The use of embedded-DRAM technology has become widespread, especially in higher end system designs, because of its superior performance, silicon-area savings, and low power compared with discrete-memory approaches. Traditionally, in cost-sensitive consumer applications, large memory arrays were usually best suited to discrete, commodity-memory implementations. But, as the supply of low-density DRAM wanes and prices rise, system designers are finding that embedding DRAM at densities of 16 Mbits and below is more cost-effective than discrete-memory alternatives.

A highly integrated embedded-DRAM approach also simplifies board design, thereby reducing overall system cost and time to market. Even more important, embedding DRAM enables higher bandwidth by allowing a wider on-chip bus and saves power by eliminating DRAM I/O buffers. Today, designers can take advantage of these capabilities as various 0.25-µm, embedded-DRAM technologies enter production, with 0.18-µm variants scheduled to appear by the end of this year.

Technology alternatives

The three most common ways to combine embedded DRAM and logic are the DRAM-based, blended (or hybrid), and logic-based technologies. A DRAM-based process is practically the same as commodity DRAM. It uses DRAM-periphery devices to build logic circuitry with perhaps the addition of one or two metal layers for logic routing. Blended technology uses additional front-end masks to enhance DRAM-periphery devices to speed logic performance. Logic-based embedded DRAM enables transistors that have performance compatible with leading-edge logic processes, resulting in an improved DRAM-to-logic interface and a logic-optimized path to implementing system-on-chip designs.

System designers are turning to embedded DRAM for several reasons. Unlike with commodity DRAMs, which are available only in a standard range of densities, such as 4, 16, and 64 Mbits, you can specify the exact amount of memory required—for example 5, 9, or 17 Mbits—for your system design in the embedded-DRAM macro block. Thus, no memory is wasted, and area and cost are optimized. In addition, you can specify the exact memory configuration and interface in the macrocell, thereby offering flexibility and optimum system performance.

Each of the three embedded-DRAM types combines the functions of both memory and logic on a single die. The elimination of the additional I/O bonding pads required for two separate chips saves about 5 to 10% of overall silicon area over discrete approaches. This space saving is particularly significant for ASICs having 300,000 logic gates or fewer, because it alleviates the pad-limitation
problem common in these designs.

**DRAM-based embedded DRAM**

DRAM-based embedded-DRAM chips begin with a DRAM-process architecture, usually one with two metal layers on top of which the foundry adds one extra metal layer for logic routing. The philosophy behind this type of embedded DRAM is usually the same as that employed by discrete, commodity-DRAM manufacturers. The intent is to make the cell as small as possible, because a smaller cell means a smaller and thus a less expensive die. Typically, the DRAM cell is 50 to 100% smaller than a cell of logic-based technology of the same generation. However, in this approach, the peripheral circuitry used for logic design is the same as that for commodity-based DRAM. The high thermal cycles introduced in the DRAM-based process, just before processing the first metal level, induce the diffusion of transistor dopants. This induced diffusion degrades device performance.

The use in commodity DRAM of polycide in the polysilicon gate makes it impossible to introduce an advanced PMOS device to a DRAM-based logic-plus-memory process. Polycide is necessary to make a self-aligned bit-line contact in the DRAM cell, thus eliminating otherwise-necessary design-rule space between the transfer gate and the bit-line contact and consequently reducing the cell size by at least 20%. Because of this self-aligned contact, commodity DRAM can use only buried-channel PMOS, a technology that disappeared from traditional logic processes after the 0.35-µm generation.

For these reasons, DRAM-based technology lags behind logic technology performancewise by at least two generations. For example, the performance of devices from 0.18-µm, DRAM-based transistors is roughly equivalent to the performance of cutting-edge, 0.35-µm, logic-process transistors.

