

StrongArm Punches Up ARM Performance

2.0-V Core Implementation Catapults Performance to 240 MIPS



by Jim Turley

Digital Semiconductor has revealed the world's fastest embedded microprocessor core, which promises to vastly boost ARM performance while keeping power consumption low. At last month's Microprocessor Forum, Digital consulting engineer Greg Hoepfner said the StrongArm core will run at up to 215 MHz and deliver 240 Dhrystone MIPS, a sixfold increase over any existing ARM design. StrongArm executes the same instruction set as current ARM7 chips; the higher performance is gained through a combination of circuit-design techniques, higher clock rates, lower voltages, and a new five-stage pipeline. The first chips based on the StrongArm core are expected to ship in 1Q96.

To reach the outrageous clock speeds that have become a hallmark of Digital's microprocessors while keeping power consumption modest, the initial StrongArm implementation will be fabricated in a very low voltage 0.35-micron process. The core will be characterized for either 1.5-V or 2.0-V operation, with a 10% supply tolerance. At 1.5 V, the core can reach 160 MHz; at 2.0 V, StrongArm will run at 215 MHz. In both cases, I/O pins are compatible with 3.3-V signal levels.

Partners Forge a Different Business Deal

StrongArm is the result of a collaboration between Advanced RISC Machines (ARM) and Digital Semiconductor. The initial concept was worked out at ARM's Cambridge, U.K., facility, but the implementation de-

tails were left to Digital. The core logic was designed at Digital's Austin (Texas) facility, with the cache and MMU designs carried out in Palo Alto (Calif.). The company expects to see first silicon in a few days, with formal announcement of price and availability near the beginning of 1996. Fabrication will take place at Digital's Hudson (Mass.) installation on 200-mm wafers.

The agreement between the companies includes turning over the design of this and future StrongArm cores to ARM. ARM is then free to license the cores to its other partners. For its part, Digital will develop a family of packaged parts, which will not necessarily be shared. Future products will leverage the StrongArm core, but they will probably add application-specific peripheral functions. Although Digital must cross-license its core, it is unlikely that other ARM licensees will be able to manufacture StrongArm chips any time soon—at least not at 215 MHz—because of inadequate fab technology, giving Digital a more or less permanent lead in standard parts.

Digital has never offered ASIC services in the past and the company has announced no plans to enter that business with the StrongArm core design. Thus, it will not compete with VLSI or other ASIC-oriented ARM licensees, though once the core is relicensed through ARM, it may appear in other vendors' core catalogs.

Digital's agreement does not allow it to extend the ARM architecture with new instructions beyond what is already allowed through the coprocessor software interface. The company would not be motivated to do so anyway; gaining access to third-party support tools was a major reason to adopt a licensed architecture like ARM in the first place.

Pipeline Bottleneck Removed

The biggest difference between StrongArm and previous ARM-designed cores is its new five-stage pipeline. All previous generations of ARM processors use a rudimentary three-stage pipeline (fetch, decode, execute). Such a simple internal structure has allowed ARM vendors to produce very small and power-efficient cores. But it has also been the biggest bottleneck preventing ARM processors from reaching higher clock frequencies. Although VLSI, GEC Plessey, and others are building ARM7 chips in 0.5-micron geometries and smaller, clock frequency is currently limited to 40 MHz. In contrast, many PowerPC, MIPS, and even Pentium chips easily surpass 100 MHz in similar processes.

ARM7 implementations cram nearly all the work

ARM7:

Fetch	Decode	Execute
latch instr	decode	shift/rotate ALU op D-cache access sign/zero ext commit result

StrongArm:

Fetch	Decode	ALU	Cache	Writeback
calc PC I-cache access	decode cc mux clk ctrl CAM bypass reg file ctrl	shift/rotate ALU op	D-cache access sign/zero ext post result	commit write

Figure 1. Current ARM processors force virtually all operations into the last state of their three-stage pipeline, but StrongArm extends the pipe to five stages to increase clock frequency.

into the final (execute) pipeline stage. As Figure 1 shows, the fetch stage does little more than hold the most recently fetched instruction in a latch. The decode stage deconstructs ARM's fixed-length instruction word and controls the register file and downstream multiplexers. The final stage includes the ALU and the architecture's unusual in-line shifter. For arithmetic or logic instructions with an intrinsic operand shift, the shift and the ALU operation must be performed serially, along with the register writeback, in a single clock cycle, resulting in ARM7's critical timing path.

