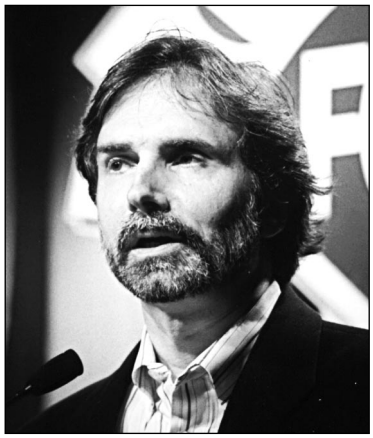
MicroUnity Lifts Veil on MediaProcessor

New Architecture Designed for Broadband Communications

by Michael Slater

Startup companies based on original microprocessor architectures don't come along very often. Even rarer is an architecture that breaks new ground in instruction-set design, or an implementation that is so much faster than previous chips that it makes possible entirely new applications. And startups that build their own fabs are virtually unheard of. Yet MicroUnity Systems Engineering, a secretive and much-watched startup, has laid out its plans to do all this and more.

MicroUnity, which now has nearly 200 employees, has been the subject of frequent speculation since its founding in 1989 by MIPS cofounder and former IBM researcher John Moussouris. At the recent Microprocessor Forum, Moussouris finally described the MediaProcessor architecture and provided an overview of its first implementation—the result of more than 500 man-years of effort. When fabricated in the company's proprietary BiCMOS process, the MediaProcessor is expected to run at clock frequencies up to 1 GHz—three times greater than the fastest microprocessor today.



MICHAEL MUSTACCHI

John Moussouris, CEO of MicroUnity, describes the unusual design of the MediaProcessor.

The company's funding partners—which have backed the company with more than $100 million so far—still have not been identified (see sidebar "The People Behind MicroUnity" at end of article). Indeed, MicroUnity's device appears to be at least a year away from volume production, with years more before it can reach its potential. The company already has an income stream from technology licensing but expects to require additional equity investments to help fund the production ramp.

This investment is justified, MicroUnity hopes, by the large future market for communications systems. MicroUnity's goal is that MediaProcessors will someday be ubiquitous throughout communications networks, serving as the universal processing engine both at central "head-end" facilities and in end-user equipment.

The First Broadband Microprocessor

All of MicroUnity's efforts stem from two central premises: that broadband digital communications will be widespread, and that a very fast, yet general-purpose, microprocessor will be the central building block.

The MediaProcessor is intended to deliver not just a few times the performance of today's high-end microprocessors, but hundreds of times their performance—at least on the targeted signal-processing applications. But this is much more than just another supercharged DSP: the MediaProcessor is a general-purpose device that also has exceptional signal-processing capabilities.

This performance level is necessary for the MediaProcessor to handle the data rates needed to implement broadband modulation and demodulation. By implementing techniques like quadrature amplitude modulation (QAM) in software, the device is able to exist in a wide range of different environments and deal adaptively with changing signal and media conditions.

The MediaProcessor can also implement the multimedia functions targeted by conventional multimedia processors, including video compression and decompression, audio processing, and 3D rendering. Compute-intensive DSP tasks such as echo cancellation can be handled as well, along with packet protocols for communications links, cryptography, and routing. It can also run general-purpose applications and operating systems; a 64-bit version of OSF Unix has been ported. This enables the chip to serve as the CPU in head-end servers.

MicroUnity believes that by combining all these functions in a single, programmable processor, the cost of systems ranging from cable modems and interactive set-top boxes to cellular base stations and head-end switch servers can be significantly reduced. Instead of requiring an assortment of special-purpose ASICs, DSPs, and microcontrollers, each with its own private memory, all functions are performed by a single device.

The result is a dramatic reduction in chip count, and potentially in cost. Equally important is the system's agility; unlike a typical ASIC implementation, in which specific algorithms are hard-wired in the chip designs, everything in a MediaProcessor-based system is "soft." In a world where standards are still rapidly evolving, this could be a significant advantage; ASIC-based solutions will become more integrated, but it will be hard for them to match the agility of the MediaProcessor. The MediaProcessor aims to bring to communication devices the same software-based approach that micro-