**Blended embedded DRAM**

Blended, or hybrid, embedded DRAM is similar to the DRAM-based type but with several additional mask layers to enhance the DRAM-periphery devices, which also serve as logic transistors. In essence, a blended process incorporates some additional steps lacking in a commodity-DRAM process to enhance the performance of peripheral circuitry. Normally, this added processing involves slightly reducing the after-transistor thermal cycle, thereby reducing dopant diffusion; adding a source/drain silicide process outside the DRAM array; and more aggressively reducing the channel lengths of peripheral transistors. The blended embedded-DRAM process architecture still looks much more like a DRAM-based device than a logic-based device, however, because of features such as buried-channel PMOS transistors and possibly due to polycide instead of silicide gates.

The hybrid process, like the DRAM-based process, is not library-compatible with logic-targeted circuits. Therefore, you can't easily port logic-process-intended designs to DRAM-based and hybrid processes. This redesign scenario is common, because system designers usually first design a stand-alone logic chip, and only later decide to create a second, embedded-logic design for a more optimized, or higher end, product offering.

The hybrid device speed/power figure of merit is closer to 1.5 generations behind that of logic than the two generations behind of DRAM-based embedded DRAM. For example, the performance of 0.22-µm hybrid technology (contrasted with the more advanced and costly 0.18-µm, embedded-DRAM technology) is roughly equivalent to that of 0.35-µm logic. Note, however, that when comparing figures of merit, you must consider several variables, such as speed, power dissipation, design rules, and gate densities. Manufacturers worldwide have introduced a range of hybrid types, and their performances vary, depending on the process approach.
**Logic-based embedded DRAM**

Logic-based embedded DRAM derives from an existing logic process, so it has the same design rules and Spice models as the advanced stand-alone logic technology. Thus, you need not struggle with any sacrifice of speed, because the speed/power figure of merit is the same as that of the derivative-logic process. Logic-library compatibility also allows any design tested in a stand-alone logic technology to easily migrate without modification into a logic-based embedded-DRAM implementation. In addition, logic-based embedded DRAM uses the extensive variety of design libraries developed for stand-alone logic, thus making logic-based embedded-DRAM designs more convenient for designers.

Yet, the logic-library compatibility of embedded-DRAM technology comes at a price. Logic-based embedded-DRAM processes are usually the most complex, requiring five to eight more masking steps than traditional logic processes require. DRAM and logic processes substantially differ from each other, and vendors must incorporate a number of "tricks" to combine them both in a single flow. Process complexity is not the only issue. Logic-based embedded DRAM creates a memory cell roughly 70% larger than the one of a cutting-edge DRAM-based technology, because the DRAM-based process is always optimized to yield the smallest possible memory cell.

Deciding which approach is best for your design is usually a simple task. For example, in designs in which logic dominates the chip layout, logic-based process designs are more economical because logic-design rules are denser than those of commodity-DRAM periphery devices. But if the area balance shifts toward the DRAM array, DRAM-based or hybrid designs are more economical, even though they offer lower performance than logic-based embedded-DRAM designs.

You can produce a small DRAM cell in a 0.18-µm, logic-based embedded-DRAM process. The approach shown in Figure 1, for example, uses a self-aligned polysilicon bit-line contact and polycide word line. This technique results in a higher performance DRAM array, as well as a smaller cell. Yet, the DRAM structure uses metal as a bit line. This approach is good for reducing mask count and wafer cost. It allows for the removal of at least two critical masks, compared with a commodity-DRAM, front-end process. Moreover, the resistance of a metal bit line is lower than that of a conventional polycide bit line typically used in commodity DRAMs, thereby allowing higher speed and lower power dissipation. Finally, the logic circuitry is similar to conventional logic technology, using cobalt salicide, dual-gate poly (p+poly NMOS and n+poly PMOS), and abrupt pn junctions for high performance.

