WinChip 4 Thumbs Nose at ILP

IDT Targets High Frequency to Challenge Mendocino

by Keith Diefendorff

"Tricks are better than transistors." So sayeth Glenn Henry, president of Centaur Technology (a division of IDT), laying out the philosophy behind the company's new WinChip 4 processor at the recent Microprocessor Forum. The new processor eschews the generalized superscalar and out-of-order techniques that many of its competitors are relying on to boost instruction-level parallelism (ILP) and, hence, performance. Henry says these techniques, unless applied very judiciously, just increase complexity, reduce the clock rate, and add cost without much performance gain; he prefers simple, fast, and cheap.

As evidence that this approach is viable, Centaur points to its current WinChip 2, which, at 58 mm^2, is the smallest x86 processor available; yet it performs as well as its superscalar counterparts on a cycle-for-cycle basis. Unfortunately, at only 266 MHz, WinChip 2 doesn't get enough clock cycles to match the competition on performance. But Henry is confident these frequency limitations can be overcome while maintaining a simple design and a small die size.

To that end, the Centaur team has started from scratch to design the WinChip 4 (which it abbreviates to "C4"), with frequency as the top priority. Using a long 11-stage pipeline and custom-designed dynamic circuits, the processor is targeted for 500 MHz in IDT's 0.25-micron six-layer-metal CMOS-10.5 process. Instead of a complex high-ILP core, the C4 invests most of its 11.5-million transistors in the memory system, as Figure 1 shows, providing large 64K instruction and data caches, large 128-entry TLBs, extensive buffering, and smart prefetching.

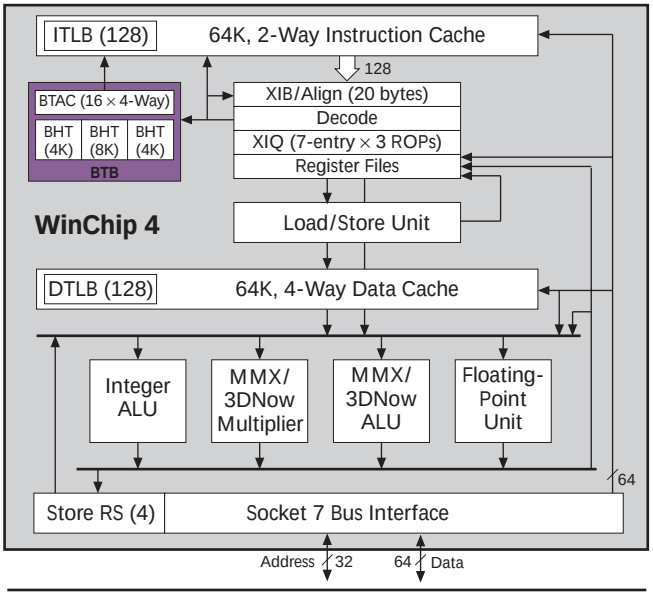


Figure 1. Centaur's WinChip 4 uses a sophisticated branch predictor (purple) to reduce stalls in its simple in-order pipeline.

The C4 will occupy just 100 mm^2 of silicon, giving it a manufacturing cost of less than $40, according to the MDR Cost Model. The processor is now in final layout, and IDT expects to sample it to customers by the middle of 2Q99, with volume shipments beginning in 2H99. By 1H00, the part will move into IDT's 0.18-micron CMOS-11.5 process, reducing its size to 60 mm^2 and lowering its manufacturing cost by 25%. In the newer process, the part should clock at up to 700 MHz, approaching the top frequency of processors we expect from Intel, AMD, and Cyrix. Interestingly, the C4 is designed to also be manufacturable in IBM fabs, although IDT appears capable of meeting demand and has no current plans to exercise the IBM option.