StrongArm Uses Five-Stage Pipeline

Digital broke this final step into three parts, separating the shift/ALU operations, the buffer/cache access, and the writeback stage to create a less congested five-stage pipeline. With a full clock cycle to execute each function, Digital can push clock frequencies far beyond what was possible with the previous three-stage design.

As Figure 2 shows, StrongArm's fetch stage now computes simple program-counter displacements and accesses the instruction cache. The decode block is essentially identical to that of previous generations of ARM processors, with the addition of power-saving conditional clocking.

The execute stage has a single ALU for calculating all arithmetic and logic operations. In StrongArm, the ALU includes a 32-bit hardware multiplier to accelerate integer multiply and multiply-accumulate operations. A handful of ARM7-based processors also include this multiplier as an optional macrocell.

One of the ARM architecture's unique features is its ability to shift one of the source operands before it is used in an arithmetic operation. A 32-bit barrel shifter placed between the register file's second read port and the ALU performs this step within the time allotted for the execute cycle. When the shift count is specified as an immediate value in the instruction word, it is shunted directly to the shifter's control logic; if the shift count is specified as a register, StrongArm requires an extra clock cycle to extract the value from the register file and deliver it over the Rs bus to the shifter.

The buffer stage includes the data cache, MMU, and bus interface chores like sign- or zero-extending loaded items. The load/store address is sent to the data cache, write buffer, and optional MMU to check for hits or access exceptions. A simple adder, operating during the previous execute cycle, can increment the load/store address for move-multiple instructions. Finally, results are written to the register file in the fifth stage.

Power-Saving Considerations

Keeping power consumption down posed new challenges for Digital's Alpha-trained engineers, for whom 20 W constitutes low power. To reach the company's goal

of less than 500 mW of total power dissipation, including caches and the bus interface, some special design techniques were required.

The chip's low-voltage core goes a long way toward reaching that goal. Operating the core at 2.0 V drops power consumption to about one-third of what a 3.3-V device would consume; at the core's lower limit of 1.5 V, power consumption drops even further, to just 120 mW. Not incidentally, the lower voltages also prevent the small-geometry features from damaging themselves.

Design tricks taken from the Alpha tool bag also appear in StrongArm: many latches are edge-triggered or are fed with conditional clock inputs to reduce power consumption. As in the newest Alpha, clock lines are segmented and are not driven to those portions of the chip that are not used in a particular cycle. Occasionally, function blocks are either bypassed or switched off. For example, the in-line operand shifter is bypassed when

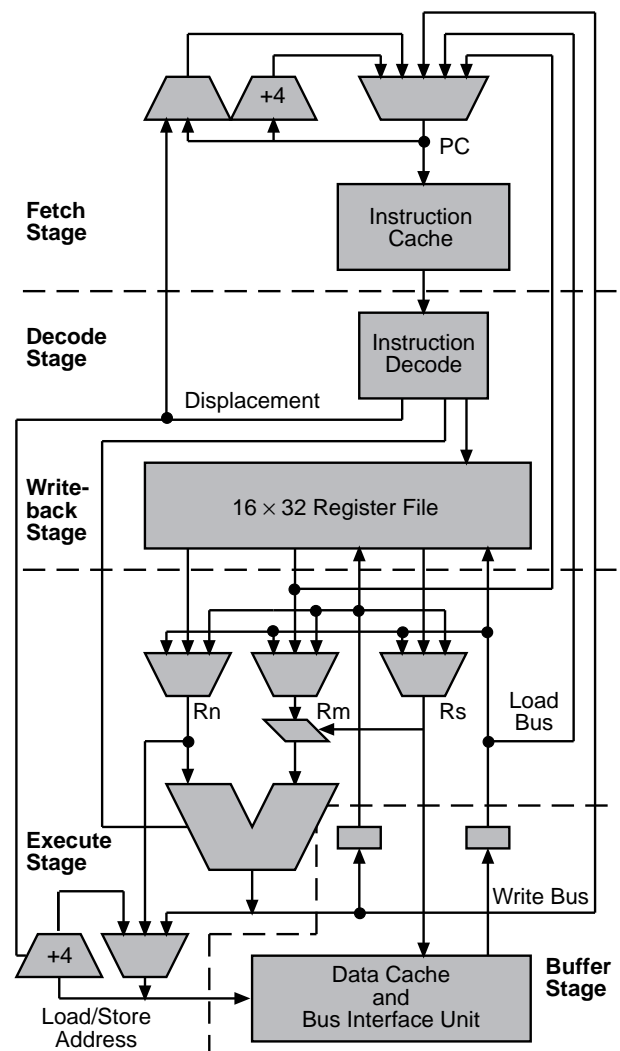


Figure 2. Digital's StrongArm processor core retains ARM's unusual in-line operand shifter but moves cache accesses and register writeback into separate pipeline stages.

the specified shift count is zero.

