Intel's P9 Could Make 286 Architecture Obsolete

But It's Too Late for OS/2

It has been widely rumored that Intel has developed a microprocessor, internally called the P9, that has a 16-bit external data bus but uses the 386 architecture. Some published accounts have stated that it is pin-compatible with the 286, but this has been denied by knowledgeable sources. The P9 would allow new 16-bit bus machines to have the benefits of the 386 architecture, while retaining the cost benefits of the 16-bit bus. Some large customers are said to have samples now, and public announcement is expected this fall. Intel declined to comment.

In many respects, the P9 is to the 386 as the 8088 is to the 8086. (Could it be the 80388?) The narrower data bus reduces component cost by allowing a smaller package and reduces system cost by halving the number of data bus buffer ICs, connector pins, and traces, and reducing memory cost for small memory configurations.

The real need for this device, however, is a result of design flaws in the 286, primarily with regard to memory management and 8086 emulation. Intel has had difficulty designing a truly 8086-compatible processor with additional features, as evidenced by problems with the 186 and 286. With the 386, Intel had another chance to do it right, and it seems that they have come closer this time. The 386's 32-bit address and data buses increase performance and make good advertising copy, but the architectural improvements are even more important.

Because the 286 operating in protected mode is not object-code-compatible with 8086 programs, IBM's OS/2 will not allow multiple unmodified 8086 programs to run concurrently. Existing programs can run unmodified only in the DOS 3.x "compatibility box."

DOS 3.x programs must be modified to be fully supported under OS/2. Thus, software vendors will be forced to do additional work to support this OS. A 386-based OS, on the other hand, will be capable of running multiple DOS 3.x programs concurrently, eliminating the need for this rewriting. But because of the current dominance of the 286, IBM chose to base OS/2 on the 286 architecture.

It has become apparent that OS/2 will be slow and will require large amounts of memory (2 megabytes minimum if the compatibility box is used). All things considered, the personal computer world may well have been better off without OS/2. Many users may skip it entirely, and continue with DOS 3.x until they move up to a 386-based machine with a 386-based OS.

The P9, if it had been available when IBM was planning OS/2, would have made it possible to go directly to an OS based on the 386 architecture. IBM's Models 50 and 60 could have been based on the P9, with its less-expensive 16-bit data bus, while having all the advantages of the 386 architecture. If a P9-type chip that is pin-compatible with the 286 were available, existing ATs and AT compatibles could be upgraded by simply changing the processor chip.

Intel has learned a painful series of lessons in microprocessor upgrading and compatibility with

---

IN THIS ISSUE

Bugs and Quirks

This regular monthly column provides information on bugs and idiosyncrasies in widely-used ICs. If you have knowledge of such bugs (ideally a vendor-provided bug list), send it in — we'll provide a free three-month subscription extension to the first person to send in a bug list that we use.

80386 Multiply Bug is One of Many

The 80386 has such a high profile that the bug in the 32-bit multiplication function has received widespread publicity, from *PC Week* to *Time Magazine*. Less well know are the numerous other subtle bugs. The saving grace is that these bugs primarily affect protected mode operation or instructions using 32-bit addressing, neither of which is used in a "super-AT" running MS-DOS, and all have work-arounds.

If you're designing with the 386 or writing assembly language software for it, ask your Intel rep for a copy of the *80386-B1 Stepping Information*. It's 12 pages long, and lists 10 "specification changes," 20 "errata," and 4 "design notes." This document applies to 16-MHz parts with the label code S40344 and 20-MHz parts labeled S40362. The devices can also be identified by the value 03 in both DH and DL after reset. The next release will be the D stepping, which is expected to fix all these bugs and to be in production early in '88.

Most of the items in this document are too technical to detail here. One, however, may be of general interest to software developers. The "Bit String Insert" and "Bit String Extract" instructions, which are listed in the data sheet and documented in the programmer's reference manual, will not be implemented. Intel's claim is that their multi-bit shift and rotate instructions are sufficient, and the complexity of implementing the bit string instructions was not justified. (Perhaps they were implemented and didn't work, and Intel doesn't think they're worth fixing?) TI's 34010 and Motorola's 68020 both include similar bit field extract and insert instructions, which are extremely useful for managing bit-mapped displays.

Altera EPLD Problems in Standby Mode

Altera's EP600, 900, and 1800 devices have an automatic standby mode which is initiated whenever all inputs have not changed for approximately 100 ns. This feature is designed to save power when operating with slowly changing inputs. Unfortunately, if the device is in standby mode, glitches may occur on combinatorial outputs when unrelated inputs change. So, unless your system is fully synchronous and can tolerate glitches on the EPLD's outputs, set the "Turbo" bit to prevent the chip from ever going into standby mode.

Motorola 146818 Subject to Data Loss

The 146818 clock/calendar chip is used in the AT, compatibles, and PS/2 machines. This chip has a host of peculiarities, and has a tendency to trash the contents of the time and date registers when certain timing conditions are violated. In particular, if CE/ changes state within a certain window on either side of the falling edge of AS, the internal circuitry can malfunction, causing corruption of the time and date. The "A" version may fix the problem, but is sole-sourced.

The oscillator also requires care. First, don't use the on-chip oscillator — it's a power hog, and a much lower-power oscillator can easily be built with 4069's. Second, be very careful in the layout of the oscillator circuit. Noise coupled into oscillator signals from adjacent digital traces can also cause corruption of time and date.

From The Editor

Welcome to the premier issue of *Microprocessor Report*. Our charter is to be the best information resource available for designers of microprocessor-based hardware. We're here to give you the information you need to do your job better and with less effort.

In this issue, we begin our coverage of the Micro Channel with an article on its automatic configuration features. Next month we'll begin our coverage of the NuBus with an overview of its features, and details of Apple's implementation in the Mac II.

This month's IC Selection Guide covers one-megabit dynamic RAMs, devices that most of you are likely to be using in the next year or two. Next month's IC Selection Guide will feature SCSI interface ICs, and we'll also have an article on debugging tools for SCSI.

This month's tutorial feature is an excellent description of the troublesome phenomenon of metastability. This article will continue next month with design guidelines for minimizing potential problems. In coming months we'll provide tutorials on cache memory, battery-backup circuits for CMOS RAM, and use of interleaved access and page mode to reduce average memory access time while using slower memory ICs.

Please feel free to call or write with your comments, criticisms, or suggestions. I'd especially like to know what you find most and least useful in this issue, and what you'd like to see covered in future issues. We'll do everything we can to make *Microprocessor Report* your first source for up-to-date information.

Michael Slater

In The News

Motorola Announces 68030 First Silicon

In a press release with more than the usual amount of hype, Motorola announced that they had "achieved first silicon April 16, 1987" on the 68030 "superchip." All the hype aside, the 68030 does promise to be a very popular chip, with up to twice the performance of a 68020 and on-chip memory management. According to Jeff Nutt, Technical Marketing Manager for the 68000 family, Motorola has UNIX up and running on the 030, and has delivered samples to selected customers. General sampling is slated to begin in October, with full production by the end of the year. A Technical Summary (an abbreviated data sheet) is available from Motorola sales offices; full data is available now on a non-disclosure basis, and will be generally available in October.

If you're anxious to get moving on a 68030 design, check out the article "PLDs and 68020 Provide Launchpad for 68030 Design" on page 45 in the August issue of ESD. It provides a complete schematic for a circuit that combines a 68020, 68851 MMU, and some glue logic, and plugs into a 68030 socket. It's not a perfect emulation and doesn't implement the data cache or burst-mode transfers, but it should allow considerable hardware and software debugging to be done before 68030 samples are available.

Intel Takes to the Courts

Intel is keeping its lawyers busy these days. First, there's the battle with AMD over the termination of their 10-year technology exchange pact, which centers on rights to the 386. Intel is saying "no way" to any 386 second source, and is now trying to take away AMD's rights to the 186, 286, and 80C51. AMD claims to have provided Intel with a long list of chip designs as part of the exchange pact, while Intel claims that all but one were unacceptable. AMD claims not having the 386 is costing them $1 million per week. The case has gone to arbitration, and a hearing is scheduled for Sept. 8.

Fanning the flames is AMD's announcement of a 16-MHz 286. Intel says it has no intention of releasing such a part. Is AMD's process that much better, or is Intel just protecting the market for the 386 (and the P9)? Meanwhile, clone makers are busy designing 16-MHz machines, for which they will have to buy the processors from AMD.

AMD is not the only company with whom Intel is in court. Intel is suing Hyundai for allegedly infringing on Intel's EPROM patents and Vitelic over DRAM patents. And the litigation with NEC over the V-series continues to drag on, with NEC in the second round of appeals trying to disqualify Judge Ingram for owning $80 worth of Intel stock. Now Intel is asking Customs to seize incoming shipments of V-series parts for copyright infringement. Whatever the final outcome of this squabble, Intel will have achieved their goal — the V-series parts will not get significant numbers of design wins in the U.S.

