The big switch: SRAM-replacement DRAMs

Brian Dipert - October 16, 2003

As the number of functions you’re squeezing into your system designs increases and as the required amount of code and data RAM proportionally—or, more likely, exponentially—also grows, the cost-per-bit advantage of DRAM over SRAM becomes more difficult to ignore (Figure 1). To decide whether a switch from SRAM to DRAM might make sense, you might first want to recall why you chose SRAM in the first place. Perhaps you just didn’t want to hassle with DRAMs’ multiplexed address bus, refresh requirements and access-collision avoidance, varying cycle times in random-versus-sequential modes, or other idiosyncrasies. Good news: Latest generation PSRAM (pseudo-SRAM) DRAMs are looking and acting more than ever like the SRAMs they’re striving to replace.
Maybe power-consumption concerns drove your initial selection of SRAM instead of DRAM. SRAMs, after all, don't incorporate a perpetually leaking per-cell charge-storage capacitor that requires periodic refreshing, bloating the memory's standby-current draw. Striving for the lowest possible standby current is also one of the key motivations that drove the conversion from four-transistor-two-resistor SRAM cells to today's dominant six-transistor-cell approach. At modern deep-submicron lithographies and low supply voltages, though, all those transistors' leakage currents combine to slowly but surely narrow the gap between SRAM and DRAM standby-current draw. Keep in mind that the power-consumption differences between DRAM and SRAM may look more significant on paper, when you do a component-versus-component data-sheet analysis, than they are in real life, when you need to squeeze a multi-SRAM array into your system to match the density of the alternative single-chip DRAM.

Power-optimized DRAMs enable you to turn off refresh to all or a portion of the device. If the memory finds use as an oft-updated video buffer, for example, you might not even need to refresh it at all. And in applications requiring high performance, the memory is in standby mode such a small percentage of the time that standby-power-consumption differences between SRAM and DRAM become largely irrelevant. DRAM's low price may be nice, but its higher reliability than SRAM in the face of alpha-particle and cosmic-ray bombardment may end up being an even more compelling reason to choose it (Reference 1). The higher capacitance of a DRAM storage cell than that of an SRAM latch means that DRAM is more immune to the bit-flipping injected charge of incoming ionizing radiation. Reliability-cognizant design of the DRAM cell can further improve its radiation tolerance, and system-level parity and EDAC (error detection and correction) boost the memory subsystem's reliability to an even loftier level (Reference 2).

And what about speed; doesn't SRAM still hold an edge here? Indeed it does; fundamentally, accessing an active-transistor-latch structure is faster than reading or writing a passive storage capacitor. A hefty percentage of the overall performance difference between the two technologies, though, derives from DRAM's cost-focused die-size-slimming decisions; DRAM could be almost as fast as SRAM if it incorporated large sense amps, numerous small sub-arrays, and other architecture alterations, but then the favorable price differential between it and SRAM would evaporate. The bottom line is that, with every lithography reduction, DRAM naturally gets faster, and, for an increasing number of applications, it's fast enough, which is all that really matters, right (Reference 3)?

**Slow sippin'**

DRAM manufacturers have for years been professing their commitment to non-PC markets, but, as long as PCs gobbled up all the chips the suppliers could make, DRAM vendors never had to move beyond talking the talk to walking the walk of developing and delivering products tailored for other applications. Before now, you were stuck with minor specification and packaging tweaks that vendors made to fundamentally PC-defined chips. The advantage of this approach was that you benefited from the high-volume cost efficiencies of the PC market. The disadvantage, though, was that you ended up with a chip that only marginally met your unique needs if at all.

Much as the high-speed SRAM vendors have split into two opposing camps—the QDR Consortium and the SigmaRAM Consortium—PSRAM suppliers have generally divided themselves into two alliances, with the exception of companies such as Hynix, which, at least for now, seems content to go it alone (Reference 4). The CellularRAM Co-Development Team originally comprised Infineon and Micron Technology, and Cypress Semiconductor joined them in September 2002. In May, the alliance announced that Infineon was sampling 16- and 32-Mbit parts, that Micron was making 32- and 64-Mbit variants available for sampling, and that Cypress was scheduled to ship its first samples
next year. CellularRAMs come in asynchronous-SRAM-compatible, 48-bump BGA packages; some densities are also available in 54-bump, superset-featured packages that support NOR-flash-compatible page mode and full synchronous burst mode for reads and writes.

