Xilinx Improves Its Semiconductor Supply Chain Using Product and Process Postponement

Alexander O. Brown — *Owen Graduate School of Management, Vanderbilt University, 401 21st Avenue South, Nashville, Tennessee 37203*

Hau L. Lee — *Graduate School of Business and the Department of Management Science and Engineering, Stanford University, Stanford, California 94305*

Raja Petrakian — *Xilinx, Inc., 2100 Logic Drive, San Jose, California 95124*

The semiconductor firm Xilinx uses two different postponement strategies: product postponement and process postponement. In product postponement, the products are designed so that the product's specific functionality is not set until after the customer receives it. Xilinx designed its products to be programmable, allowing customers to fully configure the function of the integrated circuit using software. In process postponement, a generic part is created in the initial stages of the manufacturing process. In the later stages, this generic part is customized to create the finished product. Xilinx manufactures a small number of generic parts and holds them in inventory. The use of these generic parts allows Xilinx to hold less inventory in those finished products that it builds to stock. And for some finished products, Xilinx can perform the customization steps quickly enough to allow it to build to order.

High technology industries, such as semiconductors and computers, are characterized by short product life cycles and proliferating product variety. Faced with such challenges, companies in these industries have found that delaying the point of product differentiation can be an effective technique to cut supply-chain costs and improve customer service. This postponement technique is a powerful way to enable cost-effective mass customization [Feitzinger and Lee 1997]. To use

0092-2102/00/3004/0065/$05.00
1526–551X electronic ISSN
This paper was refereed.

INVENTORY—PRODUCTION—MULTI-ITEM, ECHELON, STAGE
INDUSTRIES—COMPUTER—ELECTRONICS

INTERFACES **30:** 4 July–August 2000 (pp. 65–80)

postponement effectively, companies must carefully design their products and processes. Through careful design of the product and the process, many electronics and computer companies have been able to delay the point of product differentiation, either by standardizing some components or processes or by moving the customization steps to downstream sites, such as distribution centers or retail channels. Lee [1993, 1996]; Lee, Billington, and Carter [1993]; Lee, Feitzinger, and Billington [1997]; and Lee and Sasser [1995] give examples.

Postponement concepts have also been applied in other industries, such as the automobile industry [Whitney 1995] where product modularity enables delayed customization of auto parts. Indeed, Ulrich [1995] showed that a high degree of product modularity coupled with component-process flexibility could render postponement possible and effective. Lee, Padmanabhan, and Whang [1997] also said that both product and process modularity support postponement. Modular designs for products or modular processes (a manufacturing process that can be broken down into subprocesses that can be performed concurrently or in different sequential order) are techniques that enable postponement.

The semiconductor industry has been plagued by a proliferation of product variety because of the overlapping product life cycles—companies introduce new or enhanced versions of products before existing products reach the ends of their life cycles. In the programmable-logic segment of the industry, new customers will use the enhanced versions in their products, but some existing customers may delay adopting the new versions despite their improved performance and price. Periods of appreciable demand for a version of a product may range from six months to two years, with products sometimes having an extended period of very low end-of-life demand. Thus, semiconductor companies must offer many products simultaneously. The product-variety problem is compounded by unpredictable demands and long manufacturing lead times.

Semiconductor firms face unpredictable demand, in large part, because of their upstream position in the supply chain. An integrated circuit (IC) made by a semiconductor firm is a component of other subassemblies or final products. Thus, it must pass through other companies, such as contract manufacturers, distributors, and resellers, before the final product reaches the end consumer. Lee, Padmanabhan, and Whang [1997] describe the "bullwhip effect" in which demand fluctuations increase as you travel upstream in the supply chain. Since semiconductor firms are located far upstream in the supply chain, they often face such large fluctuations.

Manufacturing cycle times in the semiconductor industry are still very long despite advances in the process technology. The manufacturing process, consisting of wafer fabrication, packaging, and testing, takes about three months. With such long manufacturing lead times, the semiconductor companies must hold large inventories of finished goods or their customers—computer assemblers, telecommunication manufacturers, or other electronics manufacturers—must hold large

inventories to hedge against demand uncertainties.