If WinChip 4 meets its stated goal of achieving the same IPC (instructions per clock) as Intel's Mendocino, it should compete favorably on performance against the Celeron line, which does not appear to be targeting such high frequencies in the same timeframe. With its smaller die size, the Socket 7–based WinChip 4 can be priced aggressively against other contestants for the low-end PC market, such as AMD's K6-3 (a.k.a. Sharptooth), Cyrix's MXi and Jedi (see MPR 12/7/98, p. 4), and Rise's mP6 II (see MPR 11/16/98, p. 1).

Philosophy Drove the C4 Design

Centaur is big on philosophy. Rarely is a processor designed with such clear philosophical underpinnings, but doing so has advantages. For one thing, it protects the design from external forces that tug it in different directions, usually increasing complexity and always extending development time. As a testament to this approach, Henry's 20-person design team will take the C4 from start to tapeout in only eight months. This includes the time to develop several CAD tools, which were needed because commercial ASIC-style design tools were not suitable for WinChip's custom high-frequency design.

One overarching philosophy behind the C4 is that logic transistors are bad, and control-logic transistors are worse. Hence Centaur's decision to forgo generalized superscalar dispatch and out-of-order execution. But what the C4 gives up in parallel execution, it hopes to regain in fewer pipeline stalls. Indeed, most of the chip's features are designed specifically to eliminate stalls. For example, the pipeline sets the cache ahead of the ALU to eliminate load-use stalls. To minimize branch-induced stalls, the chip uses one of the most aggressive branch predictors of any current x86 processor.

Most of the C4's transistors are dedicated to caches and buffers to reduce the number of external memory accesses and to match the high-speed internal data paths with slow external memory. Logic transistors are minimized by implementing tricks narrowly focused on eliminating the common causes of pipeline stalls. The C4 avoids broad-brush solutions, which consume large numbers of transistors but result in only small incremental improvements. Centaur sees little sense in spending transistors on scarce ILP when performance is constrained by branches and bus traffic anyway.

Caches, TLBs Reduce Memory Traffic

The C4 provides a 64K two-way set-associative instruction cache and a 64K four-way set-associative write-allocate data cache. Like many processors, the C4 prefetches sequential lines into the instruction cache. But unlike most processors, it also prefetches data into the data cache. The data prefetch algorithm is not a complex one, simply prefetching lines sequentially, but it is an intelligent algorithm in that it monitors all internal activity to continuously adjust the aggressiveness of the algorithm and prevent prefetches from interfering with more urgent demands for the bus.

Centaur has elected to go with large L1 caches rather than the increasingly popular alternative of small L1s backed by an on-chip L2. To make this decision, Henry's team simulated a processor with dual 32K L1 caches backed by a 256K on-chip L2 and another with dual 64K L1s and no L2. They found that while the former had 1–2% better performance on the applications they simulated, it also required 40% more die area—a tradeoff that's hard to swallow when your target is low cost. This decision is somewhat curious, however, considering that the architects of Mendocino, the K6-3, the mP6 II, and Cyrix's M3 (see MPR 11/16/98, p. 24) have all come to the opposite conclusion.

Centaur paid special attention to address translation. According to Henry, with simple TLB structures such as the 32/64-entry TLBs on Pentium/MMX, as much time can be wasted on address translation as is lost to cache misses. The problem is getting worse as the memory footprint of software continues to grow. To reduce address-translation delays, the C4 has large 128-entry eight-way set-associative TLBs, one each for instructions and data, and a 16-entry page-descriptor cache, which Centaur claims gives a 99% hit rate for page descriptors.

Although the C4 issues, executes, and completes instructions in strict program order, it does perform some limited reordering of loads and stores, which is critical for any modern CPU. Stores that are waiting on data from previous instructions are set aside in a four-entry reservation station, allowing subsequent instructions, including loads that hit in the cache, to proceed. Once data arrives in the reservation station, it can be forwarded directly to subsequent loads, thus avoiding some cache accesses altogether.