**Applications**

DRAM-based and blended-embedded-DRAM technologies often appear in applications that require high memory density in a small area. Typically, these are systems with as much as 128 Mbits of memory in a 0.18-µm process technology. DRAM-based and blended technologies are also best for applications that are more cost-sensitive (Figure 2). Such applications include CD-ROM, DVD-ROM, disk drives, printers, lower end graphics, 10/100-Mbit Ethernet switches, stand-alone SRAM replacements, and custom-designed DRAMs.

The major benefit of logic-based embedded DRAM is higher speed. Thus, logic-based technology often finds use in high-performance applications, such as high-end consumer and networking designs. Applications that depend on video-signal encoding, such as digital videocameras, laptop PC graphics, smart cellular phones, and PDAs, also benefit from logic-based embedded DRAM. Portable applications also benefit from embedded DRAM’s lower power dissipation.

In addition, some fast custom-memory designs are now possible using logic-based embedded-DRAM
technology. Because commodity-memory standards do not apply to embedded DRAM, embedded DRAM is more flexible to use. Specialized designs oriented toward speed, bandwidth, and low power, rather than the low price and efficiency that have historically been the aims of the DRAM macrocell, are therefore possible. Architectural innovations that this technology makes possible include higher bandwidth DRAM with a very wide bus for handling a lot of parallel data in high-end graphics applications and networking switches. Other embedded-DRAM designs emulate SRAMs through fast random access, rather than the typical DRAM page-mode access, by eliminating the traditional DRAM multiplexed address bus.

Design trade-offs

One of the less obvious advantages of embedded-DRAM design is that JEDEC DRAM specifications do not necessarily apply. The tight refresh-rate specs of stand-alone commodity DRAMs, for example, are not necessarily required. Instead, the memory controller can more frequently refresh the cells. Normally, the memory controller refreshes every bit in a commodity DRAM every 64 msec, but embedded DRAM allows a refresh spec as low as 2 msec, depending on the application and design. Higher refresh rates contribute to increased DRAM yield, because yield loss in mature DRAM processes typically ties to the designed-in refresh rate.

On the other hand, a high refresh-rate spec may be undesirable from the application point of view. If you use the embedded DRAM as a cachelike buffer, for example, a too-short refresh cycle increases the probability of cache misses. More important, you cannot too frequently refresh embedded DRAM in portable applications, because of standby power concerns. Indeed, the largest contribution to DRAM standby-power dissipation comes from the periodic refresh operation, which is essentially active-circuit power dissipation. Therefore, it is crucial to minimize the refresh rate in portable systems to maximize battery life.

You can employ several design approaches to address the DRAM power-dissipation problem. One approach is "smart refresh." With this method, when the logic circuit is in the active mode, such as when a camcorder is recording or a computer game is played, you frequently refresh the DRAM. When the logic circuit is off, the DRAM macro slowly goes into slow-refresh, or sleep, mode, thus reducing standby-power dissipation. For certain applications, DRAM need not be refreshed at all in standby mode, because data loss is acceptable.

Another power-saving approach is multibank design, or designing DRAM arrays in smaller banks, which allows shorter word and bit lines. A line half the length of an alternative has a capacitance half as large. Because power dissipation is proportional to capacitance, power dissipation per bit read also decreases by roughly twice with twice-shorter bit lines, not taking into account additional decoding. A multibank approach allows faster DRAM-access speeds for the same reason: bit- and word-line charging time is proportional to capacitance.

Testing issues

DRAM ICs require extensive testing for functionality and reliability. The test flow for embedded DRAM is even more complex because it involves testing and stressing both the DRAM macro and the logic circuit. Figure 3 depicts two test flows for embedded DRAM: the flow used today and the "ideal" test flow.

Today's common test flow starts with the conventional CP (chip-probe) 1 DRAM test. After mapping failed bits, the tester replaces them with redundant, or repair, bits, which are always present for yield enhancement. Repair-bit activation uses laser-programmable metal fuses. The CP 2 DRAM test verifies laser repair. If the DRAM repair is successful and the memory array is working at the wafer-
probe level, the tester then examines the logic portion at the wafer-probe level. After inking and disposing of bad die, the good die get assembled and tested for packaging yield (FT 1).