Intel has also been on the other side of the fence. SMC approached Intel about Intel's infringements on SMC's process patents, and the end result is a patent cross-licensing arrangement and an agreement to alternate-source each other's Ethernet chips.

Motorola and Thompson Settle 68020 Battle

Motorola seems to share Intel's desire to be the sole source of its 32-bit processors. Thompson had sued Moto for half a billion dollars over rights to the 68020. That case has been settled, with a transfer of funds to Moto and a transfer of "limited technical information" to Thompson. Details have not been disclosed, and it remains unclear whether or not Thompson will produce the 68020. Moto now has a partnership with Toshiba, and Toshiba would presumably be another choice for a 68020 alternate source. The pending embargo against Toshiba could make this sticky, though. You can expect the 68030 to be sole-sourced for quite a while, if not indefinitely.

Alternate Sources No Longer Mandatory?

It used to be that a vendor had to have a second source to get significant design wins. With the latest generation of microprocessors, it seems that users aren't insisting on a second source. In the case of the 386, the only company with enough clout to insist on a second source is IBM, and they've got one — themselves. Systems manufacturers wanting to upgrade 68000 systems need the 68020 and 68030 desperately, and can't afford to haggle over availability of alternate sources. From the IC vendors' point of view, this is a great opportunity to keep the entire market to themselves and eliminate price competition.

If you can't beat 'em...

Intel obviously prefers to reserve its fab capacity for several-hundred-dollar processors and coprocessors, rather than jellybean memories, but they also want to be a full-line vendor. The solution: they have entered into agreements to sell, with Intel labels, Samsung DRAMs and Mitsubishi EPROMs. The Mitsu EPROMs are intended for sale to Intel's customers in Japan. Mitsubishi is also making 8051s for Intel.

IBM Launches First PS/2 Legal Attacks

IBM has made it clear that they will vigorously defend the PS/2 architecture. They've begun by suing Orchid and AST, not over their hardware, but over their advertisements. It seems IBM considers it a violation of their trademark for Orchid to advertise their memory board as a PS/2 product. Presumably PS/2-

Designer's Guide to Synchronizers and Metastability, Part I

By John Wakerly, Stanford University

Design Tutorial

Most digital systems designed today are *synchronous* — a single free-running oscillator provides a master high-frequency clock from which all other timing signals are derived. *Asynchronous* signals, which are not derived from the master clock, must be synchronized with the master clock before being used by the rest of the system. Common examples of synchronizer requirements in computer systems, whether micro or maxi, include the following:

- *Interrupts.* An interrupt may occur at any time, and in a typical implementation the interrupt request line is generated independently of the processor clock. The processor must synchronize the interrupt request signal with its own clock.

- *DMA request, bus hold request.* Similar to interrupts.

- *Bus ready.* In a memory or I/O bus that includes devices with different speeds, a READY signal is used to terminate each bus access. In many implementations, one or more devices may generate this signal independently of the processor clock. Therefore the processor must resynchronize the READY signal with its own clock.

- *Refresh request.* Many dynamic memory systems use a one-shot (ughhh!) or other independent timing device to initiate refresh cycles at regular intervals. If the timing device is not synchronized with the processor clock, memory requests and refresh requests can occur simultaneously. An arbiter is needed to choose one or the other when this occurs.

- *Multiprocessor handshake.* In a multiprocessor or distributed processing system with independent clocks for each processor, some kind of handshake is needed to synchronize information transfers between processors. Depending on how the processors are coupled, this may occur during bus access, in a shared memory, or in dedicated I/O ports of each processor. At some point in each implementation, information from one processor must be synchronized with the clock of another.

- *Data acquisition.* Many modern test instruments use analog-to-digital converters to sample signals for further processing and display by an embedded microprocessor. The sampling clock is usually asynchronous with the microprocessor clock, yet the sampled data must be brought into synchronization with the microprocessor clock for processing. This is particularly difficult to accomplish in high-speed applications, such as the new generation of 200–300 MHz sampling oscilloscopes and logic analyzers.

- *Multiple time bases.* In general, any system that has multiple subsystems running with different time bases needs synchronizers to accomplish communication between subsystems.

Even in systems with a single time base, synchronizers may be needed because of the unpredictability of signal propagation delays of slower parts with respect to high-speed clocks. For example, suppose a system has a master clock frequency of 15 MHz which is applied to a single-chip microcomputer and to various PAL-based state machines. Suppose a particular state machine is looking for a transition in the microcomputer's READ output signal, which is spec'ed to change between 0 and 80 ns after a rising edge of the 15 MHz clock input. Even though transitions of READ are synchronous with the 15 MHz clock, the delays are so long
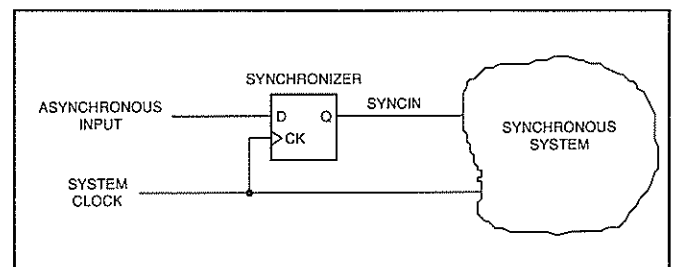

Figure 1. A single, simple synchronizer

that the transitions might occur anywhere in the 15 MHz clock period, and must therefore be treated as if they were asynchronous with it. Unwittingly treating READ as if it were synchronous is a particularly insidious bug, because systems that work perfectly with one batch of parts may fail intermittently with a faster or slower batch of parts.

Synchronizers and Metastability

Ignoring the problem of metastability, a synchronizer is very easy to build. Most digital designers are
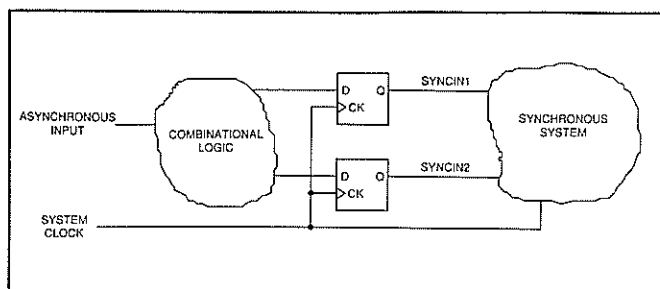
Figure 2. Combinational logic can hide the fact that there are two synchronizers.

aware that an asynchronous input should be synchronized to the system clock *at a single point*, as shown in Figure 1. If two or more synchronizers are used for a single signal, as shown Figure 2, then the system can fail by going into an inconsistent state when input timing is such that one synchronizer treats the asynchronous input as a 0 and the other treats it as a 1.

Unfortunately, the even the synchronizer in Figure 1 sometimes fails. The reason it fails is that the setup and hold times of the D flip-flop are sometimes violated. "Well, who cares?" you may say, "If the D input changes at the same time as the clock, then the flip-flop will either see the change this time or miss it and pick it up next time; either way is good enough for me!" The problem is, there is a third possibility.

When the setup and hold times of a flip-flop are not met, the flip-flop may go into a third, *metastable* state halfway between 0 and 1. Worse, the length of time it may stay in this state before falling back into a legitimate 0 or 1 state is theoretically unbounded. Other gates and flip-flops, when presented with a metastable input, may produce metastable outputs (after all, our parts are now operating in the *linear* part of their operating range). Luckily, the probability of a flip-flop output remaining in the metastable state decreases exponentially with time, though never all the way to zero.

"Synchronizer failure" is said to occur if the system uses a synchronizer output while the output is still in the metastable state. So a digital designer can build a reliable synchronizer by ensuring that the system waits "long enough" before using a synchronizer's output, "long enough" that the mean time between synchronizer failures is several orders of magnitude longer than the designer's expected length of employment.

Commercial Synchronizer Failures

Many designers are not aware of the problem of metastability, and among those who are aware, few understand its causes and cures. As a result, several products have been released that fail intermittently because of synchronizer problems.

My first encounter with the problem was in the mid-70s—many different samples of a large computer system inexplicably suffered corruption of memory words about once a month. The problem was traced to the refresh circuit, in which an asynchronous timer was used to generate refresh requests. Occasionally, when refresh and bus requests occurred simultaneously, the arbiter took more than 40–50 ns to settle, and a half-refresh, half-read cycle occurred, corrupting the data.

Subsequently, I have seen or been told of synchronizer problems in a number of commercial LSI parts, including the AMD 9513 system timing controller (fixed in the 9513A), the AMD 9519 interrupt controller (fixed in the 9519A), the Zilog Z-80 SIO first customer silicon, the READY input on Intel's original 8085 (fixed in the 8085A), and Intel's 8202 DRAM controller (fixed in the 8202A). Most recently, a large instrument manufacturer told me that they had to redesign the front end of a 140,000-transistor custom VLSI data acquisition chip four months before a scheduled product release because of a synchronizer problem.
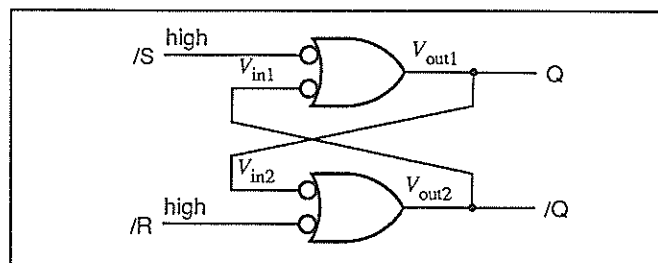

Figure 3. A pair of NAND gates forming an S-R flip-flop.

