Estimating IC Manufacturing Costs

Die Size, Process Type are Key Factors in Microprocessor Cost

By Linley Gwennap

This is the second in an occasional series of articles discussing integrated-circuit manufacturing. The first (see ***070705.PDF****) covers basic manufacturing issues. Future articles will compare different vendors' capabilities and also look at packaging issues.*

Building an integrated circuit is unlike any other manufacturing activity. It requires incredibly expensive equipment with highly trained technicians performing minutely detailed operations. The most unusual aspect is, despite this fanatic devotion to precision, more than half of the end product is typically discarded as unusable—and vendors are happy with this result! Imagine an auto manufacturer throwing away half of its cars because of defects…well, maybe that's not so hard to imagine.

With an understanding of the economics of IC manufacturing, it is possible to estimate the actual manufacturing cost of any given chip based on a few simple parameters. The cost of a motherboard, for example, is highly dependent on its design (type of components used, their performance, etc.) but this is not true for a chip. An IC is like a roll of film; it costs the same to develop no matter what the pictures are. In this case, a chip goes through much the same basic manufacturing process whether it is an Alpha CPU or a 386, with a few extra steps depending on the complexity of the chip.

Manufacturing Process Overview

Most integrated circuits are built from silicon *wafers*. These are thin disks (about one millimeter thick) that are usually 150 mm (6") in diameter, although most new fabrication plants (called *fabs*) are being built for 200-mm (8") wafers.

Each wafer contains many chips of the same type. Since most chips are square or rectangular, they are usually laid out in a grid pattern, as shown in Figure 1, and arranged to fit as many as possible on the wafer. An individual chip is called a *die*. The figure also shows that a single wafer can hold more chips if they are smaller. The number of die per wafer doubles if the die area is cut in half. (In fact, it is more than double due to wasted space around the edge of the circular wafer.)

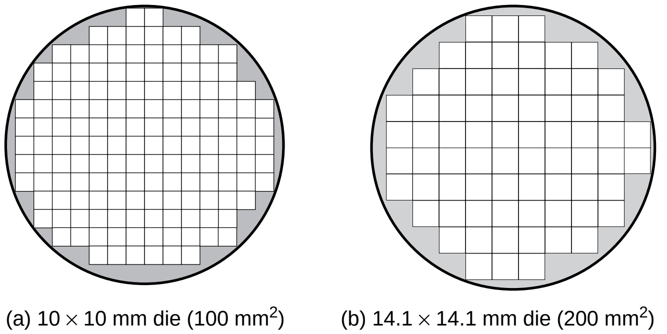
(a) 10 × 10 mm die (100 mm^2)
(b) 14.1 × 14.1 mm die (200 mm^2)

Figure 1. A 150-mm wafer can hold 150 of the smaller chips (left) but only 72 of the larger chips (right).

For any given process from a particular vendor—for example, Intel's 0.8-micron CMOS process—each wafer goes through the same steps, regardless of whether LAN chips or microprocessors are being built. Even in similar processes from different vendors, the sequence of activities is quite similar. Thus, the cost of processing a single wafer can be estimated by comparing it to a process of similar complexity.

Because the circuits being etched onto the wafer are hundreds of times smaller than the width of a hair, the smallest particle of dust can destroy a circuit if it comes in contact with the wafer during manufacturing. These *defects* can also be caused by tiny impurities in the silicon or a minor glitch in processing the wafer. In most cases, defects can be thought of as scattered randomly across the surface of the wafer. A single defect can render an entire die useless.

To determine which chips are defective, each die is tested while it is still part of the wafer. The wafer is then cut, or "scored," so that the individual chips can be removed and the defective ones discarded. The good dice are placed in packages and usually retested. Parts that pass these final tests are ready for sale.

Wafer Cost

The equipment needed to process a wafer is extensive, including the "clean room" itself, precision lenses and "steppers" that project the image of the design onto the wafers, plasma ovens, acid baths, and high-speed testers. Once all of this equipment is in place, a new fab begins a long series of trial runs to determine the best way to configure the system so it produces wafers with the smallest transistors and the fewest defects.