Storage and Synchronization
load (unsigned) 8/16/32/64/128 bits little/big-end (aligned) (immediate) store 8/16/32/64/128 bits little/big-end (aligned) (immediate) add/compare/multiplex (and swap) 64-bit (immediate)
Branch
branch and-equal/and-not-equal/less/less-equal zero branch equal/not-equal/less/greater-equal branch floating-point equal/not-equal/less/greater-equal 16/32/64/128 branch (immediate) (and link) branch gateway (immediate) branch down/back
Fixed-Point
Data types: 64-bit scalar group 128×1/64×2/32×4/16×8/8×16/4×32/2×64/1×128
add/sub (immediate) (overflow) (unsigned) multiply (and add) (unsigned) divide
and/or/and-not/or-not/xor/xnor/nor/nand (immediate) multiplex (unsigned) set/sub equal/not-equal/less/greater-equal (immediate)
(unsigned) shift/rotate right/left (immediate) (overflow) (unsigned) compress/expand/extract (immediate) select bytes shuffle/deal/swizzle (unsigned/merge) deposit/withdraw immediate (shuffle) 4/8-way multiplex
and sum of bits log most significant bit Galois-field multiply/polynomial multiply-divide 8/64-bits
Floating-Point
Data types: half/single/double/quad precision group 8×16/4×32/2×64
add/sub/mul/div (near/truncate/floor/ceiling/exact) abs/neg/sqr/sink/float/inflate/deflate (near/truncate/floor/ceiling/exact) set equal/not-equal/less/greater-equal

Table 1. The MediaProcessor instruction set is unusual in that most instructions can function in group as well as scalar modes. "()" indicates optional features; "/" separates alternatives.

processors brought to control systems two decades ago.

The agility may have significant advantages in adapting to varying communications environments. Cable modems, for example, have proved to be very difficult to get working reliably and have to be tuned to each cable system; a MediaProcessor-based cable modem could adapt to the communications channel.

Architecture Blends RISC and DSP

The MediaProcessor instruction-set architecture is a 64/128-bit RISC-style design with several unique features. At the heart of the architecture is the assumption that software will operate on data items of a wide range of sizes. Unlike conventional architectures, which waste much of the available memory and register bandwidth when using operands of less than the maximum width, the MediaProcessor architecture aims to make full use of the bandwidth, regardless of the data size, by supporting parallel subfield operations. Similar operations are being added to other architectures for multimedia support, but no other CPU is likely to offer as extensive a set of functions or data types as the MediaProcessor.

While some of the operations performed by MediaProcessor instructions are more complex than those in conventional RISC designs, the overall structure is simplified even further. The register file consists of 64 registers of 64 bits each, which can also be accessed as 32 register pairs of 128 bits each. Unlike conventional architectures, there is no separate floating-point register file; all data types use the single unified file.

Furthermore, there are no special registers as part of the user state; the register file and the 64-bit program counter are the entire user state. Condition codes are stored in general registers. Other controls that usually are implemented in special registers, such as floating-point rounding or byte-ordering selectors, are included in the opcodes. These characteristics are thus specified on an instruction-by-instruction basis, eliminating them from the machine state. This simplification of the user state facilitates deeply pipelined designs and the implementation of precise exceptions.

Regular Instruction Encodings

All instructions are encoded in one of a few 32-bit formats. The basic format is an 8-bit opcode and four 6-bit operand fields. For instructions requiring fewer operands, one or more of the fields can be used for immediate data or for opcode extensions. The four-operand format is unusual and allows more complex instructions to be implemented in a single cycle. One price is an additional read port on the register file.

Table 1 summarizes the instruction set. Many of the instructions are typical of any processor, though they have some unique twists. Other instructions are quite different from those in any conventional processor.

Fixed-point data types of 1, 2, 4, 8, 16, 32, 64, and 128 bits are supported, along with four floating-point resolutions: IEEE-standard single- and double-precision, plus 16-bit "half-precision" and 128-bit quad-precision formats. Loads and stores for all data types are provided, each with little-or big-endian data. Unaligned accesses are allowed. Addressing modes are base register plus a 12-bit offset or base register plus index register.

A variety of atomic synchronization instructions support coordination of tasks within one processor or across multiple processors. These include add and swap, compare and swap, and multiplex and swap.

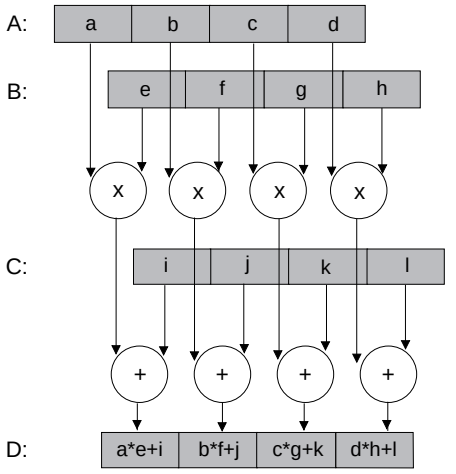


Figure 1. Group floating-point multiply-add operating on four sets of single-precision floating-point numbers.