The Source of Metastability

Metastability can be observed and explained by looking at the simple S-R flip-flop shown in Figure 3. As you know, an analysis of this circuit from a strictly digital point of view shows that it has two stable states. However, the circuit has more to reveal if we consider its operation from an analog point of view.

The solid line in Figure 4 shows the steady-state (DC) transfer function T for a single NAND gate with one of its inputs tied high; the output voltage is a function of input voltage, $V_{out} = T(V_{in})$. With two NAND gates connected in a feedback loop as in Figure 3, if /S and /R are tied high we know that $V_{in1} = V_{out2}$ and $V_{in2} = V_{out1}$. Therefore, we can plot the transfer functions for both NAND gates on the same graph with an appropriate labeling of the axes. The solid line is the transfer function for the top NAND gate in Figure 3, and the dotted line is the transfer function for the bottom one.

Considering only the steady-state behavior of the S-R feedback loop, and not dynamic effects, the loop is in equilibrium if the input and output voltages of both
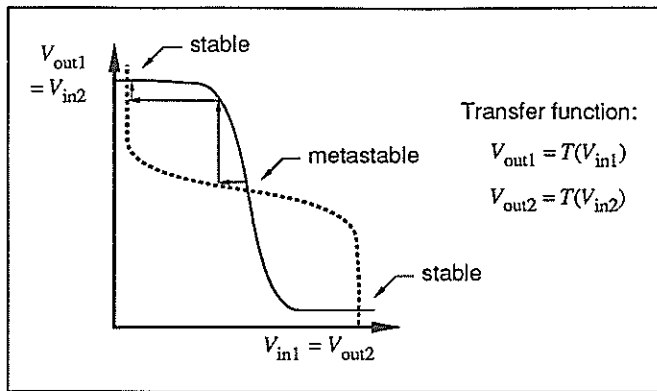
Figure 4. Transfer functions for NAND gates in S-R flip-flop.

Figure 5. Ball-and-hill analogy for metastability.

inverters are constant DC values consistent with the loop connection and the inverter DC transfer function. That is, we must have

$$V_{in1} = V_{out2} = T(V_{in2}) = T(V_{out1}) = T(T(V_{in1}))$$

Likewise, we must have

$$V_{in2} = T(T(V_{in2}))$$

We can find these equilibrium points graphically from Figure 4; they are the points at which the two transfer curves meet. Surprisingly, we find that there are not two but *three* equilibrium points. Two of them, labeled "stable," correspond to the two states that a "strictly digital" analysis identifies, with Q either 0 or 1. The third equilibrium point, labeled "metastable," has V_{out1} and V_{out2} sitting halfway between a valid logic 1 voltage and a valid logic 0 voltage; so Q and /Q are not valid logic signals at this point. Yet the loop equations are satisfied; if we can get the flip-flop to operate at the metastable point, it could theoretically stay there indefinitely.

Closer analysis of the situation at the metastable point shows that it is aptly named. It is not truly stable because random noise will tend to drive a circuit which is operating at the metastable point towards one of the stable operating points as we'll now demonstrate. Suppose the S-R flip-flop is operating precisely at the metastable point in Figure 4. Now let's assume that a small amount of circuit noise reduces V_{in1} by a tiny amount. This tiny change causes V_{out1} to *increase* by a small amount. But since V_{out1} produces V_{in2}, we can follow the first horizontal arrow from near the metastable point to the second transfer characteristic, which now demands a lower voltage for V_{out2}, which is V_{in1}. Now we're back where we started, except we have a much larger change in voltage at V_{in1} than the original noise produced, and the operating point is still changing. This "regenerative" process continues until we reach the stable operating point at the upper left-hand corner of Figure 4. However, if we perform a "noise" analysis for either of the stable operating
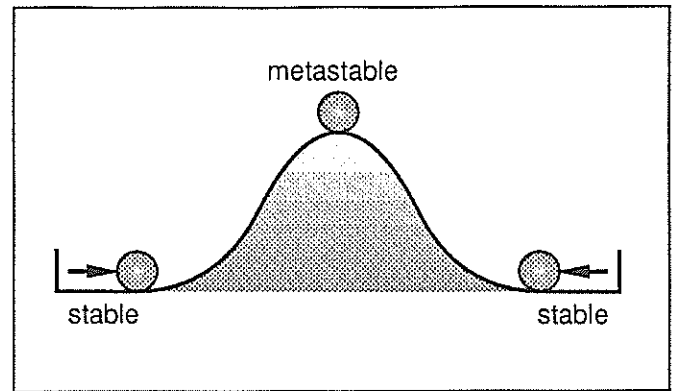
points, we find that feedback brings the circuit back towards the stable operating point, rather than away from it.

The operation of the S-R flip-flop can be likened to a perfectly smooth ball and hill, as shown in Figure 5. If we drop a ball from overhead, it will probably immediately roll down to one side of the hill or the other. But if it lands right at the top, it may precariously sit there for a while before random forces (wind, rodents, earthquakes) start it rolling down the hill. Like the ball at the top of the hill, the flip-flop may stay in the metastable state for an unpredictable length of time before nondeterministically settling in to one stable state or the other.

If even the *simplest* sequential circuit is susceptible to metastable behavior, you can be sure that *all* sequential circuits are susceptible.

Returning to the ball-and-hill analogy, consider what happens if we try to kick the ball from one side of the hill to the other. Apply a strong force (Pélé), and the ball goes right over the top and lands in a stable resting place on the other side. Apply a weak force (Pee Wee Herman), and the ball falls back to its original starting place. But apply a wishy-washy force (Charlie Brown), and the ball goes to the top of the hill, teeters, and eventually falls back to one side or the other.

This behavior is completely analogous to what happens to flip-flops under marginal triggering conditions. In S-R flip flops, a minimum pulse width is specified for the S and R inputs. Apply a pulse of this width or longer to the S input, and the flip-flop immediately goes to the 1 state. Apply a very short pulse, and the flip-flop stays in the 0 state. Apply a pulse just under the minimum width, and the flip-flop may go into the metastable state. Once the flip-flop is in the metastable state, its operation depends on "the shape of its hill." Flip-flops built from high gain, fast technologies have "sharper hilltops" and tend to come out of metastability faster than ones built from low performance technologies.

Table 1. Events that trigger metastability in common flip-flops.

Flip-flop type	Examples	Metastability triggers
S-R flip-flop	Cross-coupled 74x00, 74x279	1. S or R pulse width shorter than two gate delays 2. S and R negated simultaneously (within two gate delays of each other)
D latch	74x75, 74x373	1. Enable (G) pulse width too short 2. Setup or hold time of data (D) inputs with respect to G violated
D edge-triggered	74x74, 74x374	1. Clock (CLK) period too short 2. Setup or hold time of data (D) inputs with respect to CLK violated 3. PR or CLR pulse width too short 4. PR and CLR negated simultaneously

Metastability Triggers

Table 1 lists the ways that several common flip-flop types can get into the metastable state. In addition, power supply disturbances can cause metastability. Recent research has shown that power supply disturbances can cause intermittent failures in digital circuits, not by reducing DC noise margins as is commonly believed, but by lengthening the propagation delays of logic gates. Obviously, longer propagation delays may cause violations of flip-flop setup and hold times, and thus metastability may occur.

There are two ways to get a flip-flop out of the metastable state:

(a) Force the flip-flop into a valid logic state using input signals that meet the published specifications for minimum pulse width, setup time, and so on.

(b) Wait "long enough," until the flip-flop comes out of metastability on its own.

Inexperienced designers often attempt to get around metastability in other ways, and they are usually un-successful. Figure 6 shows an attempt by the same sort of designer who hangs capacitors on gate outputs to suppress decoding spikes and other logic hazards. Rather than eliminate metastability, this circuit enhances it — with the "right" components, the circuit will oscillate forever once it is excited by negating /S and /R simultaneously.

By the way, flip-flops in some older, slower technologies may exhibit metastability by oscillating, even without extra load capacitance of Figure 6. However, fast technologies tend to exhibit metastability by quietly sitting in the middle of the linear region, halfway between logic 0 and logic 1, as we've been discussing.