To get from an empty plot of land to a functioning fab can take two or three years, much of which is spent putting together and tuning the new IC process. Even when the new process is ready for volume production, the percentage of defects may still be relatively high, increasing the cost of the chips using that process. It may take another year or two of gradual improvement before the defect rate declines to its final value. At that point, the process is called *mature*.

A typical 0.8-micron CMOS fab today builds 150-

mm wafers at about 5000 per week. It cost around $200 million to $300 million to build and develop and has now reached maturity. The price tag for future facilities is going up; Intel recently began work on a 0.35-micron fab in Arizona and expects to spend $800 million before the first production chips roll off the line in 1996.

This development cost must be spread across all of the chips built at that facility. According to VLSI Research (San Jose, CA), equipment depreciation accounts for about a third of the total cost of processing a wafer in a 0.8-micron fab. Process development and other overhead costs take another third, and the remaining costs are split between the actual labor and material needed to process a wafer.

Since chip vendors do not disclose their actual cost of processing a wafer, we must estimate it based on the complexity of the process. We assume that companies using similar processes have similar costs. A few vendors may have inefficient fabs, but most are world-class manufacturers; others (like Sun and MIPS) use external foundries and have shopped around for the best deal.

Based on industry sources, a 150-mm, 0.8-micron, three-layer-metal CMOS wafer costs about $1300 to process, including depreciation and overhead costs. Adding extra process steps for a fourth metal layer or BiCMOS circuits adds about 15% to the wafer cost. Each new process generation (for example, moving to 0.6 micron) increases costs by about 60%, mainly due to higher equipment and process-development costs. These figures are typical for most microprocessors; high-volume chips such as 386 and 486 processors could see a 5%–10% reduction due to efficiencies of scale.

These costs do not include any so-called "indirect" costs. A real company will incur design costs, marketing costs, sales costs, corporate overhead, and taxes. Since these costs are highly dependent upon the type of chip, the structure of the company, and market factors, we cannot include them in this analysis.

Another difficult issue is the effect of capacity utilization. Since the depreciation and overhead costs are both large and fixed, the true wafer cost goes up considerably if the fab is underutilized. At 75% capacity, these costs would be 33% higher on a per-wafer basis, increasing wafer cost by about 20%. As it is difficult to determine the utilization of vendors' factories, we assume that they are at or near capacity.

Yield Issues

If all wafers are created equal, then why does the cost of different chips vary so much? The most important issue is die size. As we saw in Figure 1, a standard-size wafer holds more small chips than big chips; since the cost of the wafer is the same either way, more chips per wafer means less dollars per chip. Only chips without defects can be sold, so the *yield*, or percentage of non-defective chips, is also important.

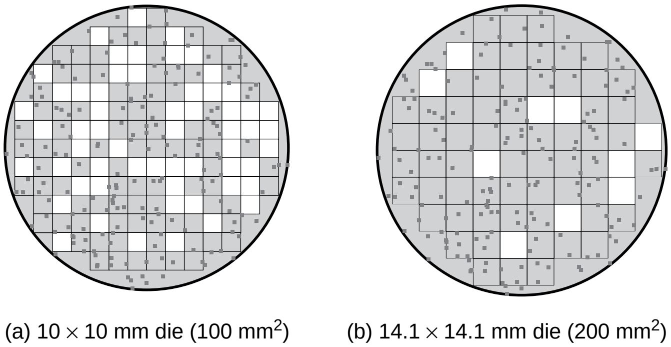
(a) 10 × 10 mm die (100 mm^2) (b) 14.1 × 14.1 mm die (200 mm^2)

Figure 2. The wafers from Figure 1 have been processed with 1.0 defects per cm^2. The yield of the smaller chips is 39% (58 of 150) while the yield of the larger chips is just 14% (10 of 72).

Yield is strongly related to die size. Figure 2 shows two wafers with the same random defect pattern. The smaller chips have more than twice the percentage yield as the larger ones, which are more likely to contain a defect. In this example, a single wafer produces nearly six times as many good 100-mm^2 chips than 200-mm^2 chips. This indicates that the rule of thumb—die cost increases as the square of the die area—is too low.