The StrongArm core is not fully static, but it supports two power-saving modes: sleep and off. To enter sleep mode, software executes a WAITI (wait-for-interrupt) instruction; pipeline activity stops, and clock-distribution trees are shut off. The chip's phase-locked loop (PLL) keeps running, and external I/O pads remain active. In sleep mode, power consumption is reduced to a few dozen milliwatts. An external interrupt will restart the chip after a few clock cycles with no loss of state.

In the off condition, even the PLL is stopped, and typical power consumption drops to just 50 μ A of leakage current from the I/O pads. This condition is more drastic and also irreversible. Only a hardware reset can restart the chip after it has been shut off, and all internal state information is lost.

Bus Uses Variable Clocks

The on-chip PLL allows StrongArm to operate at full speed from a moderate-frequency crystal. In fact, Digital put in some extra effort to make clocking easy and inexpensive. The chip operates from a single 3.68-MHz input—the cheapest crystal oscillator commonly available.

From this modest input frequency, the PLL generates the 160- or 215-MHz pipeline clock. The chip's external bus runs at a selectable fraction of the pipeline frequency, from one-half to one-ninth, or from an asynchronous input of up to 66 MHz. To further reduce power consumption (and wasted CPU cycles), the core automatically drops to the bus's frequency during all external load and store operations.

Multiply Performance Improved

The basic ARM design does not define a dedicated multiplier. In most existing ARM chips, multiply and multiply-accumulate are stepped functions, returning two bits of result per clock cycle (i.e., a Booth multiplier). An optional macrocell speeds integer multiply operations and adds the ability to accumulate 64-bit results. Digital also took this approach, including a hardware multiplier in StrongArm.

Total execution time for 32-bit multiply and multiply-accumulate instructions is 2–4 clock cycles, depending on the magnitude of the operands. A 64-bit operation requires one additional cycle. The multiplier calculates 12 bits per cycle, using an early-out algorithm to shave one or two cycles off the worst-case time if the operands have enough leading zeros or ones. The current version of the ARM instruction set has neither floating-point support nor a divide instruction; integer division is still a

software-controlled hundred-cycle ordeal.

Although the new core is theoretically compatible with the Thumb code-compression module (see [090401.PDF](#)), Digital has no plans to press Thumb into service for its StrongArm chips. The purpose of Thumb is to minimize code size at the expense of flexibility and performance. StrongArm, in contrast, is clearly intended to deliver the best performance possible. Thumb-equipped processors deliver their best results with 8- and 16-bit buses, while initial StrongArm chips will likely have a 32- or 64-bit data bus.

Fab Process Aids Performance

Although the initial StrongArm implementation had not taped out at the time of Hoepfner's presentation, he did reveal some details of its physical specifications. Without caches, MMU, or bus interface, the core logic accounts for 115,000 transistors, or just 4.3 mm² in Digital's 0.35-micron three-layer-metal CMOS-6 process. This is the same process used to produce the company's amazing 417-MHz 21164A Alpha microprocessor (see [0914MSB.PDF](#)), which also has a 2.0-V core. In comparison, StrongArm's 215-MHz pace seems almost leisurely.

Based on Dhrystone 2.1 simulations with sufficiently large caches, Digital estimates the part will deliver 185 MIPS at 160 MHz and 240 MIPS at 215 MHz. These numbers are within the realm of possibility, based on existing ARM7 implementations and the vagaries of Dhrystone. However, Dhrystone is fairly insensitive to bus latency. In real applications, the large difference between StrongArm's pipeline and bus frequencies will severely penalize programs for accessing memory.

Second Coming of PDAs?

The arrival of StrongArm gives new hope to boosters of Apple's Newton PDA. The first Newtons fell somewhat short of expectations as their ARM610 processors struggled to keep up with the onerous task of handwriting recognition. Apple was widely rumored to be considering alternatives to the ARM architecture, such as PowerPC, to increase Newton's anemic performance. StrongArm appears capable of boosting that performance sixfold while keeping power consumption almost identical to the 610's.

With so much additional performance to draw upon, it seems likely that a StrongArm-based Newton could dramatically improve its responsiveness while easily keeping up with basic handwriting recognition. In fact, speech recognition would not be out of the question.



Digital Semiconductors' Greg Hoepfner discusses the internal structure of the StrongArm core.