From the testing point of view, replacing the standard CP 2 memory testing with BIST (built-in self-test) can lead to more cost-effective results. BIST translates the logic tester’s 1 and 0 signals into the timed signals needed for DRAM testing. It allows a logic tester to examine DRAM, thereby enabling you to combine logic and DRAM testing on one piece of equipment. Yet BIST circuitry can occupy 5 to 10% of the overall area of the DRAM macro, depending on DRAM density and the amount of test patterns to be covered. Although BIST is available in many DRAM intellectual-property offerings, most system designers using embedded DRAM today still prefer separate DRAM and logic CP 2 tests.

Burn-in tests stress the DRAM part of the circuit at high temperature to screen out "soft" defects. These defects do not fail under normal test conditions, but slowly become worse and cause cell failures after years of failure-free operation. Therefore, circuits with these defects must be screened out and marked as nonyielding to avoid product failures in the field. For several reasons, package-level burn-in is more expensive for embedded DRAM than for commodity DRAM. First, the socket board is custom-made and can be expensive. More important, because embedded-DRAM ICs are usually large (because they contain both DRAM and logic), each socket board fits only a small number of die, increasing the costs of throughput and, therefore, burn-in. Finally, package-level burn-in testing requires pins that provide special access to the DRAM, thus reducing the pinout savings advantage of embedded DRAM.

You can address the drawbacks of package-level burn-in via wafer-level burn-in, in which the DRAM block is stressed on the wafer level at internal stress pads. This approach has a faster throughput. Wafer-level burn-in time is only a few minutes, compared with the hours required for package-level burn-in testing. You can also repair weak bits that have failed wafer-level burn-in, increasing overall product yield. However, some studies that correlate package and wafer burn-in results indicate that you can't screen out all package-level failure modes at the wafer level; only 80% or so of failures are detectible at the wafer level with the right burn-in conditions. Therefore, it appears that wafer-level burn-in cannot completely replace package-level burn-in. Instead, a combination of the two might be a better approach. Wafer burn-in screens out most failures at the beginning of the test flow so that they may be repaired if possible, whereas either full or shorter package burn-in perform the ultimate reliability check of a packaged device.

In addition, other issues make embedded-DRAM testing different from commodity-DRAM testing. Stand-alone DRAM is usually specified to work at temperatures as high as 85°C, and all testing is done at this temperature. Yet logic is always more computationally intensive than DRAM, and it heats up the chip to more than 100°C in a real operating conditions. Therefore, most semiconductor manufacturers benchmark their logic technology at temperatures as high as 110 or even 125°C, and they must accordingly benchmark the embedded-DRAM block. This restriction does not apply to every system, however, and you should determine appropriate refresh times and testing conditions for each design. For example, ICs used for mobile applications do not heat up as much as other designs, so you can increase DRAM refresh time and perform testing at lower temperatures.

Another testing issue is sensitivity to switching noise. Because of their logic content, embedded-DRAM ICs are inherently noisier than stand-alone DRAM ICs. You can address noise issues at the design level by higher sense amplitude margins and by increasing the charge-coupling ratio in a DRAM array. Some designs also use a shielding grounded metal plate above the DRAM block to protect it from metal logic routing. The noise problem is even more acute if logic metal routing is done over the DRAM block. You cannot completely verify DRAM-circuit robustness with respect to noise in every design by using macro-verification test chips. Yet you can partially address it by using
substrate-noise generators near the on-chip DRAM well and via metal lines over the DRAM array that swing up and down during testing.

Author info

Chung S Wang, PhD, is director of the Memory Technology R&D Division at TSMC (Taiwan Semiconductor Manufacturing Co), and Edward CK Chen is director of the Special Technology Product Marketing Division at TSMC.