The contending alliance, with The Jetsons-reminiscent acronym COSMORAM (Common Specifications for Mobile RAM), which emerged in February, comprises Fujitsu, NEC, and Toshiba. Integrated Silicon Solution also touts PSRAMs compatible with COSMORAM specifications. COSMORAM is the latest output of a multistage partnership between the three companies that began in late 1998 when they announced a joint approach to flash-plus-SRAM multidie chips (Reference 5). In the spring of 2002, they extended their partnership to include page-mode SRAMs and the stacked multidie devices that contained them, and COSMORAM is a further extension that incorporates support for synchronous PSRAMs but is, unfortunately, pinout- and otherwise-incompatible with CellularRAM.

Fujitsu adapted its 64-Mbit-page-mode FCRAM (fast-cycle-RAM)-based devices, which have a 32-bit data bus, and its 32-Mbit, 16-bit-bus FCRAM devices to fit the COSMORAM specifications, and, less than two months ago, the company announced a 128-Mbit PSRAM. Fujitsu calls its parts Mobile FCRAMs to signify the emphasis on low power consumption, and the company's product portfolio also includes conventional asynchronous 16- and 32-Mbit memories. Fujitsu originally developed FCRAMs, as their name implies, for applications requiring high performance. The highly partitioned FCRAM array delivers power advantages, too, specifically in the necessity of activating only the portion of the array necessary to service a pending access request.

Although CellularRAMs and COSMORAMs differ in their details, quite a bit of conceptual commonality exists between them. They internally handle array refresh operations without need for external control, and you also needn't worry about whether you're accessing a location that's undergoing a simultaneous in-progress refresh. Minimally, you can turn off refresh for the lowest possible power consumption; documentation often refers to this mode as deep power-down. Some chips also support the ability to refresh only a portion of the array. You might even be able to adjust the refresh rate to account for the current ambient operating temperature, because storage capacitors more rapidly "bleed" charge when they're hot. With some devices, your system processor needs to determine the temperature using circuitry external to the memory and subsequently write a value to a PSRAM register, whereas more advanced PSRAMs integrate the temperature-monitoring and refresh-rate-adjusting functions.

**Speed racers**

Perhaps not surprisingly, two opposing camps have also emerged to do battle for the future of ultra-high-speed DRAMs. Ironically, the teams comprise many of the same players as in the low-power PSRAM tug of war. Fujitsu was one of the pioneers in developing fast DRAMs. The company's high-speed FCRAMs came in variants with both multiplexed and nonmultiplexed and, therefore, SRAM-like address buses. Other, alternatives to Fujitsu's high-speed FCRAMs include Ramtron with its now-defunct EDRAMs and synchronous ESDRAMs; NEC with its similarly obsolete Virtual Channel DRAM; and MoSys, whose MDRAM has achieved much greater success as an embedded memory than as a discrete device, with the notable exception of its presence, both embedded and discrete, in Nintendo's GameCube (Reference 6).

At least for the moment, however, Fujitsu is content to allow its FCRAM licensees, Toshiba and, more recently, Samsung, carry the high-speed torch while it focuses on low-power applications for the technology. The highly segmented FCRAM array is beneficial not only in minimizing active current draw but also in enabling fast reads and writes by shortening the internal address- and data-
routing lines and by employing a multistage pipeline that allows multiple accesses to simultaneously be in different points of internal progress. Toshiba's first-generation Network FCRAMs acted as a superset of DDR-1 SDRAMs, operating at higher random-access speeds but maintaining their predecessors' I/O interfaces and organizations, along with a unified bidirectional data strobe (Table 1).

Second-generation Network FCRAM-II parts, which Toshiba's recently announced 288-Mbit device exemplifies, incorporate parity-inclusive 9-, 18-, and 36-bit I/O-organization options, befitting their networking-application focus, and they migrate to dual unidirectional strobes—an approach that, its backers claim, is easier to implement in a high-speed-system design. Commensurate with these architectural tweaks, the parts run at greater-than-333-MHz clock rates and have random-access times as low as 20 nsec. FCRAM advocates tout not only the memories' fast random accesses, but also their high bus efficiency, along with an estimated 15% lower power consumption than that of DDR-I SDRAM at the same frequency. Toshiba plans a 512-Mbit Network FCRAM-I device in early 2004 and a 576-Mbit FCRAM-II upgrade in early 2005.

RLDRAM (reduced-latency DRAM), which Infineon and Micron Technology offer, arrived later to market than FCRAM, yet its backers claim that their additional development time has been well-spent, resulting in a superior approach. FCRAM-II meets or exceeds many of the first-generation RLDRAM features, specifically those regarding bus width and read and write speeds. Unlike FCRAMs, RLDRAMs come in both multiplexed (for RLDRAM-II only) and nonmultiplexed (for both RLDRAM generations) variants; the multiplexed option enables designers to build packages with fewer pins, and the nonmultiplexed alternative improves address-bus and, therefore, access efficiency (Table 2).