The number of defects per wafer also affects yield. This rate is usually measured by the average number of defects per cm^2 and is called the *defect density*. Most of the work in developing and improving a particular IC process focuses on reducing the defect density, as it directly affects the cost of making good chips. The type of chip being built, however, does not affect defect density.

Like wafer cost, defect density varies from vendor to vendor, but we will assume that leading manufacturers all have about the same rate. When a new process first goes into production, defects can be as high as 2.0 per cm^2, but this rate declines over time, reaching about 1.0 per cm^2 (and sometimes even lower) at maturity. Processes with extra steps (more metal layers or BiCMOS) may have slightly higher defect densities.

To slightly complicate matters, not all defects render a chip non-functional. Most chips have a ring of pads around the outside edge, as shown in Figure 3; a small defect in the pad area is generally not fatal. Also, some microprocessors contain redundant circuits (typically in memory areas) that can compensate for a single defect.

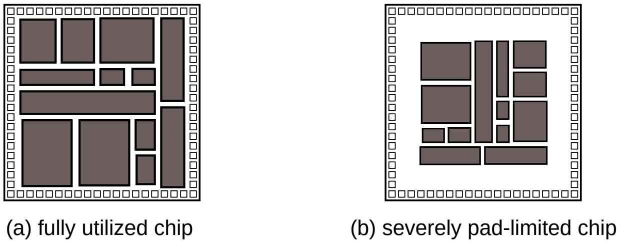
(a) fully utilized chip (b) severely pad-limited chip

Figure 3. Differences in effective area. (a) 85% (b) 45%.

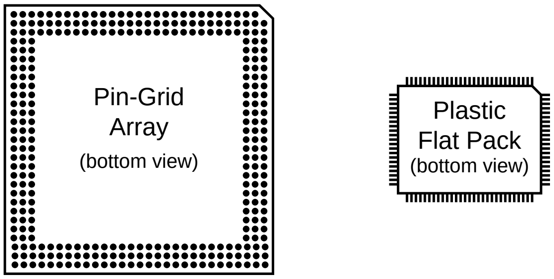

Figure 4. Standard package types. (a) PGA, (b) PQFP.

Finally, some chips have empty areas, because their large number of pads forces their die to be bigger than the area needed just for the circuitry. These chips, like the one in Figure 3(b), are called *pad-limited*.

All of these factors go into calculating the *effective area* of a die. Most microprocessors have an effective area of about 85%, which takes into account the area used by the pad ring. The PowerPC 601 has an effective area of nearly 100%, since it has no pad ring *(see **070602.PDF**)*. SuperSPARC has an effective area of 73% due to redundant rows and columns in its caches. (These figures are calculated from measurements of die photos.)

Testing and Packaging

Once a wafer has been fully processed, the individual dice must be tested and packaged. To avoid the cost of packaging defective parts, some testing is usually done when the chips are still in the wafer. During *wafer test*, small probes make contact with the pads of a chip, initiating test vectors and checking the results. Chips that fail are marked so they can be discarded before packaging. The major variable at this point is test time; a highly-integrated processor can take a few minutes to test, while a simple memory chip might take seconds. Bad chips generally fail within the first few seconds, terminating the test.

After wafer test, the good chips are removed from the wafer for packaging. This involves placing the die in the package, attaching tiny bond wires to connect the pads to the pins in the package, and then sealing the completed part. A final test determines which parts are fully functional. This test catches any chips that may have been incorrectly packaged and may also exercise additional functions beyond the initial wafer test. Vendors often make a trade-off between reducing the duration of the wafer test and discarding more packaged parts during the final test.

The most significant cost at this point is the package itself, which depends on the type of package and the number of pins. Microprocessors generally use one of the two types of packages shown in Figure 4. Pin-grid arrays (PGAs) can handle chips with as many as 600 pins and have excellent thermal characteristics, but they typically cost $20 to $30; special high-performance packages can cost even more. Plastic quad flat packs (PQFPs) are smaller and may cost only a few dollars, but they are limited as to the number of pins and the amount of heat that they can handle.