Stores that miss the cache are transferred to a 4-entry × 8-byte write buffer. While waiting for the missed line to be allocated in the cache, subsequent stores can be gathered into the write buffer, thus minimizing the number of write cycles to the bus. With this feature, also used in other processors such as the P6, a series of eight single-byte stores to sequential addresses, for example, can be collapsed into a single quad-word write, saving considerable bus bandwidth. Henry says that write combining and write-allocate each improve performance by 1.5% on Winstone 98; together these are equivalent to about one speed grade for most processors.

Short-Decode, Long-Execute Pipeline

The C4's pipeline, while similar in length to the P6's, uses fewer stages to get instructions into execution, but more to execute them. The C4's approach is like that taken by Rise in the mP6: as Figure 2 shows, cache accesses occur ahead of ALU operations, thus eliminating the load-use penalty for the all-important load-op instructions as well as for loads followed by dependent reg-op instructions.

Centaur's approach has the downside of increasing to two cycles the address-generate interlock (AGI) time for memory-referencing instructions that need the result of a previous load to compute their memory address; those that need an ALU result interlock for three cycles. The C4 eliminates the AGI for loads and stores after simple adds, moves, and increments, by using the adder in the address generator as a simple ALU, a feature that speeds many stack operations.

Like the mP6 and Cyrix's Jalapeno, the C4 fetches instructions asynchronously, staying ahead of decode and execution. To implement this capability, the C4 provides buffers between the early pipeline stages. The first stage of the pipeline fetches 16 instruction bytes from the cache. In the second stage, the cache tags are checked, the way selection made, and the instruction bytes placed in a 20-byte buffer called the XIB. These two stages repeat to keep the XIB full, sequentially prefetching lines into the cache as necessary.

The XIB is a variable-length shifter used for aligning the next instruction with the instruction decoder. One trick Henry's team devised is to detect short forward branches whose target is in, or on the way into, the XIB. In these cases, and when the branch is predicted taken, the target instruction

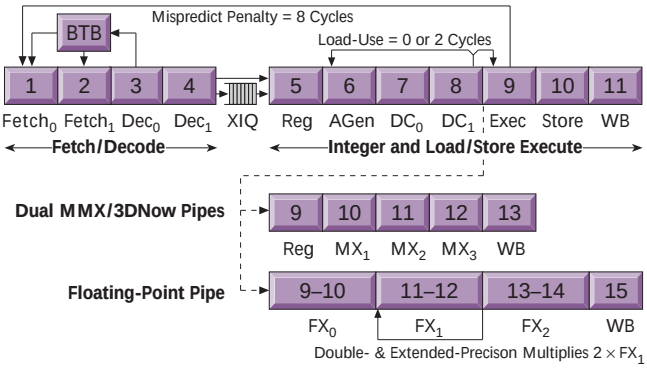


Figure 2. The C4's pipeline places cache access ahead of the execute stage to eliminate the load-use penalty for many instructions.

For More Information

To view Glenn Henry's Microprocessor Forum presentation, point your browser to *www.winchip.com/mpf98*.

is simply shifted into place on the next cycle, eliminating the instruction-cache fetch cycle that would otherwise be needed.

Instruction decode begins in stage 3 of the pipeline. Between stages 3 and 4, partially decoded instructions are temporarily held in a four-deep queue, the primary purpose of which is to prevent the stage-3 decoder from stalling when stage 4 is stalled or is busy decoding a complex (microsequenced) instruction. After instruction decode is completed in stage 4, the decoded instructions enter a seven-entry queue called the XIQ. If the XIQ is empty, instructions go directly to stage 5 without delay. The stage-4 decoder can decode two x86 instructions per cycle, one complex and one simple. This dual-decode capability keeps the XIQ relatively full, so that fetch bubbles are absorbed.

Register Renaming: "Work of the Devil"

An instruction waits in the XIQ until its operands are available and an execution pipeline slot is available. Instructions are issued from the queue in strict program order, making it extremely important that the execution pipeline be as free of hazards as possible. Most processors implement register renaming to avoid stalling the pipeline on antidependencies and output dependencies. But Henry calls register renaming "the work of the devil," referring to the extra complexity that brings with it evil side effects, such as reduced clock rate. The C4's simple in-order pipeline avoids most of these false pipeline hazards, making register renaming superfluous.