Product variety, long production lead times, and demand unpredictability negatively affect the manufacturing efficiency and performance of both semiconductor companies and their customers. These characteristics also affect the customer's product-development processes. For example, one part of a telecommunications-equipment manufacturer's product-development process might be the custom design of application-specific integrated circuits (ASICs). The design process often includes creating a number of prototypes before settling on a final working design. Because of the long production times, there is often a significant delay between designing and receiving prototypes. Since time to market is a key factor in the success of high-tech products, this delay may be very costly for the manufacturer. To compress the cycle, such manufacturers may request many prototypes towards the beginning of the design process, resulting in additional design and development costs.

Semiconductor companies offer many products simultaneously.

Product variety, long lead times, and demand unpredictability are all unavoidable and problematic characteristics of the semiconductor industry. However, some companies are finding new ways to cope with them. Xilinx, Inc., uses innovative design principles of postponement to avoid excessive inventory while providing great service to its customers. It uses both product and process postponement extensively.

In product postponement, the firm designs the product so that it can delay its customization, often by using standardized components. Xilinx relies on a more extreme form of product postponement. Instead of the firm performing the final configuration during manufacture or even distribution, it designs the ICs so that its customers perform the final configuration using software. Consequently, Xilinx greatly shortens the product-development cycles of its customers, as the customers do not have to specify the full features and functionalities of the ICs before production.

Using proprietary design technologies, Xilinx creates many types of ICs, differentiated by such general features as speed, number of logic gates, package type, pin count, and grade. Although the customers perform the final configuration of the logic, they must order products with the appropriate general features. For example, a customer with a large and complex design requiring high speed must select a physical device type with a large number of logic gates and a high speed. Later the customer can configure the logic of the device using software, creating an enormous number of possible designs. Product postponement is very suitable for programmable devices because a near-infinite number of designs can be created from a few thousand physical-product permutations.

In process postponement, the firm designs the manufacturing and distribution processes so that it can delay product differentiation, often by moving the push-pull boundary or decoupling point toward

the final customer. A push-pull boundary is the point in the manufacturing-and-distribution process at which production control changes from push to pull. Early in the process, prior to the push-pull boundary, the firm builds to forecast. Later in the process, after the push-pull boundary, it builds to order. Often, process designs allow manufacturers to change their push-pull boundaries. A highly celebrated example of process postponement is the case of Benetton, which used to make sweaters by first dyeing the yarns and then knitting them into finished garments of different colors. Its push-pull boundary used to be at finished sweaters—all production was built to forecast. Benetton resequenced its production process so that it first knits undyed garments, and then dyes them (and thereby customizes them to the different color versions) on demand. Hence, its new push-pull boundary is between knitting and dyeing [Dapiran 1992].

To improve its manufacturing process, Xilinx focused on creating a new push-pull boundary, working with its suppliers. Rather than going through all the steps to create an IC in its finished form from a raw silicon wafer, Xilinx divides the process in two stages. In the first step, its wafer-fabrication supply partners manufacture unfinished products, called dies, and hold inventory of this material. This inventory point is the push-pull boundary. Based on actual orders from the customers, another set of supply partners pull dies from inventory and customize them into finished ICs.

The Xilinx Supply Chain

Digital semiconductor devices can be broadly grouped into three categories: memory, microprocessors, and logic. While the general-purpose microprocessors can execute almost any logical or mathematical operation, logic devices provide specific functionality at lower cost and greater speed. However, the traditional method of defining the functions of a logic device is to configure it during the fabrication process. Recently, with the introduction of programmable logic devices, it has become possible to customize a generic but more expensive logic device using software after the logic device has been completely manufactured and packaged.

Xilinx was one of the first to use a virtual business model.