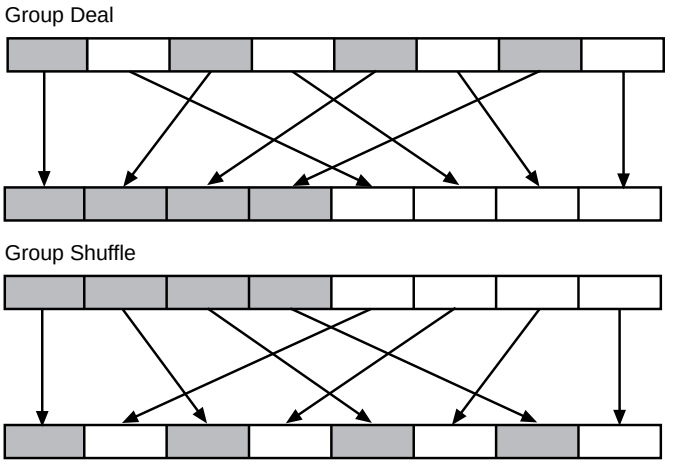


Figure 2. Group deal and group shuffle instructions, shown here operating on 16-bit data.

Branches are not delayed. Both fixed- and floating-point compare-and-branch instructions are provided. The special "gateway branch" changes the privilege level and also loads two 64-bit values from memory: one to reset the program counter to effect the branch, and another to load into a register for use as a data pointer.

In addition to all the usual arithmetic and logical operations, several more unusual operations are provided. These include:

- Find most/least significant "1" in word.
- Count each "1" in word (population count).
- Multiplex. On a bitwise basis, operand register A selects whether the bit from register B or register C is written to the destination register.

Several special arithmetic instructions support extended math functions for signal processing. Only one of these instructions has been disclosed; it implements Galois field arithmetic, which is useful for techniques such as Reed-Solomon error correction.

The architecture supports 64-bit virtual addresses with arbitrary page sizes. It provides four privilege levels in the TLB and 16-bit address-space identifiers as part of the virtual address. All addresses are 64 bits.

Group Ops Support DSP Algorithms

The signal-processing orientation of the instruction set is most evident in the group operations, which are an optional form of most fixed- and floating-point instructions. These instructions are similar to the multimedia extensions added to some RISCs, such as in Sun's UltraSparc, in that they operate on the full register width but subdivide it into multiple data items.

The MediaProcessor's group instructions go beyond those implemented in other processors in that they support floating-point as well as integer instructions and a wider range of data sizes and functions. These instructions operate on 128-bit register pairs divided into fixed-point sizes of 1, 2, 4, 8, 16, 32, or 64 bits, or 16-, 32-, or 64-bit floating-point values. For example, a single instruction could operate on 32 four-bit values, 16 eight-bit values, or two double-precision floating-point values.

Group operations include multiply, add/subtract, and shift/rotate. Figure 1 shows the most complex group operation: multiply and add. This is a four-operand instruction, operating on 384 bits of input data and delivering 128 bits of result. The figure shows this operation on single-precision floating-point data, allowing four sets of calculations to be performed in parallel. For the integer version of this instruction, only 64-bits of registers A and B are used, so the result can carry twice the number of bits as the input data.

Another set of group instructions, called switching operations, performs complex rearrangements of the data within a register pair. Figure 2 shows two examples: group deal and group shuffle. The switching logic in the processor is able to perform any random rearrangement of the bits in a word, but it takes several instructions to provide enough data to specify an arbitrary remapping. Several common switch operations have been encoded in single instructions:

- Compress, which extracts any contiguous half-size field from each subfield (e.g., 16 bits from each of four 32-bit words) and combines them in a 64-bit word.
- Extract, which does the same as compress but starts with 256 bits of data and delivers 128.
- Expand, which does the opposite of compress.
- Swizzle (copy-swap), which reverses the order of the bits in each subfield within a register pair.

Extensive Software Support

With a new architecture, the burden of software support is great, and MicroUnity has invested heavily in this area as well as in the silicon itself. A number of simulators have been used to support the creation and

