
Core Size (0.13 μ m)	250K gates	250K gates	115K gates	140K gates
Power (150MHz) †	~105mW	~100mW	~127.5mW	~150mW
Availability	Now	Now	Superseded	Superseded

Table 1. Silicon Hive's Avispa family is based on a common ULIW architecture, but one member can't run another's software without recompilation, due to the extreme variations of their implementations. No other CPU architecture offers so much flexibility when defining instruction formats and issue widths. Even the narrowest processor in this family can pack up to 15 operations into a single instruction word, and the widest implementation (so far) can issue 60 operations per clock cycle. Note that the communications-oriented Avispa-CH1 has superseded the Avispa and Avispa+ processors. *Configurable feature. †0.13-micron CMOS, worst-case military specifications.

Untold is how well these extreme processors will compete with conventional DSPs, licensable DSP cores, and licensable RISC processors with DSP extensions. Silicon Hive's performance examples are illustrative, but independent benchmark tests would be more illuminating. Among other things, independent testing could reveal how efficiently the *HiveCC* compiler handles common signal-processing code written in C. Most programmers are accustomed to writing hand-tuned assembly language to wring full performance from a DSP. If *HiveCC* can deliver similar results from unmodified ANSI C, as Silicon Hive claims, programmers

would rejoice. For best results, Silicon Hive acknowledges that programmers should expose more instruction-level parallelism in the source code by using the special intrinsic functions and pragmas the company provides.

Although Silicon Hive is a licensable-IP vendor of configurable processors, it operates a little differently than other configurable-processor vendors, such as ARC International, MIPS Technologies, and Tensilica. All the competitors put the processor-configuration tools into the hands of customers. Tensilica has some proprietary back-end tools for transforming the customer's configuration template into synthesizable Verilog or VHDL, whereas ARC and MIPS don't require an intermediate step between configuration and synthesis. Silicon Hive prefers to configure the processor in house and deliver a synthesizable model to the customer's specifications. Silicon Hive also offers preconfigured cores, as do ARC and MIPS.

We believe the modus operandi of ARC, MIPS, and Tensilica encourages more experimentation, because developers can rapidly generate and test successive iterations of a processor to find the optimal configuration for a particular application. In addition, ARC, MIPS, and Tensilica allow developers to create their own processor extensions. Silicon Hive doesn't offer the same flexibility.

Silicon Hive's Avispa family will appeal to adventurous customers with specific signal-processing tasks that can benefit from the extreme parallelism of this ULIW architecture. It's truly unique and demands more due diligence than an easily understood DSP or RISC processor. Even among high-end DSPs, which are famous for their embrace of VLIW and other unusual concepts, the Avispa family stands apart. Frankly, it's a tough sell. Silicon Hive must work closely with potential customers to convince them that ULIW isn't as unbelievable as it first appears to be. ♦

To subscribe to Microprocessor Report, phone 480.483.4441 or visit www.MDRonline.com