MICHAEL MUSTACCHI

Price & Availability

Digital Semiconductor has not formally announced the initial StrongARM implementation, to be called SA-110. Chips are expected to begin sampling in 1Q96. For more information, contact Digital (Hudson, Mass.) at 800.332.2717, or 508.628.4760; fax 508.626.0547; or access www.digital.com/info/semiconductor.

With larger caches and a 32- or 64-bit interface to external SRAMs, such a device would have formidable capabilities and still be compatible with previous Newtons.

While there is certainly some incentive for Apple to keep present and future PDAs on the same instruction set platform, it is by no means a requirement. Newton was designed from the outset to be architecturally independent. Applications are distributed in NewtonScript, a portable, interpretive language that could be ported to any microprocessor. Staying with the ARM instruction set reduces Apple's porting time and costs at a time when the company's board of directors has discussed a move to jettison the project entirely. In the end, StrongArm could be the processor that saves Newton.

Or it could be another failure: StrongArm does have competition. IDT's R4650 (see [081504.PDF](#)) delivers 175 Dhrystone MIPS, approaching StrongArm's performance turf, and at 133 MHz, does it with an even better MIPS-per-clock ratio. But the IDT part burns more than 2 W, well in excess of Digital's goal for its first packaged part.

The most impressive MIPS core so far is LSI Logic's recently disclosed CW4020 (see [0915MSB.PDF](#)), which promises more than 120 MIPS in less than 10 mm². If the company can increase the core's clock rate beyond 133 MHz, it could also be a viable competitor to StrongArm. But LSI projects the two-issue superscalar CW4020 will consume well over 1 W for just the core alone.

StrongArm has all these competitors beat, at least for the time being. No other embedded processor yet announced promises to deliver as much performance. The StrongArm core belts out an astonishing 1,540 MIPS/watt (making it the first 1.5 BIPS/watt embedded design), far beyond any other 32-bit core. Digital expects the first StrongArm chip, even as a packaged part, to deliver 300 MIPS/W or better. It is also remarkable for its compactness; at better than 55 MIPS/mm², it outdistances other cores by a factor of three or more. Running at 160 MHz and up, though, StrongArm chips will have a voracious appetite for instructions, making large caches and wide memory buses mandatory.

StrongArm's Effects Felt Beyond PDAs

The majority of ARM-based volume has been outside its two most publicized design wins, 3DO and Apple.

NEC Picks Up ARM License

NEC Electronics recently acquired a license from ARM to add the ARM7 core to its stable of microprocessor cores. The agreement also includes the Thumb module, built-in debug features, multiplier, and options for other designs. This makes NEC the tenth ARM licensee announced so far, and the largest by a wide margin.

Until now, ARM had been fairly careful to select licensees that did not compete directly with each other, either geographically or in terms of market focus or technical capability. NEC, however, is likely to infiltrate the markets of a number of fellow licensees, a situation that has other vendors up in arms.

VLSI Technology, in particular, will now face competition for high-volume ASIC business from the formidable forces at NEC. Texas Instruments, Sharp, and Cirrus Logic will also likely see their markets eroded as NEC moves onto their turf.

Opinions are divided even within NEC. The company has been an active MIPS licensee and has also developed its own proprietary V800 family for the relative low end of the 32-bit market. With yet another 32-bit architecture now in its portfolio, it is not clear where the company will be focusing its efforts. The new V852 notwithstanding, developments in the V800 family may slow considerably. The move may have come at the behest of a European ASIC customer that either was not happy with the capabilities of MIPS or simply wanted to promote a European-designed architecture.

ARM has indicated that negotiations with still more potential ARM licensees are in the pipe "to add breadth of applications and market" to its list of adherents. If the company proliferates its architecture much further, however, it may generate internecine fighting among the ranks as more companies battle for the same design wins.

3DO has already moved on, selecting IBM's 602 for its next-generation video game platform (see [090704.PDF](#)), and Apple—at least publicly—has not committed to the future of Newton.

Many more ARM chips wind up in automotive, telecommunications, and global-positioning applications. In most cases, these are systems that will not immediately benefit from StrongArm's increased performance. Instead, Digital's new chip design will be an enabler for applications that are currently not possible, like voice-activated PDAs.

At the very least, StrongArm should allay the fears of observers concerned about the architecture's viability. Like a nightclub bouncer, StrongArm is successful just by being visible. For potential customers concerned about ARM's apparent lack of an upgrade path, StrongArm is a sign that there is strong performance potential ahead. ♦