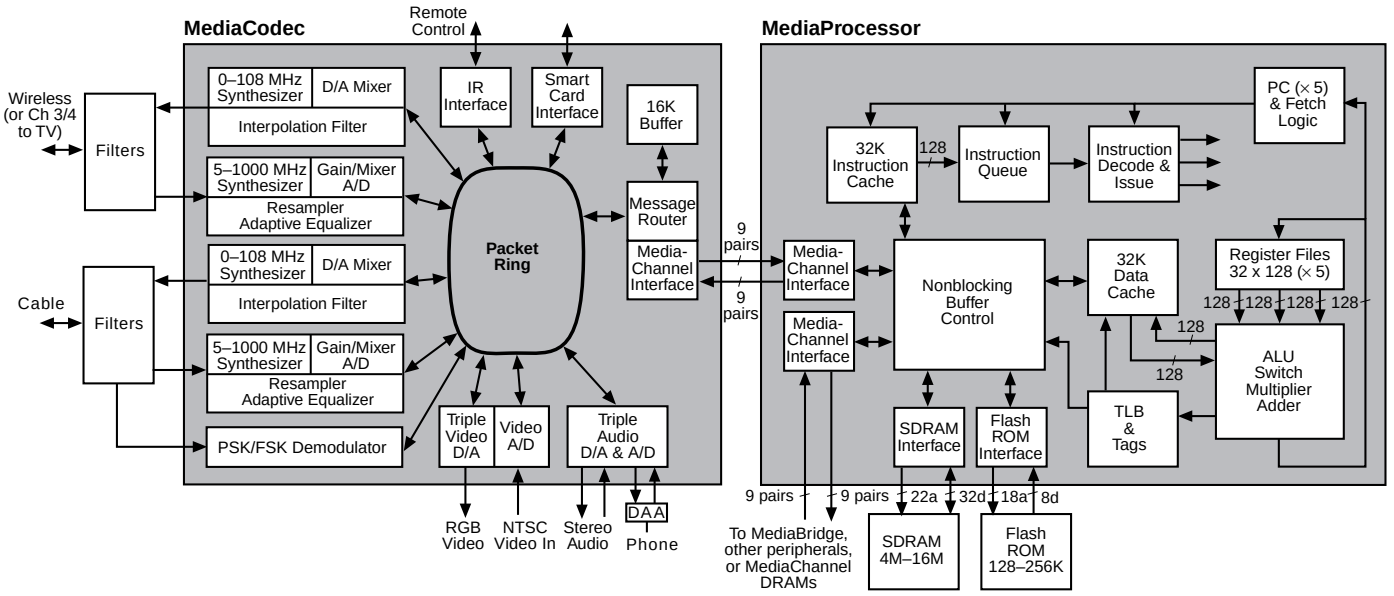


Figure 3. The MediaProcessor provides a direct connection to SDRAM and flash ROM, while using the MediaChannel to connect to the MediaCodec and MediaBridge (not shown). The MediaCodec implements two agile digital radios and other peripheral interfaces.

debugging of a wide range of software, even though no hardware is yet running.

The architecture is supported by C and C++ compilers, as well as a range of simulators, a full 64-bit Unix implementation (for servers), and a real-time kernel (for client devices). Libraries include algorithms for audio (MPEG, Dolby AC-3), video (MPEG-1, MPEG-2, and NTSC encode and decode), communications (QAM and QPSK modems, Reed-Solomon error correction, DES encryption, and broadband MAC protocols), and network protocols (Ethernet, MPEG control, and ISDN).

Chip Set Implements MediaComputer

MicroUnity has completed the design of its first MediaProcessor and two support chips. Figure 3 shows the block diagram for a MediaComputer using the processor and the MediaCodec. A second support chip, not shown here, provides a bridge from the MediaChannel interface to conventional DRAM and PCI bus devices.

The design of all three chips has been completed for many months, but initial test wafers turned up difficulties with delamination of thin insulator strips between wide metal traces in the company's unique process (see "Why Build a Fab" sidebar below). Design-rule changes to remedy the problems have been made, and a revised layout of the processor is now in fabrication. The first functional samples are expected by year-end. Revised layouts for the MediaBridge and MediaCodec will be created after the process and design-rule refinements are proved using the MediaProcessor.

Two versions of the MediaProcessor will be built. One, designed to run at up to a 1-GHz clock rate, will be built in MicroUnity's fab using the company's proprietary BiCMOS process. This device was designed to meet the performance needs of a full-function broadband MediaComputer, of which a digital set-top box is a subset. A second design, logically identical but implemented in a 0.5-micron CMOS process and built by an outside foundry, has a target clock rate of 300 MHz. This chip is designed to meet the performance needs of an agile, broadband digital radio, such as a high-speed cable modem or a cellular base station. Tape-out of the CMOS version is scheduled for this month.

MicroUnity's BiCMOS process provides exceptional density. With 10 million transistors on a 100-mm^2 die (see Figure 4), the MediaProcessor is about one-third the die size of other leading-edge processors that have fewer transistors and are built in 0.5-micron CMOS processes. The tight 1.0-micron metal pitch on the first four interconnect layers is responsible for the density advantage.

The BiCMOS version is projected to dissipate a toasty 85 W at 1 GHz. While this is far more than any commercial microprocessor, MicroUnity claims that this is less than the total power consumed by all the devices it can replace in applications such as a high-quality videoconferencing system. To be sure, it focuses this heat in a smaller area; an impressive heat sink and good airflow are required. For less demanding applications, the clock rate can be reduced; at 200 MHz, power dissipation is a more modest 14 W.