Operands are read from the registers (or from the forwarding buses) in stage 5 of the pipeline. All instructions

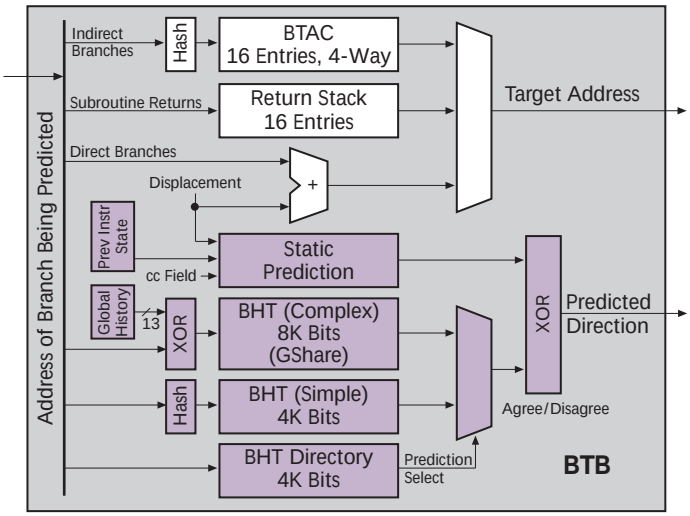


Figure 3. The C4's BTB uses three methods (purple) to predict branch directions and three (white) to predict target addresses.

then pass through stages 6, 7, and 8, where memory addresses are calculated and the data cache accessed, if necessary. From stage 8, instructions pass to the execution units.

The C4 implements two parallel MMX/3DNow pipelines, allowing arbitrary dual issue of these instructions. With only a single integer pipeline, however, the dual issue of integer instructions is limited to pairs in which one of the instructions is a load, store, or load effective address. While somewhat restrictive, this mechanism handles many important pairing cases, improving ILP while avoiding the cost of two complete integer pipelines, one of which would often sit idle. Loads followed by dependent instructions can be paired, as can instructions followed by dependent stores. Floating-point instructions, however, are never paired with any other instruction. Henry agrees with the Cyrix architects that issuing more than two instructions per cycle gains very little additional performance.

The C4's integer ALU executes all operations in one cycle (except multiply and divide). MMX instructions also have a latency of one cycle, except for multiplies, which take four cycles (one cycle longer than Mendocino's). The 3DNow instructions have a four-cycle latency, and all 3DNow and MMX instructions have single-cycle throughput.

To reduce design time, Centaur cut a corner with the C4's FPU, reusing the same base unit that is in the WinChip 2 (C2) with a few minor enhancements. Not wanting to re-pipeline the unit for the C4's higher clock rate, Centaur chose to clock the FPU at half the CPU frequency. In C4 clocks, the latency of single-precision operations and all precisions of FP adds is six cycles, with a throughput of one instruction every two cycles; double- and extended-precision multiplies have an eight-cycle latency and four-cycle throughput. This is noticeably worse than Mendocino, which executes single-, double-, and extended-precision FP multiplies with five-clock latency and two-cycle throughput (3/1 for FP adds).

The enhancements Centaur made to the C4's floating-point capability are limited to zero-cycle FXCHGs, out-of-order stores, and faster load-FPU processing. As in most x86 processors, FP instructions that cannot generate an exception proceed asynchronously without blocking the other pipelines. Most FP instructions fall into this category, since most real code executes with exceptions disabled.