Figure 7 shows a valiant but also failed attempt to eliminate metastability within one period of a system clock. Circuit M is a memoryless analog voltage detector whose output is 1 if /Q is in the metastable state, 0 otherwise. The circuit designer's idea was that if line /Q is detected to be in the metastable state when CLOCK goes low, the NAND gate will clear the D flip-flop, which in turn eliminates the metastable output, causing a 0 output from circuit M and thus deactivating the CLR
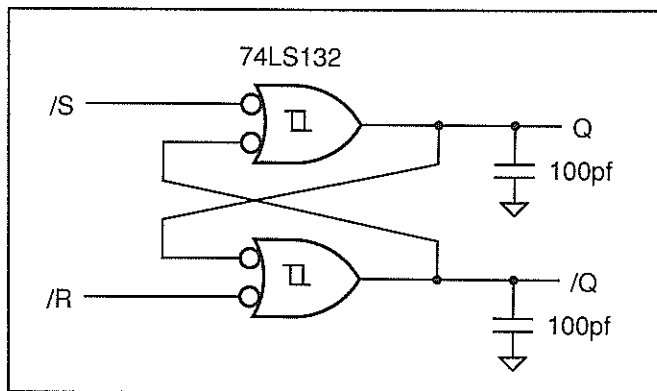

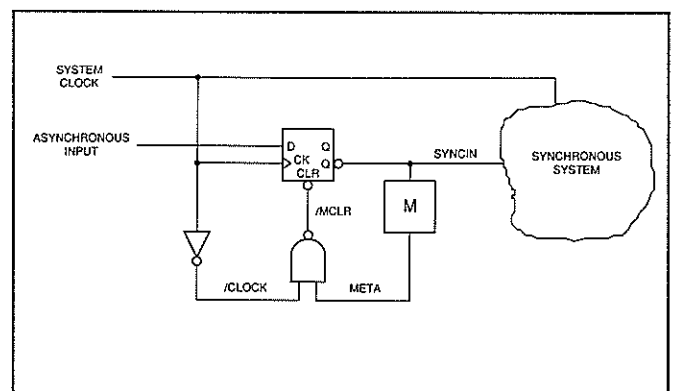Figure 6. A failed attempt to eliminate metastability.

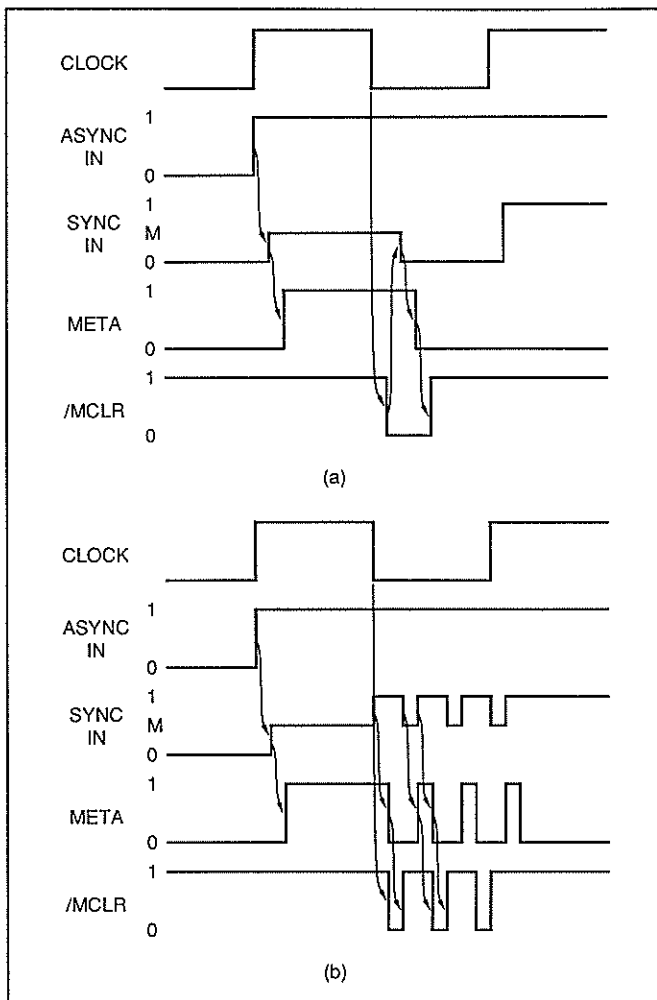Figure 7. A metastability "eliminator" with a subtle failure.

Figure 8. Timing waveforms for the metastability "eliminator":
(a) expected (b) possible.

input of the flip-flop. The circuits are fast enough that this all happens well before CLOCK goes high again; the expected waveforms are shown in Figure 8.

Unfortunately, quite often the metastable output will resolve on its own. In particular, suppose it settles to the 1 state as shown in Figure 8(b), just as CLOCK goes low and the NAND gate is about to clear the flip-flop. If the timing is right, this can cause a narrow pulse on the CLR input, driving the flip-flop right back into the metastable state! Even though circuit M will be activated again and attempt to bring the flip-flop out of metastability a second time, under the right timing conditions the same sort of behavior can be repeated, with the circuit popping in and out of metastability well into the next clock period.

What's to be done?

These examples should give you the sense that synchronizer problems can be very subtle, so you must be

careful. The only way to make synchronizers reliable is to wait long enough for metastable outputs to resolve.

But how long is "long enough"? Next month, in the conclusion of this article, we'll give some practical guidelines and show how to calculate the probability of synchronizer failure from published studies. We'll also give some practical design examples for Intel 80386 systems, show the pitfalls of scaling designs for higher speed operation (such as changing an Intel 80186 system from 8 MHz to 16 MHz operation), and provide a list of references for additional data.

In The News
Continued from page 3

compatible is the wording they're after. In the case of AST, IBM is also complaining about the use of "/2" in Rampage/2 and Advantage/2. This seems to be taking things a bit far.

32-bit Microcontrollers at Intel?
In a recent employment ad, Intel lists a position for a design engineer to join a team developing "a new family of advanced 32-bit CMOS microcomputers/microcontrollers." Are 32-bit single-chip micros on their way? (And who needs them?)

Micro Channel Support Chips Expected
Chips and Technologies and Faraday have already announced their intentions to provide ICs for Micro Channel support. With the AT bus, most support ICs are designed for system or video boards. The Micro Channel, however, requires more logic on peripheral cards and provides less space. So, expect to see Micro Channel interface chips for peripheral boards from all the usual suppliers, plus some new ones (such as National, and perhaps Intel). An open question is just what IBM's pending patents cover, and whether this will limit what will be offered.

Intel has lost much business to Chips and Technologies, since the C & T chip set eliminates the need for many of Intel's peripheral chips. It seems likely that Intel will take advantage of the change in architecture from the AT bus to the Micro Channel to try to regain some of that business.

Clipper Support in Question
Users of Fairchild's Clipper may be in for some nervous times. National, which recently announced that they would acquire Fairchild, is believed to have little interest in Clipper. Since Clipper is sole-sourced, this could mean trouble for the 50 or so system designers Fairchild claims are building systems around it. I'd sure think twice (or three times) before adopting it now.

Implementing Programmable Option Select on IBM's PS/2

By John Figueroa, On Target Associates

The Micro Channel bus used in the PS/2 Models 50, 60, and 80 is a significant departure from IBM's XT and AT buses. This is the first of a series of articles that will explore the new aspects of the Micro Channel. This article concentrates on the *Programmable Option Select (POS)* functions, which must be implemented on all Micro Channel cards.

Unlike the earlier buses, the Micro Channel includes a "card select" signal (–CD SETUP) for use during setup that is unique on each slot. This allows the CPU to access each *adapter* (IBM's term for an add-in card) for setup purposes without requiring any unique address decoding on the adapter itself.

Switches for adapter configuration will become a thing of the past, but not without some new problems for the adapter designer. Implementing Programmable Option Select requires additional PC board space, which is already at a premium due to the relatively small size of the Micro Channel cards.

Programmable Option Select Operation

IBM eliminates hardware switches in its new generation of PS/2 machines with hardware registers and standardized *Adapter Description Files*, which have the extension ".ADF." The power-on self-test

Address (hex)	Function
0091	Card Selected Feedback Register
	Bit 0 = Adapter is responding
0094	System Board Enable/Setup Register
0096	Adapter Enable/Setup Register
0100	Adapter ID low byte
0101	Adapter ID high byte
0102	POS[0] - Option Select Data
	Bit 0 = Card Enable
0103	POS[1] - Option Select Data
0104	POS[2] - Option Select Data
	Bit 7 = Channel Check
	Bit 6 = Channel Check Status Indicator
0105	POS[3] - Option Select Data
0106	POS[4] - Subaddress Extension low byte
0107	POS[5] - Subaddress Extension high byte

Table 1. POS Register Addresses.

(POST) routines compare the adapters in the system with information stored in CMOS RAM to determine if any changes have been made to the system. Changes detected by the POST invoke the auto-configuration utility.

For each adapter card there is a corresponding ADF file, whose name corresponds to the card's adapter ID. (For example, "@7E7E.ADF" is the file for use with the adapter whose ID is 7E7E hex.) The configuration utility (supplied by IBM) prompts the user in configuring the adapter, using the text provided in the ADF files. The configuration utility then sets the hardware registers as specified in the ADF file for the selections the user has entered in response to the prompts. The data written to the configuration registers is also stored in CMOS RAM so the adapter can be initialized during subsequent power-ups. IBM cautions against any direct setting of the POS registers or direct access to the CMOS RAM.