The CMOS version is expected to dissipate 40 W at 300 MHz. (Both the CMOS and BiCMOS designs use a 3.3-V power supply.) The high power consumption is due, in part, to the fact that the logic design and microarchitecture were designed for bipolar circuits and ported directly to CMOS. Three-fourths of the power is

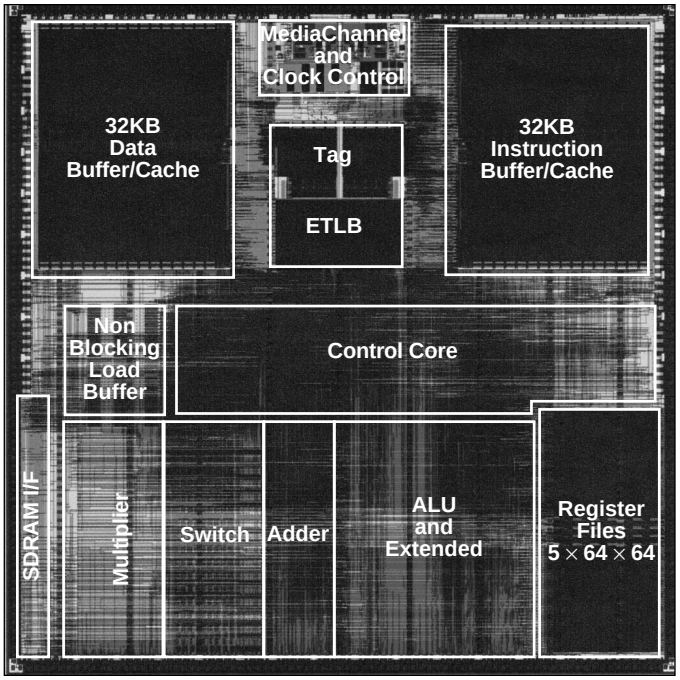


Figure 4. Die plot for the MediaProcessor chip, which integrates 10 million transistors on a die that measures 10 × 10 mm using MicroUnity's 0.5-micron five-layer-metal BiCMOS process.

dissipated in the clock drivers; by putting these drivers off chip, MicroUnity expects to cut power dissipation to 10 W. Future versions are planned to reduce this power consumption even further through a combination of more power-optimized designs and lower supply voltages. The CMOS design has a few more transistors, for a total of 10.5 million, and the die size is 290 mm^2 using a three-layer-metal process with a 2.0-micron metal pitch.

Each of the other chips uses the same 100-mm^2 die size in the BiCMOS process. This size is one-fourth of the reticle used in MicroUnity's steppers, which is a sweet spot in production capacity. All three chips are packaged in identical 312-lead TAB packages. The processor has 197 signal leads and 115 power and ground connections. MicroUnity uses a silicon interposer between the chip and the TAB frame to spread the leads and provide thermal-expansion matching.

Superpipelined and Superthreaded

Designing a microprocessor to operate at 1 GHz presents some formidable challenges, and the resulting microarchitecture is quite different from today's processors. In addition to a very deep pipeline, MicroUnity's proprietary process, low-voltage-swing on-chip signals, and special circuit and logic design techniques all play a role in achieving the high clock rate. The design is not superscalar, eliminating the complexity of multiple-instruction dispatch, dependency checking, out-of-order execution, and other techniques that are nearly universal in today's high-end microprocessors. Like conventional processors, the MediaProcessor has a pipelined

Why Build a Fab?

For a startup to build its own fab is almost unheard of today. Why did MicroUnity do this? The company wanted to build high-performance, very high speed mixed-signal chips, and the founders believed that no existing process met this need—and they had some innovative ideas that would create a faster, denser process than others had created. The company has already been granted more than two dozen patents, covering the process as well as the architecture and implementation techniques, and many more are pending.

The crunch in foundry availability also played a role. The company's first implementation was designed for a foundry, but the owner of that fab decided to get out of the foundry business. Other foundries were asking for up-front payments of tens of millions of dollars to provide access to capacity. MicroUnity built a small fab for not much more, and in the process created some valuable intellectual property that it can license to others.

MicroUnity developed a 0.5-micron BiCMOS process with very high density. The effective gate length is 0.3 microns. MicroUnity has designed its process and its fab to scale to 0.35-micron drawn dimensions.

The process uses five metal layers. The first four all have a tight 1.0-micron metal pitch, which provides greater connection density than any production process today. (Intel comes close, but others are further behind.) Instead of conventional aluminum interconnect, MicroUnity uses a gold alloy. Another exotic feature is that metal layers 3 and 4 can optionally be air-bridged; the insulating material below the trace is etched away, leaving the conductor floating in air. This reduces capacitance, power consumption, and crosstalk noise.

The process is extremely planar (flat within 0.1 micron at every layer). In the metal layers, this is done with proprietary lift-off techniques; unlike conventional processes, no polishing is used because current polishing technology doesn't work with gold interconnect.

The CMOS structures are used for memory cells, while the bipolar circuits are used for logic and analog functions. Logic circuits use a MicroUnity-developed constant-current variable-bias ECL design.

The fab has 9,000 ft^2 of clean-room space and is designed to produce 5,000 6" wafers per month when fully equipped. Though the fab is small by today's standards, MicroUnity believes that its process's high density will give it greater chip volume than larger fabs. So far, the company has spent less than $50 million to build the fab and install the equipment to deliver about one-third of its full capacity. When the fab is fully outfitted, the company expects the total cost to be about $100 million.