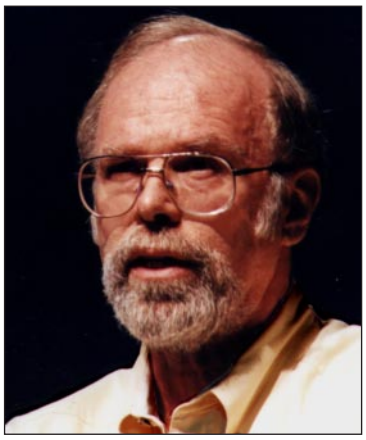
Cutting this corner will put the C4's FP performance below its competitors' on a per-clock basis. But Henry defends the decision, pointing out that much of the demand for floating-point performance is shifting to 3DNow anyway, which the C4 handles at competitive rates. But the decision is still questionable, as it leaves the C4 with a weakness its competitors can point to, putting IDT on the defensive.

Aggressive Predictor Reduces Branch Stalls

Although most tradeoffs in the C4 were made in favor of simplicity, the one area where Centaur had to invest some serious hardware is the branch predictor. While this may seem contrary to the basic philosophy, it was necessary to achieve some

modicum of performance with its simple in-order design, which mandates that all sources of pipeline stalls be minimized. Unfortunately, not everything can be simple.

As Figure 3 shows, the C4 uses three different mechanisms to predict branch-target addresses and three to predict branch directions. The target addresses of register-indirect branches are predicted from a 16-entry four-way set-associative branch target address cache (BTAC), indexed with a hashed version of the branch address. Subroutine return addresses are predicted from a 16-entry return-address stack. Direct (program-counter relative) branches are computed by a superfast adder rather than being predicted from a BTAC as in processors such as the P6, Jalapeno, and the mP6. The C4's approach has the advantage of always getting the target address correct but the disadvantage of delaying it—and thus the fetch of a new instruction stream—by a couple of cycles, until the branch can be decoded far enough to determine the displacement. But these cycles are normally absorbed by the XIQ, and only 15% of the predictions actually cause bubbles in the execution pipeline.



MICHAEL MUSTACCHI

Glenn Henry, president of Centaur Technology, describes IDT's new WinChip 4 at the Forum.

On Winstone 98, according to Henry, indirect branches account for 1.2% of the dynamic instruction mix, 65% of which the C4's BTB predicts correctly. Returns from subroutines are 2.6% of the instruction mix and are correctly predicted 90% of the time. Direct-branch addresses, correctly predicted 100% of the time, account for 18.5% of all instructions. The C4 does not predict far branches, but these are infrequent, accounting for only 0.7% of the dynamic mix.

The C4's branch-prediction complexity shows up more in the methods it uses to predict branch directions than in those it uses to predict target addresses. The C4's simplest predictor is its static predictor, which predicts branch directions with 70% accuracy based on the branch direction, the condition code being tested, and the type of instruction that last set the condition code. This latter feature is unique among static predictors and improves accuracy by several percentage points. Easily predicted branches, such as branch on overflow, are confined to the static predictor, conserving branch history table (BHT) entries and thus improving the accuracy of the dynamic predictors.

Dual Dynamic Predictors Are Small But Effective

According to branch theory, there are generally two types of branches: simple branches, such as loop-closing branches that are easily predicted from the direction the branch last took, and complex branches, e.g., those found in complex control-flow sequences. The latter usually require a longer branch history to achieve good prediction accuracy.

To handle these two branch types, the C4 implements two dynamic predictors. The simple predictor is a 4K-entry BHT, each entry containing a single history bit. The simple BHT is indexed by a hashed version of the branch address.

The complex predictor is a Gshare predictor (see MPR 11/17/97, p. 22), which uses a 13-bit global history register XOR'd with the low-order bits of the branch address to index an 8K-entry BHT. The single-bit entries in both the simple and the complex BHTs implement "agrees" mode, which, due to the accuracy of the C4's static predictor, is nearly as effective as the more complex two-bit predictors used in processors such as the P6, according to Henry.

On every branch, both dynamic predictors make a prediction. A third 4K × 1-bit BHT holds a directory that tracks which of the two predictors made the correct prediction on the previous occurrence of the branch. This BHT is updated after the branch is fully resolved and the correct direction is known. As a further optimization, only the simple or complex BHT that made the correct prediction is updated with new history. In this way, simple branches do not pollute the complex predictor, or vice versa.