POS Hardware

The POS interface consists of:

- A 16-bit adapter ID code (read-only)

- Up to 32 bits of read/write adapter setup data (with 3 bits reserved for control functions)

- Up to 16 bits for subaddressing (write-only) or error status (read-only)

Table 1 shows the registers used for POS. The system board and all adapters share addresses 0100-0107 hex during setup. The Adapter Setup/Enable Register

---

Bus Overview

- Address bus A0–A23

- Address bus A24–A31 (model 80 only)

- Data bus D0–D15

- Data bus D16–D31 (model 80 only)

- 16 bit I/O address with support for 8- to 32-bit I/O transfers

- Arbitration control bus for up to 15 devices

- Level sensitive interrupts (as opposed to the edge sensitive type in the PC family)

- POS (Programmable Option Select) to eliminate adapter switches

- Advanced DMA support for 8 devices

- Enhanced hardware error management

- Additional ground pins distributed throughout bus for noise reduction

- 50-mil connector spacing for higher density

- Smaller card size — 33 sq. in. vs. 48 for AT

at address 0096 controls which slot's –CD SETUP signal will be asserted during accesses to the shared addresses. Only the adapter whose –CD SETUP signal is asserted should respond to these addresses.

For adapters that require access to more than these few registers for configuration, a technique called *sub-addressing* is used. This allows memory on the adapter to be initialized by the POS routines. POS registers 6 and 7 are written with the address of the memory location to be accessed, which is then read or written via the general-purpose POS register locations.

Figure 1 shows a schematic for the basic POS circuitry required on an adapter card. The Micro Channel does not include the usual READ and WRITE signals; the type of transfer is indicated by the –S0, –S1, and M/–IO signals, and the timing is controlled by –CMD. Address and transfer type signals are not stable throughout the cycle, and are latched at the trailing edge of –ADL. The 74LS155 dual decoder provides read and write strobes for the POS registers. Note that it is not necessary to decode any higher-order address bits, since –CD SETUP is asserted only for accesses in the range of 0100 to 0107, and only to the card that has been previously selected via port 0096 on the system board. Read-back input ports allow the card's configuration to be detected by driver software.

0000	Device not ready
0001-0FFF	Bus master
5000-5FFF	DMA devices
6000-6FFF	Direct program control or memory mapped I/O
7000-7FFF	Storage or multifunction cards
8000-8FFF	Video

Table 2. IBM Guidelines for Adapter ID Numbers.

Bit 0 of POS[0] acts as the card enable control. Adapters should not respond to any addresses (except in setup mode) or generate any bus requests or interrupts until this bit has been set by the setup software.

This basic schematic does not include the logic required for channel check or subaddressing. When a channel check error occurs on the bus, the error handling routine polls bit 7 of POS[2] on each adapter to determine which adapter produced the error. Bit 6 of POS[2], when set to 0 by the adapter, indicates that status information describing the reason for the channel check error is available from POS[4] and POS[5].
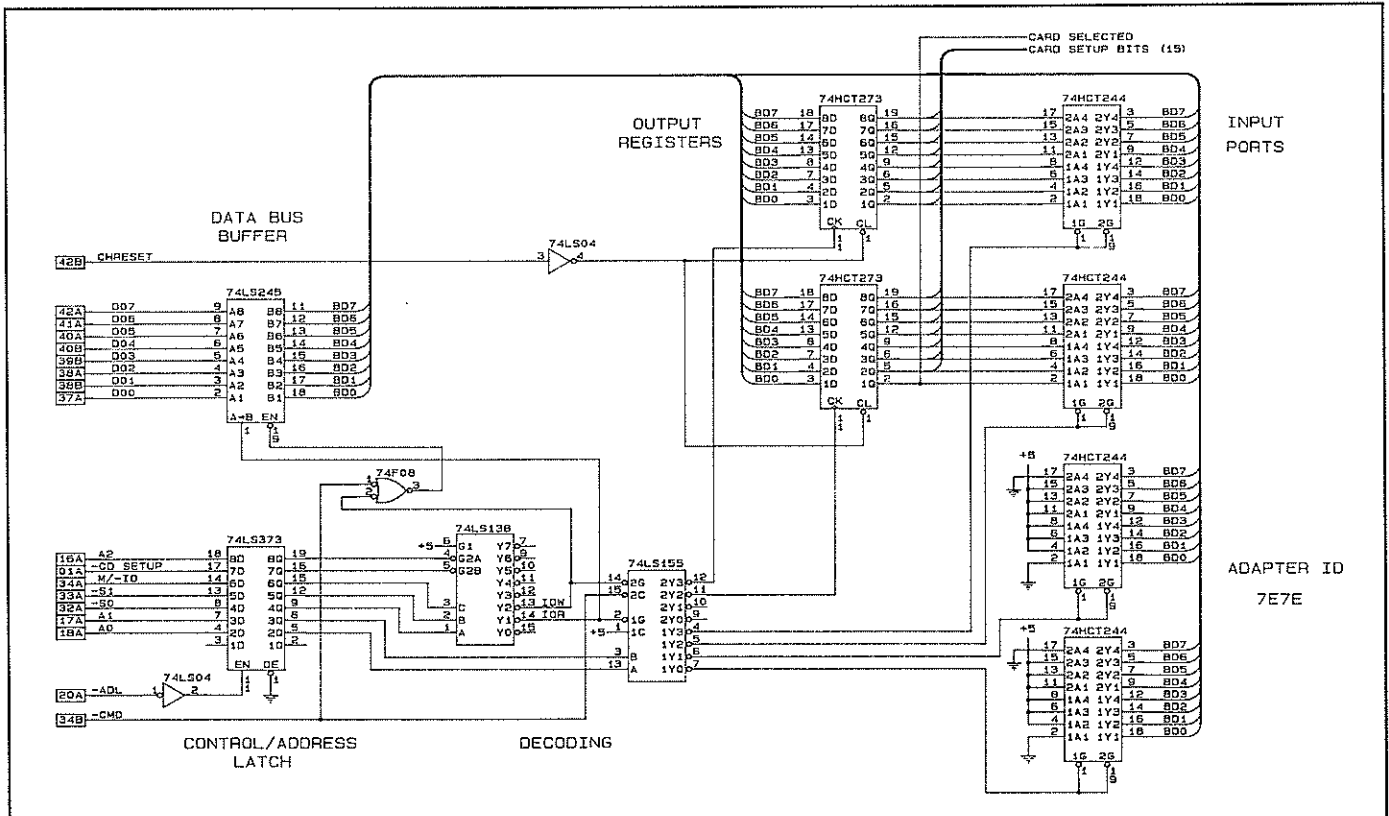

Figure 1. Basic POS Circuitry for Adapter.

```
; A simple adapter description file (@7E7E.adf)
;
; Lines beginning with semicolons are comments
;
AdapterId 07E7Eh
AdapterName "ON Target Asso. PS Protocard"
NumBytes  2
;
; NumBytes specifies number of POS registers - 4 max
;
FixedResources POS[1]=10100101b
;
; Initializes IO reg. 0103h with arbitrary pattern A5 hex
;
NamedItem
Prompt "On boot-up Port 0102h will be loaded with"
Choice "the byte 02h" POS[0]=0000001Xb
Choice "the byte 04h" POS[0]=0000010Xb
Choice "the byte 08h" POS[0]=0000100Xb
Choice "the byte 10h" POS[0]=0001000Xb
Choice "the byte 20h" POS[0]=0010000Xb
Choice "the byte 40h" POS[0]=0100000Xb
Choice "the byte 80h" POS[0]=1000000Xb
;
Help
"This demo can be used to set port values to
control your adapter circuitry.
This help message must be less than 1000 bytes."
```

Listing 1. Example ADF File.

Listing 1 shows an example of an adapter description file. The AdapterId, AdapterName, and NumBytes keywords are required; all others are optional. The FixedResources statement specifies control settings and resource requirements that are independent of the user-selected configuration, while the Choice statements allow control settings and resource allocation to vary depending on user selections during the configuration process.

For each of these statements there may be a "POS setting" and/or a "resource setting." A POS setting is simply a value to be written to the specified POS register. A resource setting may specify a range of I/O addresses, a range of memory addresses, a list of interrupt levels, or a list of arbitration levels.

The NamedItem keyword begins a block of statements used to prompt the user to make a selection of some operating parameter. The Prompt statement provides a string to be displayed to indicate the parameter being selected. The Choice statements provide a string to be displayed to the user to describe the selection, and a POS setting and/or a resource setting to be set if the user selects that choice. The example shown in the listing is designed for use by a developer, who may want to write specific bit patterns to a POS register to test the hardware. A more typical sequence, for an adapter such as a serial port, would be as follows:

Adapter IDs

IBM is now providing adapter ID numbers to registered developers. Table 2 shows the guidelines IBM has provided for adapter number assignments.