MicroUnity's chip designs use a sea-of-gates ASIC design approach with automatic routing and timing optimization, minimizing design time and making the designs easily adaptable. Custom macros have been created for memories and analog blocks.

Price & Availability

Development systems based on the MediaProcessor are expected to be available in the first half of 1996, along with samples of the processor chip in both CMOS and BiCMOS versions. Volume production is planned for late 1996. Samples of the MediaBridge and MediaCodec are expected in the first half of 1996, with production in early 1997. Pricing has not yet been set.

For further information, contact Tony Stelliga, VP Product Marketing, MicroUnity Systems Engineering (Sunnyvale, Calif.), at 408.734.8100; fax 408.734.8141; e-mail *tony@microunity.com*.

execution unit, instruction and data caches of 32K each, and a paged memory-management unit.

A look inside the execution unit, as Figure 5 illustrates, reveals a level of pipelining far deeper than any other microprocessor. Just the execution and data access pipeline—not counting instruction fetch, register read, and write-back—is 15 stages deep. Each pipeline stage consists of a single (but up to 16 × 16 wide) and-or-invert gate, followed by a latch.

The logic blocks within the pipeline are similar to those in conventional processors, though the ALU is more complex, to handle the group and extended math (e.g., Galois field) instructions. The ALU takes three 128-bit inputs and produces a 128-bit result. There is an additional switch block that rearranges data for the group switching instructions. The multiplier is followed by an adder to support multiply-add instructions.

The initial MediaProcessor does not implement the floating-point aspects of the architecture. The company took the 100-mm^2 die size as a limit; within this constraint, implementing the FPU would have meant a substantial reduction in the cache size. A floating-point version of the processor has been designed, and future implementations will include FP capability. The FP instructions will simplify some signal-processing algorithms and improve their performance, because they eliminate the need for scaling.

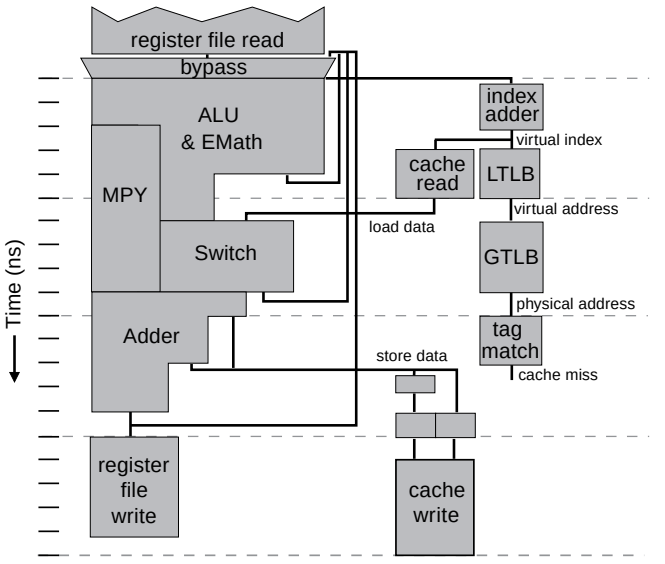


Figure 5. The MediaProcessor is superpipelined to support the high clock rate. Each tick on the left shows one pipeline stage, or 1 ns at the 1-GHz clock rate. The dashed lines every five stages show the major cycle, which is the clock rate seen by each of the five threads.

Superthreading Reduces Latency

Design at the 1-GHz target frequency has some special challenges. Moving a signal through a wire just a third of the way across the chip takes close to one clock cycle (1 ns), so some pipeline stages have nothing in them but a wire and a latch.

In a conventional design, such a deep pipeline would result in long latencies, sapping performance. MicroUnity overcame this problem by designing the chip to support five concurrent execution threads, a technique they call superthreading. There are five copies of the register file and program counter. Since there are no special registers, this provides five complete sets of user state. Each set is accessed on successive clock cycles; during a five-clock period, each set has one clock cycle of access to the pipeline, caches, memory, and I/O system.

In effect, the processor runs five separate threads in parallel, with each thread having the performance of a 200-MHz processor. The major benefit of this approach is that the effective latency (as seen by an individual thread) of the arithmetic units, register file, and memory system is divided by five. Only a single load-delay slot—which can frequently be hidden by the compiler—is needed, just as in many conventional RISCs.

One cycle of the 1-GHz clock is called a minor cycle, while five cycles—one for each thread—is called a major cycle. In terms of the latency seen by each thread, only the major cycles count. Simple ALU operations have a one-cycle latency; switch and load operations have a two-cycle latency; and multiply and multiply-add operations have a three-cycle latency.

The register file has three read ports and one write port. It is accessed in a single clock, supporting the maximum bandwidth of four operands per instruction

The processor implements only a simple branch prediction algorithm. Five branch target buffers store the target address of the most recent backward branch for each thread, so after the first pass through a loop instruction, fetching can follow the loop without delay. The penalty for a mispredicted or unpredicted branch is three cycles. One reason the MediaProcessor architects decided not to devote silicon area to full branch prediction is that the instruction set includes special features, such as the multiplex instruction, that eliminate many branches.