The assembly cost is relatively small, particularly since most packaging today is done mechanically. Final test costs are also fairly small, since the testing can be done with relatively simple equipment compared to an expensive wafer tester.

Building the Cost Model

The µPR Cost Model combines all of these factors to estimate the manufacturing cost of current microprocessors. Because a number of factors in this model are rough estimates, the resulting costs could have a significant deviation from actual manufacturing costs. This model is not accurate enough to detect the difference between a vendor that has a 65% gross margin and one that has a 75% gross margin, for example. It is most useful for comparing the manufacturing costs of different chips, because errors in the wafer cost and defect density tend to cancel out if we assume that leading vendors have similar costs. These issues should be kept in mind when using the results of this model.

The model begins with the wafer cost, based on the type of process being used. The number of die per wafer is calculated from the die area, assuming a 150-mm wafer.

The yield calculation is complicated. First, the defect density is estimated based on the complexity and maturity of the process. We then use a statistical model to calculate the effect of randomly-placed defects on the wafer:

$$Yield = \left(1 + \frac{Defect\ Density \times Die\ Area}{3}\right)^{-3}$$

(Dingwall's Equation). The die area is adjusted by the effective-area percentage to account for those portions of the die that cannot be affected by defects.

The cost of processing the wafer must then be spread across the number of good chips:

$$Die\ Cost = \frac{Wafer\ Cost}{Die\ Per\ Wafer \times Yield}$$

This results in the cost per good die but does not include testing and packaging costs.

Since all dice in a wafer must be tested, the total wafer test time must be spread across all of the good chips. We apply a rate of $5 per minute for wafer test. The total test time includes a full test for each of the good chips and a few seconds each for bad chips. We assume that microprocessor test time (for good chips) varies between 30 seconds for a simple device, such as a 386, to as long as five minutes for a multimillion-transistor chip like SuperSPARC or Pentium.

Finally, the model adds the cost of packaging and final test. Since there are several independent package vendors, the material package cost is readily available. The cost of packaging the chip and performing a final test is small; we assume one cent per pin. The model includes an adjustment for final test yield; for PQFP chips, we assume a final yield of 93%, but for PGA chips (which are usually subjected to a more rigorous wafer test) we use a figure of 97%. A significant portion of the final test cost is the cost of discarding the chips that fail; this cost is greater for chips with expensive dice or costly packages.

Playing the Frequency Game

Adding the factor of clock frequency complicates the cost picture. Even if a set of chips passes all functional testing, some may operate at a higher frequency due to minor process variations, as shown in Figure 5. Vendors typically assign their chips to speed grades using a "binning" process and charge more for the faster parts, although the cost model shows that all speed grades cost the same amount to manufacture.

A problem arises if customer demand does not match the yield curve. Using the example in the figure, only 5% of the chips can be sold as 33-MHz parts. If the demand for the fastest parts is 10% of all orders, the vendor must make twice as many wafers to meet this demand, leaving a huge oversupply of the slower parts. In this situation, the vendor would have to charge enough of a premium to cover the cost of the unused parts.

At the other end of the scale, some chips may be fully functional but may not operate at a high enough frequency to be marketable. In the example shown, the vendor may need to discard the 5% of the parts that fall below 20 MHz, slightly increasing the cost of the remaining chips. There are too many variables—the yield curve, the demand curve, the price structure—to easily model this situation; we assume that vendors are able to sell all (or nearly all) functional chips.

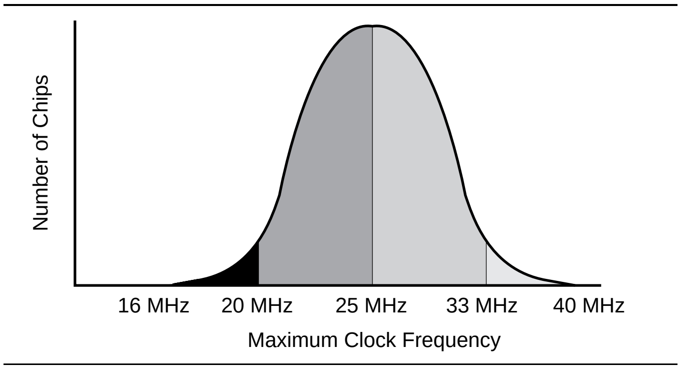

Figure 5. A hypothetical frequency distribution for microprocessors produced using the same IC process.