IBM has published the following adapter ID numbers for their products:

ESDI Drive adapter	DDFF
PC network II/A	EFEF
Fixed Drive adapter	DFFD
Modem Adapter	EDFF
Multiprotocol	DEFF
Memory Expansion	FEFE and FAFF

Adapter Description Files

The adapter developer must create an adapter description file (in ASCII format) to be provided with the adapter. Each PS/2 machine is shipped with a "Reference" disk, which contains the system configuration utilities. The purchaser of an adapter card uses the "Copy an Option Diskette" utility to merge the ADF file for each adapter onto the reference disk. Thus, the reference disk will have all the information needed to configure the system, even if third-party adapters have been installed.

Resources

- The following publications are available from IBM, and include descriptions of the Micro Channel Signals, POS operation, and bus timing. Unlike the XT and AT manuals, schematics are not provided, but timing specifications are. These manuals may be ordered by calling (800) IBM-PCTB. Price for each is $125.

   Personal System/2 Models 50 and 60 Technical Reference, Part Number 68X2224

   Personal System/2 Model 80 Technical Reference, Part Number 68X2256

- IBM also provides seminars and seminar notebooks for registered developers. You must be a registered developer to be assigned an adapter ID number. For information call the IBM Independent Developer Assistance Program (IDAP) at (305) 241-5455.

- The August issue of *PC Tech Journal* includes several articles on the PS/2 machines, including a good overview of the Micro Channel ("An Architecture Redefined," pg. 58).

- Prototyping and extender cards for the Micro Channel are available from On Target Associates, (408) 980-7118. Proto cards include bus documentation, and start at $99.

Megabit Dynamic RAMs

Vendor	Page Mode	Nibble Mode	Static Column	Max Speed (ns)	Comments
Fujitsu	MB81C1000	MB81C1001	MB81C1002	100	In production; 80 ns in 4Q87
Hitachi	HM511000	HM511001	HM511002	100	In production; 80 ns in 4Q87
Micron Tech	MT4C1024	MT4C1025	MT4C1026	100	Sampling 150 ns; Production Jan. '88; projecting 80 ns mid-88
Mitsubishi	M5M4C1000	M5M4C1001	M5M4C1002	100	In production; 80 ns in 4Q87
Motorola	MCM511000	MCM511001	MCM511002	85	Available 4Q87; will use Toshiba dice
NEC	μPD421000	μPD421001	μPD421002	80	Now sampling; production 4Q87
Oki	MSM511000	MSM511001	MSM511002	100	100 and 120 ns in production; 80 ns in 4Q87; NMOS parts MSM41100x also available
Siemens	HYB511000		HYB511002	100	Available in first half '88
TI	TMS4C1024	TMS4C1025	TMS4C1027	100	No ZIP Packages; available now direct; distrib. stocking in Oct.-Nov.
Toshiba	TC511000	TC511001	TC511002	85	Volume leader; lowest FMV; 70 ns 4Q87; Embargo coming?
Vitelic	V51C100		V51C102	80	Sampling later this month; Production in 4Q87

One megabit dynamic RAMs are now available in volume from several vendors, with many more due to start shipping by the end of the year. Pricing is still high enough that, on a cost-per-bit basis, 256K-bit chips are significantly less expensive. For applications requiring a megabyte or more, megabit RAMs nevertheless deserve serious consideration.

The most common, least-expensive megabit RAMs are organized 1M × 1. The smallest 16-bit-wide memory that can be built with 1M × 1 chips is 2M bytes, and the smallest 32-bit-wide memory is 4M bytes. Many applications do not need this much memory, particularly in their basic configuration. Thus, 256K RAMs will continue to be useful even after the price-per-bit of megabit RAMs drops below that of 256K RAMs.

Another alternative for these applications is the 256K × 4 version, which allows a 16-bit-wide memory as small as 512K bytes (or 32-bit-wide as small as 1M byte) to be built. This may be the best of both worlds for many applications, provided that the × 4 devices don't carry too large a price premium.

Access Modes

Most manufacturers are providing three data access options for 1M × 1 chips — page mode, nibble mode, and static column mode. Page mode, which allows the column address to be changed without modifying the row address, is by far the most common. Most vendors are using what they call fast page mode, which is simply page mode with a faster column access time; this is also called ripple mode.

Nibble mode allows four successive locations to be accessed by pulsing CAS. It is less flexible than page mode, but is easier to use since no address is required for the additional CAS pulses.

Static-column devices allow the column address to be changed without pulsing CAS. Compaq is using 256K-bit static column devices in their Deskpro 386.

All these different access modes are generally obtained from the same die, using bonding options. Thus, the IC manufacturer produces only one chip, which is packaged and marked in a variety of ways.

Packaging Options

The dominance of the DIP package is beginning to end. Most vendors supply 1M × 1 RAMs in 18-pin DIPs, 20-pin SOJs, and 20-pin ZIPs. The small-outline J-leaded (SOJ) package is in fact a 26-pin package with the center three pins missing on each side, and is designed for surface-mount applications.

The highest possible density is obtained by using the SOJ package and mounting chips on both sides of the board. Many systems manufacturers are only beginning to use surface-mount technology, however, and mounting chips on both sides is especially difficult.

The zig-zag in-line package (ZIP) has a single row of 20 pins on 50-mil centers. The leads are formed in an alternating zig-zag pattern to make two rows of 10 pins each, all on 100-mil centers. This package provides higher density than the SOJ (assuming components are placed on only one side of the board), and is compatible with conventional pin-in-hole technology.

Yet another option is the single in-line memory module (SIMM), which consists of eight or nine $1M \times 1$ SOJ chips mounted on a small PC board. SIMMs are available either with pins or with a PC-edge-type connector. The edge-type connector is most popular, as it allows the SIMMs to be easily socketed. SIMMs were pioneered by TI, and are now produced by most vendors.

SIMMs provide very high density on the surface of the PC board, at the expense of some additional height. They allow the systems manufacturer to achieve surface-mount density without having to use surface-mount fabrication techniques themselves. Like ZIPs, SIMMs make trace routing very easy. The resulting short PC traces, along with the SIMMs' on-board bypass capacitors, minimize noise problems.

SIMMs have been very popular in personal computers. IBM uses proprietary SIMMs in their PS/2 machines, the first volume application of megabit RAMs. Apple currently uses 256K-byte SIMMs in the Macintosh II, but the design is also compatible with 1M-byte SIMMs to allow upgrading of memory capacity.

One problem for PC add-in board designers is that 1M-byte SIMMs are taller than 256K-byte SIMMs. To meet the component height limit for an add-in board, the 1M SIMMs must be mounted at an angle, which increases the PC area required.

Pricing

One-megabit RAMs remain considerably more expensive than 256K-bit devices on a cost-per-bit basis. Pricing has been heavily influenced by the FMVs (fair market values) set by the US Department of Commerce, since nearly all megabit dynamic RAMs come from Japan. Fujitsu has been saddled with high FMVs, which has made it difficult for them to compete for orders in the U.S. Toshiba was the first volume producer, has the lowest FMVs, and is currently the dominant supplier. The threatened embargo against Toshiba could have a major effect on the DRAM market, but most observers believe that DRAMs will not be embargoed due to the large impact such an action would have on U.S. systems manufacturers.

Low-volume prices are commonly in the mid-twenties for 100-ns $1M \times 1$ parts, though some gray-market dealers (such as Fry's in Sunnyvale) have been selling single units for as low as $17.99. Volume buyers are rumored to be paying prices ranging from $13 to just over $20. Prices are expected to fall into this range for all buyers by the end of the year.

Prices for $256K \times 4$ devices are 10% to 15% higher than for $1M \times 1$ parts. The SIMMs carry a similar price premium — a $1M \times 8$ SIMM sells for about the same price as 9 separate RAMs.

The price of megabit RAMs is not expected to drop as rapidly as did the price of 256K RAMs. There is a substantial capital investment in production equipment needed to produce these devices, and getting a return on this investment requires keeping the prices from falling too rapidly. There are also more pricing controls in place, both from MITI within Japan and from the FMVs in the U.S.

Access Speeds and Vendor Compatibility

Despite the diversity of access modes and package types, there is a surprising degree of compatibility among vendors. Timing is very similar for devices of a given access time. Not all vendors are shipping parts at the fast end of the spectrum, however. Most vendors are now shipping 100 ns or 120 ns parts and planning to ship 80 ns parts by the end of the year. Toshiba, having the advantage of being further down the learning curve than most other vendors, is now shipping 85 ns parts and expects to have 70 ns parts by year end.

Pin compatibility is nearly universal. The only variation occurs on pin 4, which enables a test mode. This pin is not used in normal applications, so any changes affect only those large end-users that perform incoming test.

Virtually all parts are now, or soon will be, in CMOS technology. Some vendors, such as OKI, began with NMOS and are now switching to CMOS; others, such as Toshiba, have used CMOS from the beginning.