Large Caches Backed by SDRAM

The on-chip instruction and data caches, 32K each, are each able to deliver 128 bits every two clock cycles. As a result, the instruction cache actually has twice the bandwidth required to keep the pipeline fed. During a 10-cycle interval (two major cycles), each thread gets one double-word access to each cache. The caches are direct-mapped, virtually indexed, and virtually tagged.

The on-chip caches can also be used as a software-managed buffer memory. All, three-fourths, or one-half of each cache can be allocated as buffer memory, with the remainder operating as cache.

The MediaProcessor includes a 32-bit-wide SDRAM interface that supports up to 16M of memory and a dedicated byte-wide interface for up to 512K of flash memory to store firmware. A nonblocking buffer in the memory interface allows up to 16 pending load or store operations, and threads are not blocked until they need the result of a load operation or the buffer is full.

Careful design of the threads minimizes contention for the memory. For example, in a set-top box application, one thread handles radio demodulation and a second handles audio processing; the data sets for these threads fit in the on-chip memory. MPEG-2 decoding is divided among three threads, which coordinate their processing so their memory accesses do not conflict.

Systems needing more than 16M of DRAM must use a MediaBridge chip, which provides an interface to up to 128M of DRAM, as well as a PCI bus for other peripherals. Each MediaBridge also includes a 128K buffer that can be software-controlled or can operate as a second-level cache. Up to four MediaBridge chips can be connected to a single MediaProcessor.

MediaChannel Provides 1-Gbyte/s Path

To provide a low-cost yet high-speed connection to peripheral chips, MicroUnity took an approach similar to that used by Rambus: a very fast but only 8-bit-wide connection called the MediaChannel.

As Figure 6 shows, the MediaChannel supports one master and up to four slave devices. By using a point-to-point, unidirectional connection, a simple protocol, and special timing circuits, very high clock rates are achieved: the MediaChannel runs at the same clock rate as the processor (1 GHz for the BiCMOS implementation). Despite this high clock rate, relatively long traces—and even high-quality ribbon cable—can be used because of the point-to-point design and the differential, low-voltage-swing (100 to 700 mV) signals.

The clock is carried as a ninth signal, using identical circuits to the data. Programmable skew calibrators in each device, which can adjust the delay with a resolution of tens of picoseconds, compensate for variations in delays of the chip and connecting wires. This adjustment

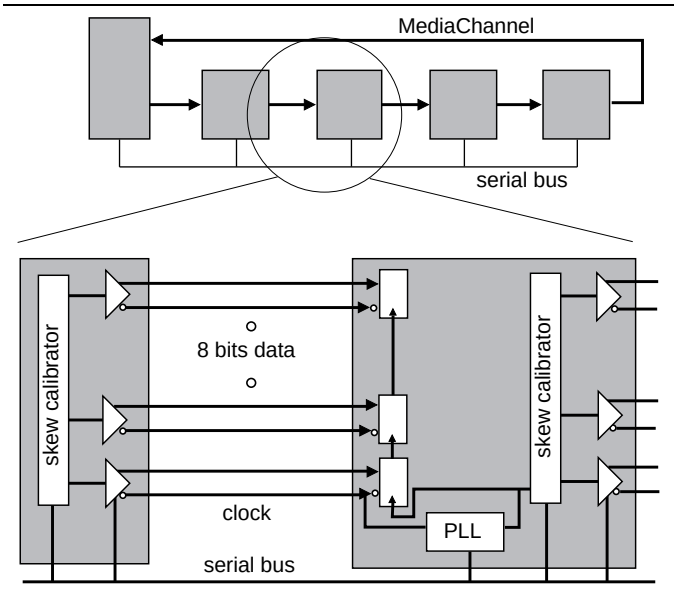


Figure 6. The MediaChannel interconnect uses a ring with one master and up to four slave devices. With a 1-GHz clock rate, it delivers a peak bandwidth of 1 Gbyte/s.

process and other diagnostics are handled by a simple, low-speed serial bus that connects all the devices.

All MediaChannel packets are fixed length, and the protocol is streamlined to support high data rates. Read transactions begin with a six-byte request packet, which includes a header byte, a four-byte address, and a check byte. The header byte identifies the type of packet as a read or write request (with or without allocate, for slave devices with caches), read or write response, idle, or error. A read response packet consists of a header byte, eight bytes of data, and a check byte.

Write transactions are initiated by a packet containing a four-byte address and eight bytes of data, framed by header and check bytes. A write response is just a header and check byte.

The MediaProcessor provides two separate MediaChannel interfaces. Either can be used for one or more MediaCodec or MediaBridge chips. MicroUnity is working with DRAM manufacturers to develop MediaChannel DRAMs, allowing the channel to be used directly for memory expansion.