Applying the Cost Model

Table 1 shows how the model works, using some current microprocessors as examples. These examples show the direct relationship between die area and manufacturing cost. As die area increases, the number of chips per wafer steadily declines, as does the percentage yield. This combination sharply reduces the number of good chips per wafer, expanding the portion of the fixed wafer cost that each good chip must carry.

Where two chips have a similar die area, factors such as package cost and wafer cost come into play. Defect density can also be important and is difficult to estimate, particularly for processes that are high on the learning curve. Pentium, for example, uses a new BiCMOS process derived from Intel's existing 0.8-micron CMOS process. Due to the new process, the defect rate is undoubtably quite high at the moment but should fall rapidly as Intel gains experience with BiCMOS.

The 386DX shows how inexpensive a microprocessor can get. The model shows a 71% yield (which may underestimate the actual yield) and a huge number of die per wafer. The $1000 wafer cost, like the other wafer costs, includes depreciation, but it is likely that Intel's 0.9-micron fab is fully depreciated; removing these costs would reduce the manufacturing cost estimate even further. The biggest cost in this example is the testing and packaging cost, which is difficult to reduce below a certain level.

The 486DX and DX2 (both are now on the 0.8-micron process) are fairly inexpensive. The 486DX2 in the table includes a costly PGA package; the lower-frequency DX parts are able to get by with a PQFP, cutting

Chip Name	IC Process (# of metal)	Wafer Cost	Defects per cm^2	Die Area (mm^2)	Die per Wafer	Probe Yield	Package Type	Package Cost	Test and Assembly	Estimated Mfg Cost
Intel 386DX	0.9µ CMOS (2)	$900	1.0	43	360	71%	132-pin QFP	$1	$4	$9
Intel 486DX2	0.8µ CMOS (3)	$1200	1.0	81	181	54%	168-pin PGA	$11	$12	$35
IBM PowerPC 601	0.7µ CMOS (4)	$1700	1.3	121	115	28%	304-pin QFP	$3	$21	$76
MIPS R4000SC	0.8µ CMOS (2)	$1100	1.0	184	71	28%	447-pin PGA	$31	$18	$104
HP PA7100	0.8µ CMOS (3)	$1300	1.0	196	66	27%	504-pin PGA	$35	$16	$125
DEC 21064 (150)	0.7µ CMOS (3)	$1500	1.2	234	53	19%	431-pin PGA	$30	$23	$202
TI SuperSPARC+	0.7µ BiCMOS (3)	$1700	1.6	256	48	13%	293-pin PGA	$20	$34	$336
Intel Pentium	0.8µ BiCMOS (3)	$1500	1.5	294	40	9%	273-pin PGA	$19	$37	$483

Table 1. Applying the µPR cost model to a few popular microprocessors shows a wide variation in factory cost, primarily due to differences in die size and process type. (Source: data in white columns from vendors; data in gray estimated from µPR cost model)

Processor	Die Area (mm^2)	Estimated Mfg Cost	Price (Volume)	Estimated Multiplier
i386SX	43	$9	$31 (10K*)	3.4×
i486SX	72	$17	$74 (10K*)	4.3×
i486DX	81	$23	$245 (10K*)	10.7×
Cyrix 486DLC	108	$26	$85 (10K*)	3.3×
R4200	81	$37	$70 (high)	1.9×
PowerPC 601	121	$76	$280 (20K)	3.7×
R4000PC	184	$82	$420 (10K)	5.1×
MicroSparc	225	$121	$179 (10K)	1.5×
R4400PC	184	$163	$450 (10K)	2.8×
SuperSparc-LE	256	$200	$250 (10K)	1.3×
21064 (200)	234	$284	$1231 (10K)	4.3×
Pentium	296	$483	$965 (1K)	2.0×

Table 2. A comparison of estimated cost and price for various microprocessors. (Source: data in white columns from vendors, *indicates 15% off 1K price; data in gray from µPR cost model)

the cost by about $10.