Part numbering is reasonably consistent, with a few exceptions. The prefix, of course, varies according to the manufacturer; for example, Toshiba uses TC and Mitsubishi uses M5M. The first digits also vary; Toshiba uses 51, while Mitsubishi uses 4C. But for the last four digits, most vendors use the same scheme — 1000 for page-mode, 1001 for nibble mode, and 1002 for static-column (all for $1M \times 1$ devices). Some U.S. vendors, such as Micron Technology and TI, use 1024, 1025, and 1026 for the last four digits for page, nibble, and static column versions. The $256K \times 4$ devices typically use 4256 for the last four digits of the page-mode version and 4258 for the static-column version. Only Fujitsu is making a nibble-mode $256K \times 4$ part, with the 4257 designation.

The highest possible density is obtained by using the SOJ package and mounting chips on both sides of the board. Many systems manufacturers are only beginning to use surface-mount technology, however, and mounting chips on both sides is especially difficult.

The zig-zag in-line package (ZIP) has a single row of 20 pins on 50-mil centers. The leads are formed in an alternating zig-zag pattern to make two rows of 10 pins each, all on 100-mil centers. This package provides higher density than the SOJ (assuming components are placed on only one side of the board), and is compatible with conventional pin-in-hole technology.

Yet another option is the single in-line memory module (SIMM), which consists of eight or nine $1M \times 1$ SOJ chips mounted on a small PC board. SIMMs are available either with pins or with a PC-edge-type connector. The edge-type connector is most popular, as it allows the SIMMs to be easily socketed. SIMMs were pioneered by TI, and are now produced by most vendors.

SIMMs provide very high density on the surface of the PC board, at the expense of some additional height. They allow the systems manufacturer to achieve surface-mount density without having to use surface-mount fabrication techniques themselves. Like ZIPs, SIMMs make trace routing very easy. The resulting short PC traces, along with the SIMMs' on-board bypass capacitors, minimize noise problems.

SIMMs have been very popular in personal computers. IBM uses proprietary SIMMs in their PS/2 machines, the first volume application of megabit RAMs. Apple currently uses 256K-byte SIMMs in the Macintosh II, but the design is also compatible with 1M-byte SIMMs to allow upgrading of memory capacity.

One problem for PC add-in board designers is that 1M-byte SIMMs are taller than 256K-byte SIMMs. To meet the component height limit for an add-in board, the 1M SIMMs must be mounted at an angle, which increases the PC area required.

Pricing

One-megabit RAMs remain considerably more expensive than 256K-bit devices on a cost-per-bit basis. Pricing has been heavily influenced by the FMVs (fair market values) set by the US Department of Commerce, since nearly all megabit dynamic RAMs come from Japan. Fujitsu has been saddled with high FMVs, which has made it difficult for them to compete for orders in the U.S. Toshiba was the first volume producer, has the lowest FMVs, and is currently the dominant supplier. The threatened embargo against Toshiba could have a major effect on the DRAM market, but most observers believe that DRAMs will not be embargoed due to the large impact such an action would have on U.S. systems manufacturers.

Low-volume prices are commonly in the mid-twenties for 100-ns $1M \times 1$ parts, though some gray-market dealers (such as Fry's in Sunnyvale) have been selling single units for as low as $17.99. Volume buyers are rumored to be paying prices ranging from $13 to just over $20. Prices are expected to fall into this range for all buyers by the end of the year.

Prices for $256K \times 4$ devices are 10% to 15% higher than for $1M \times 1$ parts. The SIMMs carry a similar price premium — a $1M \times 8$ SIMM sells for about the same price as 9 separate RAMs.

The price of megabit RAMs is not expected to drop as rapidly as did the price of 256K RAMs. There is a substantial capital investment in production equipment needed to produce these devices, and getting a return on this investment requires keeping the prices from falling too rapidly. There are also more pricing controls in place, both from MITI within Japan and from the FMVs in the U.S.

Access Speeds and Vendor Compatibility

Despite the diversity of access modes and package types, there is a surprising degree of compatibility among vendors. Timing is very similar for devices of a given access time. Not all vendors are shipping parts at the fast end of the spectrum, however. Most vendors are now shipping 100 ns or 120 ns parts and planning to ship 80 ns parts by the end of the year. Toshiba, having the advantage of being further down the learning curve than most other vendors, is now shipping 85 ns parts and expects to have 70 ns parts by year end.

Pin compatibility is nearly universal. The only variation occurs on pin 4, which enables a test mode. This pin is not used in normal applications, so any changes affect only those large end-users that perform incoming test.

Virtually all parts are now, or soon will be, in CMOS technology. Some vendors, such as OKI, began with NMOS and are now switching to CMOS; others, such as Toshiba, have used CMOS from the beginning.

Part numbering is reasonably consistent, with a few exceptions. The prefix, of course, varies according to the manufacturer; for example, Toshiba uses TC and Mitsubishi uses M5M. The first digits also vary; Toshiba uses 51, while Mitsubishi uses 4C. But for the last four digits, most vendors use the same scheme — 1000 for page-mode, 1001 for nibble mode, and 1002 for static-column (all for $1M \times 1$ devices). Some U.S. vendors, such as Micron Technology and TI, use 1024, 1025, and 1026 for the last four digits for page, nibble, and static column versions. The $256K \times 4$ devices typically use 4256 for the last four digits of the page-mode version and 4258 for the static-column version. Only Fujitsu is making a nibble-mode $256K \times 4$ part, with the 4257 designation.

Recent IC Announcements

Part Number	Vendor	Description	Price	Availability
CDP68HC68R1/R2	GE Solid State	1-Kbit (R1) and 2-Kbit (R2) CMOS static RAMs with serial interface. 8-pin DIP package.	$1.38/1000 (R1) $1.85/1000 (R2)	Production now
Bt456	Brooktree	"RAMDAC" color palette IC. Includes 256 x 12 lookup table and three 4-bit DACs. Pixel rates up to 66 MHz.	$30/100	Sampling
FE3500	Faraday	5-chip set for AT-compatible system boards. Operates at speeds up to 12.5 MHz. Supports software switching of clock rate, and operation of peripheral bus at slower speed than processor.	$95/1000	Sampling "to selected customers"
FE2400	Faraday	2-chip set for 8086-based XT-compatible system boards. Operates at clock rates up to 9.54 MHz.	$47/1000	Sampling
80286-16	AMD	Fast 80286 microprocessor	$150/500 (LCC) $175/500 (PGA)	Sampling
80C186	Intel	CMOS version of 80186. In addition to lower power, adds dynamic RAM refresh support and coprocessor interface logic.	$18/1000	Limited sampling
DS1287	Dallas Semi	Real-time clock calendar chip with quartz crystal, lithium battery, and power up/down control in the same package. Pin- and software-compatible with Moto 146818.	$15/1000	Production now
ET3000	Tseng Labs	Color graphics controller, emulates IBM VGA, EGA, MDA.	$45/5000 ceramic $39/5000 plastic	Samples mid-Sept; Ceramic production late Oct.; Plastic 1Q88
PVGA1	Paradise Systems	Color graphics controller, emulates IBM VGA, EGA, MDA.	$60/100	Samples in October
CS8245	Chips & Technologies	Color graphics controller, 2-chip set, emulates IBM VGA, EGA, MDA.	$40.50/1000	Sampling; production this month
MB834100	Fujitsu	4M-bit mask-programmed ROM, 256K x 16, 250 ns.	$15/1000	Production now
UCN-5810/5812/5818	Sprague	BiMOS drivers, with 10, 20, or 32 outputs, designed to drive VF displays. Serial interface to processor. Output voltage to 60 V.	$1.56 to $5.22/100	Sampling; production 12 to 14 wks ARO
WD42C22	Western Digital	Disk controller with integrated buffer manager and error correction, designed for PC-type bus interface. Supports ST506, ST412, and ESDI disk interfaces, with RLL 2,7, MFM, or NRZ encoding. Controls buffer up to 32K bytes and can operate with 1:1 interleave.	under $50/1000	Sampling

Micro Channel POS
Continued from page 11

NamedItem
Prompt "Async Communications Port"
Choice "Serial 1" POS[0]=XXX0000Xb
    io 03F8h-03FFh int 4
Choice "Serial 2" POS[0]=XXX0001Xb
    io 02F8h-02FFh int 3

An X in a bit field indicates that the bit is not to be changed. Bit 0 of POS[0] is The "io" field indicates the I/O addresses to be used by the adapter, and the "int"

field indicates the interrupt level that will be used. Logic on the adapter must implement these selections according to the data written to POS[0]. When writing to POS[0], bit 0 should be shown as X to avoid affecting the CARD ENABLE signal.

The POS mechanism thus adds a design burden, but should significantly decrease system setup hassles. Support chips that provide the required logic in a single IC, along with other Micro Channel interface functions, are expected soon and will decrease the burden on the adapter designer.

Literature Watch

ASICs

Buyer's guide to ASICs and ASIC design tools. Staff; Computer Design, 8/15/87, pg 68, 50 pgs.