RLDRAM-II also incorporates on-die termination with user-programmable impedance, an on-die DLL, and optional separate data-input and -output buses. The separate-buses feature is reminiscent of QDR SRAMs and Sigma-RAMs, which also include separate data-input and -output buses. All of these features increase die size, pin count, and, therefore, cost, both for the die itself and for the package containing the die, but RLDRAM's suppliers claim that the architecture's advantages more than justify the higher price. They also claim that competitor FCRAM's low yields counterbalance its smaller dies. RLDRAM-II's vendors optimistically, I think, based on their schedule-prediction track record, anticipate that it will achieve first-silicon status by year-end and that it will operate as fast as 400 MHz, along with delivering FCRAM-like 20-nsec random-access cycles.

**Conventional developments**

Note that, with both FCRAM and RLDRAM and unlike PSRAM, you're responsible for externally controlling DRAM refresh and handling any access collisions with in-progress refresh. Memory-controller designs from IP (intellectual-property) providers, such as Denali Software with its Databahn line, along with chip suppliers, such as Altera and Xilinx, ease the frustration of this stipulation. However, it opens the door to your consideration of more mainstream DRAM variants. DDR SDRAMs in short-trace point-to-point configurations, such as on graphics boards, are now available at speeds of 500 MHz, translating to 1-Gbps-per-pin peak-transfer rates, and manufacturers are developing even faster DDR-II versions. Random-access cycles are much slower than bursts, but if your design's memory-access patterns are highly predictable, these chips may deliver the optimum price/performance combination for your application.

Rambus proponents also hope that RDRAM will have a role in your future system designs, both with high-speed versions of Direct RDRAMs and with its next-generation XDR DRAMs, previously known by their Yellowstone code name. Only time will tell if partners Elpida, Samsung, and Toshiba deliver
first XDR silicon on schedule in the first half of next year, if the parts' performance and price match today's hype, and if risk-averse system engineers aware of the company's tumultuous PC past are willing to give it another shot at success. Currently planned XDR chips will initially appear in 512-Mbit densities with 16-bit data buses that deliver 6.4 or 8 Gbytes/sec peak data rates, along with corresponding 40- or 32-nsec row-cycle times.

My suggestion that you consider mainstream high-performance DRAM also applies to LPDRAMS (low-power DRAMs), which usually lack on-chip refresh controllers and other PSRAM circuits. However, these omissions mean that the chips are available in higher densities with lower per-bit costs than PSRAM alternatives. Infineon and Micron, which are apparently both fond of forging partnership programs with fancy marketing names, are synchronizing their respective developments of Mobile DRAMs, as Micron calls them. (Infineon calls them MobileRAMs.) To Micron's credit, the visionary company first discussed with me many of the concepts now finding their way into LPDRAMs many years ago at an International Solid State Circuits Conference (Reference 7).

Many of the other companies that this article mentions, along with other large DRAM suppliers such as Elpida, are also developing low-power-tuned SDRAM variants that are more or less compatible with early specification drafts coming from JEDEC-standardization efforts. At a minimum, when researching your options, you'll likely find chips that run at lower voltages and therefore draw less current, albeit perhaps running at lower speeds, than their PC-targeted peers. You might also uncover self-refresh capability, along with a deep-power-down mode that shuts off refresh to the entire chip. Some devices support finer grained subdivision of refresh enable versus disable; others might allow you to control the refresh rate based on ambient temperature; and a few might even integrate the temperature sensor, as do their PSRAM brethren. Happy hunting! I suggest that you extensively model various memory alternatives in your application before crowning a winner. Many vendors offer functional models in Verilog and VHDL formats, and Denali Software's SOMA models provide an industry-proven, unbiased alternative approach.

References

OTHER COMPANIES MENTIONED IN THIS ARTICLE:

<table>
<thead>
<tr>
<th>Company</th>
<th>Website</th>
<th>Company</th>
<th>Website</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reqlware</td>
<td><a href="http://www.reqwireless.com">www.reqwireless.com</a></td>
<td>Xilinx</td>
<td><a href="http://www.xilinx.com">www.xilinx.com</a></td>
</tr>
<tr>
<td>Joint Electron Device Engineering Council (JEDEC)</td>
<td><a href="http://www.jedec.org">www.jedec.org</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ramtron</td>
<td><a href="http://www.ramtron.com">www.ramtron.com</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nintendo</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>