None of the RISC chips in Table 1 approaches the 386 or 486 in cost, but the PowerPC 601 comes the closest, despite the high wafer cost of IBM's 0.7-micron four-layer-metal process. This advanced process, however, enables IBM to keep the die area quite small, resulting in a low manufacturing cost. The 601 also uses an inexpensive PQFP package.

The R4000 takes a different path to low cost, relying on a mature 0.8-micron, two-layer-metal process. It could nearly match the cost of the PowerPC except for its costly PGA package, which is needed to support the larger number of signals in the SC version of the part.

The DEC chip in the table is the 150-MHz version of the 21064 Alpha processor, which uses a slightly lower-cost process than its 200-MHz big brother. TI's SuperSPARC+, on the other hand, is the higher-speed version of that processor. Both have high wafer cost, a large die, and an expensive PGA package. Digital is said to have extremely high wafer costs due to its low volumes, but this model assumes DEC's wafer costs are competitive.

Pentium makes SuperSPARC's die look small, and also has a high wafer cost and expensive package. The new Intel chip is nearly unmanufacturable in a 0.8-micron process and is certainly extremely expensive. The model shows a yield of about four good chips per wafer; some sources indicate that the current number is even lower. Pentium will be much more practical when built in a 0.6-micron process on 200-mm wafers, producing 20–30 good die per wafer due to higher yields.

Relationship Between Cost and Price

There are many other factors than base manufacturing cost that determine the price of a chip. One cost that must be considered is that of designing the chip. Microprocessor design is very expensive. Vendors usually don't discuss these costs, but estimates range from about $30 million up to the $100 million that Intel is rumored to have spent developing the 486. To be profitable, a vendor must spread this design cost across the total lifetime unit volume of that chip.

For example, if Intel really spent $100 million on the 486, it may expect to eventually sell 50 million chips based on the 486 design. To pay back its investment, Intel would have to add $2 to the base manufacturing cost. Many of these 486s will be derivative chips, so the cost of making modifications would have to be amortized across the sale of these derivative chips, in addition to the few dollars that cover the initial design effort.

For other chip makers, it isn't so easy to pay for the development costs. Suppose Sun and TI spent a total of $50 million to develop SuperSPARC. Suppose that they expect to sell 500,000 of these chips. It isn't clear who pays whom in this particular arrangement, but someone will have to pay roughly $100 extra for every chip sold just to cover the cost of designing SuperSPARC.

Microprocessor vendors have other costs that must be accounted for. Marketing and sales teams attract new customers, and the support group takes care of current customers. Management oversees it all. Typically, these costs are assessed as a percentage of gross revenue from each chip, anywhere from 20% to 70%, depending mainly on volume. Accounting systems vary, but these costs cannot be ignored.

One can try to estimate the actual price of a chip by adding up all of these costs and including some percentage for profit, but this method rarely works because most chips are priced at the market rate and not according to the manufacturing cost. The cost model can give a baseline for how low a vendor could price its chips, but the actual price as determined by the market is typically much higher.

Table 2 compares the estimated manufacturing cost of several chips with their lowest published list price. The 386, for example, carries a fairly low markup due to the high volumes and intensive competition in this market. The 486DX, which until recently had little competition, has a high multiplier. Most of the RISC chips carry moderate multipliers, although TI appears to be cutting its SPARC prices to the bone. Either that company's cost structure is lower than we think, or it is trying to aggressively win business with its low-end chips.

Using readily available information, it is possible to come up with a reasonable estimate of the manufacturing cost of a microprocessor. This information can be used by designers to estimate the impact of changing the die size or package type. It can also be used to compare designs based on cost/performance rather than price/performance. Finally, it can indicate how far a vendor could cut prices in a competitive market. The results show that, whenever price is an issue, vendors must keep the die size small, either through clever design or aggressive manufacturing techniques—or both. ♦