Mastering the maze grows harder as ASIC choices proliferate. Wilson, Ron, Senior Editor; Computer Design, 8/15/87, pg 41, 8 pgs.

Roll your own controller for serial data communications. Bursky, Dave; Electronic Design, 8/06/87, pg 47, 3 pgs.

The ASIC takeover: It's coming faster. (Special Issue, 11 articles); Electronics, 8/06/87, pg 57, 21 pgs.

Zymos tools let users customize PC-clone chips. Cole, Bernard C.; Electronics, 8/20/87, pg 85, 2 pgs.

Buses

An architecture redefined. (IBM Micro Channel) Methvin, David; PC Tech Journal, 8/87, pg 58, 11 pgs.

Control MIL-STD-1553 with remote terminal chips. Snyder, Dennis R., United Technologies Microelectronics; Electronic Design, 8/06/87, pg 109, 5 pgs.

New wave interface: a chip for VSB. Pri-Tal, Shlomo, A.J. Gregorcyk, Doug Kraft, Motorola Inc.; ESD, 8/87, pg 69, 5 pgs.

Novel logic-signal standard multiplies backplane speeds. Davis, Randall W., National Semiconductor Corp.; Electronic Design, 8/20/87, pg 81, 3.5 pgs.

PC-based GPIB control and data-acquisition products. Mosley, J. D., Regional Editor; EDN, 8/06/87, pg 94, 10 pgs.

Under the Covers. The new IBM Micro Channel as seen from inside the PS/2 Model 50. Ciarcia, Steve; Byte, 8/87, pg 101, 9 pgs.

VME debugging: a test-tool picnic. Coombs, Tim, Concise Technology; ESD, 8/87, pg 85, 3 pgs.

VMEbus gets a mid-life kicker. Manuel, Tom; Electronics, 8/20/87, pg 66, 3 pgs.

Western Digital slashes SCSI bus overhead time. Waller, Larry, Jonah McLeod; Electronics, 8/20/87, pg 64, 2 pgs.

Design Techniques

DMA controller saves bus bandwidth from drowning. Allmond, Dave, Interphase Corp.; ESD, 8/87, pg 79, 2 pgs.

Maximize performance by choosing best memory. Van Loo, William, Sun Microsystems; Computer Design, 8/01/87, pg 89, 6 pgs.

Preserve high speed in CPU-to-cache transfers. Namjoo, Masood, Anant Agrawal, Sun Microsystems, Inc.; Electronic Design, 8/20/87, pg 91, 5 pgs.

Development Tools

Full functionality, low-cost CAD/CAE systems. Eckhouse, Jr., Richard H., New Product Reviews Editor; Computer, 8/87, pg 126, 4 pgs.

Low-cost CAE software comes of age. Milne, Bob; Electronic Design, 8/20/87, pg 33, 4 pgs.

State and timing logic analysis fits on a single chip. Peters, Gregory J., Hewlett-Packard Co.; Electronic Products, 8/15/87, pg 26, 6 pgs.

Miscellaneous

Advanced CMOS logic aids high-speed designs. Nadolski, James, GE/RCA Solid State; Computer Design, 8/01/87, pg 97, 5 pgs.

Advanced CMOS logic wars escalate over ground bounce. Cormier, Denny, Senior Technical Editor; ESD, 8/87, pg 53, 3 pgs.

Just when you thought it was safe to choose an operating system... Compston, David, Plessey Microsystems; ESD, 8/87, pg 61, 2.5 pgs.

Military Electronics. Allan, Roger; Electronic Design, 8/06/87, pg 81, 11 pgs.

Technical-Article Database Index Staff; EDN, 8/06/87, pg 177, 17.5 pgs.

Peripheral Chips

Adaptec's disk controller sets SCSI speed record. McLeod, Jonah; Electronics, 8/20/87, pg 61, 3 pgs.

Graphics chip attacks frame buffer bottleneck. Myrvaagnes, Rodney; Electronic Products, 8/01/87, pg 15, 1 pg.

Inside the 82786 graphics chip. Nicholls, Bill, BGW Systems, Inc.; Byte, 8/87, pg 135, 5 pgs.

Video palette raises color ante to 1024 separate hues. Mokhoff, Nicolas; Electronic Design, 8/20/87, pg 43, 3 pgs.

Personal Computers

Models 50 and 60: the new-generation ATs. Shields, Jim; PC Tech Journal, 8/87, pg 111, 15 pgs.

Processors

Inmos puts transputers into its own CAD system. Rogerson, Steve; Electronics, 8/20/87, pg 81, 2 pgs.

Introduction to the Clipper architecture. Hunter, Colin B., Hunter Systems, Inc.; IEEE Micro, 8/87, pg 6, 21 pgs.

Memory-mapped coprocessor speeds floating-point math. Bonomi, Mauro, Christopher Tice, Weitek; EDN, 8/20/87, pg 203, 5 pgs.

PLDs and 68020 provide launchpad for 68030 designs. Ruhland, Michael, Motorola Inc.; ESD, 8/87, pg 45, 4.5 pgs.

System considerations in the design of the AM29000. Johnson, Mike, Advanced Micro Devices; IEEE Micro, 8/87, pg 28, 14 pgs.

The 80387 and its applications. Perlmutter, David, Alan Kin-Wah Yuen, Intel Corporation; IEEE Micro, 8/87, pg 42, 15.5 pgs.

The transputer finally starts living up to its claims. Manuel, Tom, Steve Rogerson; Electronics, 8/20/87, pg 78, 3 pgs.

Programmable Logic

Design bus interfaces, peripherals with programmable-logic chip. Bursky, Dave; Electronic Design, 8/20/87, pg 50, 4 pgs.

Ending tie-ups in peripheral IC design. Kopec, Stan, Thomas Darbonne, Altera Corp.; Electronic Products, 8/01/87, pg 24, 6 pgs.

Intel P9
Continued from front page

regard to personal computers. The 80186 and 80188, designed to be compatible replacements for the 8086/88 and support chips, have been virtually useless as central processors in IBM-compatible personal computers. What Intel failed to see was that meeting the PC standard, which had become required for compatibility with existing applications and operating systems, also meant keeping the DMA and interrupt controllers the same. These functions were included on-board the 186 and 188 and were similar to the 8086-family peripherals, but were different enough that it was impossible to build a totally compatible PC using the 186 or 188.

With the 286, Intel has been much more successful. Unfortunately, the 286 has some major architectural flaws. One indication of this is that although 286-based PCs have been in production for three years, very little software has attempted to use the machine as anything other than a fast 8086 emulator.

The design of the IBM PC and MS-DOS also played a major role in the 286 compatibility problem. When Intel designed the 8086 architecture, they expected that segment registers would be a resource controlled exclusively by the operating system: applications programs would request segments, and would be given values by the operating system. Applications programs would not directly access the hardware via fixed addresses, but would do so via operating system calls.

In this type of OS-controlled environment, the 286 compatibility issues largely disappear. Only an OS change would be required to deal with the new aspects of the architecture. However, the PC environment didn't turn out that way — programmers have used segment registers for anything and everything, from index registers to temporary data registers. This just won't do when the segment register is a pointer into a descriptor table, as it is in protected mode. Direct access to fixed hardware addresses also became commonplace as programmers looked for more speed than was available from I/O access via operating system calls.

The user of a 286 is in a bind. Operating in real mode, the address space is limited to 1M byte. To get at the rest of the 16M-byte address space, the processor must be in protected mode. But unmodified 8086 programs cannot (in general) run in protected mode! Thus, the only way for an 8086 program to access extended memory is to normally operate in real mode, and call a memory access routine that operates in protected mode for accessing extended memory. (Note that *extended memory* is normally accessed memory above 1M, whereas *expanded memory* is the name given to the Lotus/Intel/Microsoft bank-switching scheme for accessing additional memory through a window in the lower 1M.)

Making the situation worse is the incredible lack of a method of switching back to real mode once in protected mode. To access memory beyond the 1M-byte space, real-mode software must perform the following tasks:

1. Set a special control bit in the system's battery-backed RAM.
2. Switch to protected mode.
3. Access full memory space as needed.
4. Initiate a full microprocessor reset by sending a command to the keyboard processor, which in turn asserts the processor's reset line.
5. The power-up reset routine checks the special control bit set in step 1, so it knows that this is not really a reset but just a switch back to real mode.
6. The program that made the memory access is continued.

This awkward procedure for switching modes greatly slows the 286's access to extended memory. Because the 386 provides a "virtual 8086" mode in which multiple 8086 programs each have their own 1M address space, the need to switch to real mode is eliminated. Each 8086 program can act as if it has its own PC and is operating in real mode, while in fact it is executed in protected mode as a concurrent task. This makes possible effective use of 640K (the usable portion of the 1M space) times the number of applications running at any time. For a single application to directly access a larger address space, however, still requires modifying the program so that it can run in the native protected mode.

Microprocessor Report reviews and analyzes industry news based on information obtained from sources generally available to the public and from industry contacts. While we consider these sources to be reliable, we cannot guarantee their accuracy.