MediaCodec Implements Digital Radios

Although the MediaProcessor provides the computational speed needed to process broadband signals, it is the MediaCodec that brings these signals into the digital domain. With about 10 million transistors, the MediaCodec rivals the complexity of the MediaProcessor. As a very high speed mixed-signal design, in which noise and other problems can wreak havoc, it is the most challenging of the chips to bring from concept to production.

As Figure 3 shows, the MediaCodec includes two bidirectional RF interfaces. In a set-top box, one would

The People Behind MicroUnity

John Moussouris, chairman and CEO of MicroUnity, was a cofounder and VP of R&D at MIPS Computer Systems. Before MIPS, he was a founding member of an IBM team that developed a VLSI implementation of the landmark "801" RISC processor.

The company's chief technologist, and the man behind its unique process technology, is Al Matthews. A veteran of many high-end process designs, he has worked on bipolar, CMOS, and GaAs process development at Performance Semiconductor, Avantek, Gould, Hewlett-Packard, and Intel.

The chief architect of the MediaProcessor is Craig Hansen, who managed the architecture of the R2000 and R3000 at MIPS. He also developed floating-point architectures for Weitek and Hewlett-Packard.

An important force behind the scenes is William Randolf Hearst III, former publisher of the *San Francisco Examiner* and currently a partner at Kleiner Perkins Caufield & Byers, as well as CEO of @Home, Inc.

Other board members include Lois Abraham, a senior partner at Brown & Bain and MicroUnity's general counsel; John L. Doyle, retired executive VP of Hewlett-Packard; and Bruce Ravenel, one of the original 8086 architects and now senior VP and COO of TCI Technology Ventures.

HP and TCI (the dominant operator of cable television systems in the U.S.) are both reportedly major investors in the company, as is Will Hearst. Other investors reportedly include Microsoft and Motorola. Motorola is rumored to be interested in the process technology, as well as the applications to its communications business.

connect to the cable, and the other would drive a wireless cell or an RF-modulated signal to the television. This chip is a superset of all the capabilities requested by all of MicroUnity's partners, and any individual application could use a simpler version if the company were to develop such versions. MicroUnity is designing a CMOS MediaCodec that uses external analog circuitry for the highest-speed functions; this chip will be more cost-effective for applications like cable modems.

The two RF interfaces are each capable of tuning a channel of up to 8 MHz from a center frequency of 5 to 1,000 MHz. Combined with processing in the MediaProcessor, a digital bandwidth of up to 40 Mbps can be extracted from each channel (using schemes such as 64–256QAM).

An on-chip packet ring connects the modules within the MediaCodec. The chip appears to the MediaChannel as a 16K block of memory, plus additional address space for memory-mapped control registers. The on-chip message router moves data between the dual-ported memory and the I/O channels.

A Long Road to High Volume

Although the initial MediaProcessor chip set will have some commercial applications, it is most significant as a proof of concept and as a development vehicle. Development systems, and even some applications, will be able to get by without the MediaCodec, which won't be available as soon as the processor.

The initial implementation is designed to deliver all the capabilities needed for a range of applications; future designs promise to offer higher integration and much lower power. The company is now working on a superscalar MediaProcessor designed to be implemented in CMOS. The superthreaded approach of the initial design was optimized for BiCMOS technology; for a CMOS implementation, a superscalar design running at a lower clock rate will be more power efficient.

Another factor in reducing the power consumption of future designs is lower supply voltages. MicroUnity projects that MediaProcessors eventually will be built with supply voltages approaching 1 V, with a power consumption of under 1 W—a level that must be reached for portable and low-cost consumer devices. The company also plans eventually to integrate the MediaProcessor and MediaCodec on a single chip, along with additional memory. This would result in a single-chip device that could take in an RF signal, demodulate and process the signal, and provide a digital or RF-modulated output.

In the near term, MicroUnity has more immediate challenges building and debugging the initial chip set. Not only are there all the risks associated with establishing a new architecture, but there are also the challenges of bringing up a new fab using a process technology radically different than any implemented elsewhere. When technology is pushed in so many directions, there are multiple opportunities for things to go wrong.

The company is not entirely dependent on its process technology, however; the chips can be built with foundry processes, albeit with lower performance levels. The company is also licensing its architecture and chip designs. Should the markets for broadband MediaComputers develop too slowly, the company can apply its technology to existing applications, such as cellular base stations.

Skeptics point out that it is extremely rare for any company to get as far ahead in clock rate as MicroUnity claims to be. Until working MediaProcessors are demonstrated, some skepticism is appropriate. If MicroUnity is able to meet its technology goals, its challenge then will be to find a role in the marketplace. That there will be massive growth in the communications infrastructure can be taken for granted, but the anticipated shape of the new structure seems to evolve weekly. In this regard, MicroUnity's agility, open licensing strategy, and corporate partnerships (which have yet to be announced) could prove to be the critical factors. ♦