With this scheme, Henry says that the C4 correctly predicts the address and direction of over 95% of the branches in Winstone 98. Winstone is a reasonably good test of a branch predictor, as the branches are not generally easy to predict. This prediction rate compares favorably with the P6's but requires 40% less storage space. Henry admits that the prediction accuracy could be improved by implementing two-bit saturating up-down counters in the BHTs, but he says that the C4's dual-table single-bit predictor makes a better prediction than would a single-table two-bit predictor with the same number of bits. Although the C4's branch mispredict penalty of eight cycles is not excessive compared with those of its competitors', which also have long pipelines, the C4 must avoid more stalls to make up for the lack of parallel execution.

Parts for High Speed and Low Power

IDT will offer the WinChip 4 in both a Socket 7–compatible CPGA-296 and a 320-contact PBGA for notebooks. To reduce cost, the chip will be wire bonded, rather than flip-chip mounted, into these packages. Even though the wire-bond pad ring increases die size by about 15 mm^2, IDT still finds a lower net cost.

In 0.25-micron technology at 2.5 V, a 500-MHz C4 will dissipate a maximum of 16 W. For notebooks, IDT intends to ship a version of the C4 at a lower frequency and voltage. The company has not yet determined the voltage or frequency for the low-power version, but operating it at 2.2 V and 400 MHz would bring it into the 10-W thermal envelope of notebooks. In 0.18-micron technology at 1.8 V, the part should dissipate around 6.5 W at 500 MHz.

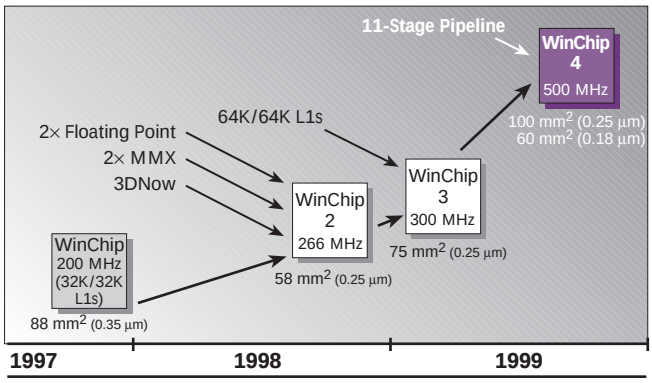


Figure 4. The C4 brings an 11-stage pipeline to the WinChip family, allowing the new processor to achieve much higher frequencies.

Challenges Lie Ahead for WinChip

While the C4 may eliminate the frequency disadvantage IDT has suffered with its current WinChips, other obstacles remain. One is Socket 7. AMD's momentum (we expect 20 million Socket 7 units next year) will keep Socket 7 viable through 1999. But by 2000, Socket 7 will face increasing pressure from Intel's Socket 370 (see MPR 12/7/98, p. 3). With AMD moving to Slot A for its next-generation CPUs, and with Cyrix focusing on integrated parts after Jedi, IDT would be left with Rise to sustain the Socket 7 market, a task they have no hope of accomplishing. As a consequence, IDT must eventually find a new infrastructure for its chips. Socket 370 is an obvious choice.

At one point, IDT had planned to integrate a north bridge onto its C3 processor, but these plans were abandoned when the company discovered that customers wanted higher levels of integration, including 3D graphics, or nothing. Perhaps integration is an eventual way out of the Socket 7 dilemma for IDT, but Cyrix is far ahead in this game and, with National's support, makes a formidable competitor.

Feature	Centaur/IDT WinChip 2	Centaur/IDT WinChip 4	AMD K6-3	Rise mP6-II	Intel Mendocino
x86 Decode	1 complex	1 cpx + 1	2 complex	3 complex	1 cpx + 2
Issue Rate	1–2 x86	1–2 x86	6 ROPs	3 x86	5 ROPs
Reorder Depth	None	None	24 ROPs	None	20 ROPs
Pipeline Stages	6 stages	11 stages	7 stages	8 stages	12–14
BHT Entries	4K × 1b	16K × 1b	8K × 2b	512 BTB	512 BTB
Return Stack	8 entries	16 entries	16 entries	8 entries	4 entries
L1 Cache	32K/32K	64K/64K	32K/32K	8K/8K	16K/16K
On-Chip L2	None	None	256K	256K	128K
ITLB/DTLB	128/128	128/128	64/128	32/64	32/64
3D Extension	3DNow	3DNow	3DNow	None	None
Clock Rate	266 MHz	500 MHz	450 MHz*	>200 MHz*	450 MHz*
Transistors	5.9 million	11.5 million	21 million	19 million*	19 million
IC Process	0.25µ 5M	0.25µ 6M	0.25µ 6M	0.25µ 5M	0.25µ 5M
Die Size	58 mm²	100 mm²	118 mm²	170 mm²*	154 mm²
Power (max)	12.5 W	16 W	15 W*	8 W*	27 W*
Socket/MHz	S7/100	S7/100	S7/100	S7/100	S370/100*
Production	Now	2H99	1Q99	1H99	2H99
Est Mfg Cost*	$25	$40	$45	$70	$55

Table 1. The WinChip 4 will have the lowest manufacturing cost of the low-end processors, and its high frequency should put it in good standing against the competition. (Source: vendors, except *MDR estimates)

Another problem is the same one facing all Intel's competitors: KNI. While IDT, Cyrix, and AMD all stand solidly together behind 3DNow today, as Katmai volumes ramp, KNI will capture the mindshare of software developers, leaving 3DNow in the lurch. Rise has already indicated that it will forgo 3DNow and go straight to KNI on future generations of chips. Sources indicate that the other vendors are also plotting their strategies for migration to KNI. IDT will eventually have to do the same. The company is very practical about such matters and has demonstrated an amazing ability to produce new chips on short schedules. Thus, Centaur should be able to make the leap to KNI without much difficulty. Since KNI is not likely to migrate to the low end of Intel's roadmap until 2000, the C4 should be safe for a while. But the next WinChip must deal with this issue.

Today, IDT ships about 250,000 WinChips per quarter, primarily outside the U.S., where low price is more important than high frequency. Although this rate represents only about 1% of the market, the sheer size of the market makes it a sustainable business for IDT. With the C4's higher frequency, IDT could hold its own in non-U.S. markets and potentially grow its overall market share to as large as 4–5%, allowing IDT to declare success.

To reach the 10%-share level, which would solidify its position and qualify it as a serious force in the market, IDT must also capture a significant portion of the U.S. market. The obstacle the company faces in this regard is the one that has plagued it since birth: the U.S. market's reluctance to adopt non-Intel CPUs. While AMD has largely overcome this barrier, and Cyrix is making progress, IDT has not managed to land a top-tier customer to lift it over the hump. AMD's plans won't help, as it is gearing up its fab capacity to dominate even the low-end PC market.

It's possible that current WinChips have just not been compelling enough to attract a top-tier customer. As Figure 4 shows, IDT made its initial foray into the PC market with the WinChip, which it improved a year later with the WinChip 2 and will tweak again with the WinChip 3 early next year. Although these parts are impressively small, their low price has not been enough to overcome their frequency disadvantage, which has consistently placed them a speed grade or two behind Intel's and AMD's low-end processors. In the U.S. market, Intel establishes the threshold of acceptability, and failure to match its lowest speed grades eliminates any chance of market entry.

But with higher clock speeds, IDT's simple in-order design philosophy may yet prove successful. Current WinChips perform nearly as well as their more complex competitors on a per-clock basis but carry a lower cost, as Table 1 shows. If Centaur—and the IDT fabs—can actually deliver a 400- to 500-MHz WinChip 4 in the $50–$70 price range next year, IDT just might land its needed high-volume customer. M