Founded in 1984, Xilinx developed the field-programmable gate array (FPGA), a programmable logic device, and it has become one of the two largest suppliers of programmable logic solutions in the world. The company's revenues in 1997 were $611 million and the gross margin was around 62 percent. Xilinx was one of the first semiconductor companies to use a virtual business model: it subcontracts out logistics, sales, distribution, and most manufacturing to long-term partners. Xilinx's only manufacturing facilities are its California and Ireland facilities that just perform some final testing. It meets about 74 percent of its total demand through distributors, whose expertise has evolved beyond traditional warehousing and inventory management to include engineering functions, such as helping customers design Xilinx parts into their systems. Xilinx

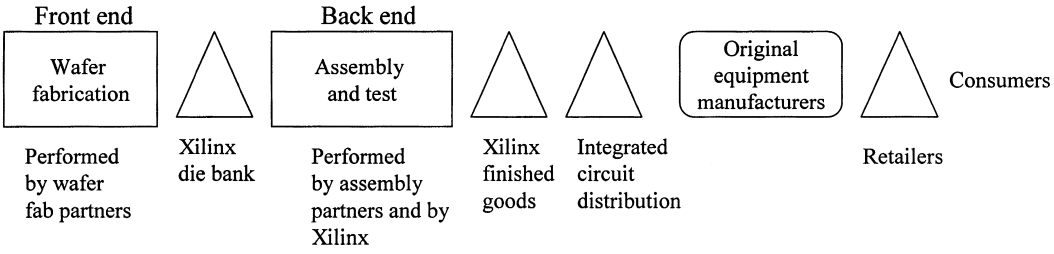

Figure 1: In the Xilinx supply chain, supply partners perform the wafer fabrication and assembly, while Xilinx manages production levels and the inventory levels in die bank and finished goods. After production, a distributor buys the integrated circuits and supplies them to original equipment manufacturers that incorporate the integrated circuit into their products. Consumers purchase the products through retailers. Triangles represent inventory stocking locations, and squares represent manufacturing processes.

keeps certain core functions in-house, such as technology research, circuit design, marketing, manufacturing engineering, customer service, demand management, and supply-chain management. This virtual business model provides Xilinx with a high degree of flexibility at low cost. Its partners benefit because Xilinx uses standard manufacturing and business processes and aggressively drives process improvements through technical innovation and re-engineering. Although the virtual model has strategic risks (the core competencies becoming commodity-like) and operational risks (unexpected lack of available capacity at suppliers), it has proven highly successful in the industry [Lineback 1997].

Today, most of Xilinx's competitors have access to the same fabrication process technology through their own wafer-fabrication partners. The technology and manufacturing gap between members of the industry is closing. Consequently, Xilinx sees management of the demand-and-supply chain as providing it with a competitive advantage in the market. In 1996, Xilinx executive management initiated a major initiative to overhaul the company's practices and processes for managing supply and demand.

In the Xilinx supply chain, the flow of materials begins with the fabrication process (front end), where raw silicon wafers are started and manufactured using hundreds of complex steps that typically take two months (Figure 1). Anywhere from 20 to 500 integrated circuits come from each fabricated wafer. In the last process steps of the front end, the wafers are sorted and tested for basic electrical characteristics. Although precise information is not available until the final test after assembly, this step provides some useful indications of the proportion of good integrated circuits on the wafer and the speed mix that they are likely to yield. After sorting and preliminary testing, wafers are stored in inventory—the die-bank. Planning wafer starts to ensure proper die-bank inventory is a major challenge, requiring such information as demand forecasts, projected yields, and work in process to determine the volume and mix of wafers needed to meet the demand and inventory targets.

The next link in the supply chain is the back end, a term that refers to both the assembly and test processes. In the back-end, wafers are first cut into dies, or individual "raw" integrated circuits. There are approximately 100 different types of dies. To be usable, the integrated circuits must be placed in a package, a plastic casing with electric lead pins, that allows them to be later mounted in a circuit board. There are usually about 10 to 20 package types from which a customer can select for a given die. The dies are wire bonded to form a permanent electrical contact with the package. The packaged dies are then tested electrically to determine if they meet stringent design and quality requirements and to determine their speed. There are usually about five to 10 different possible speed grades. The packaged devices that pass the quality tests are then stored in finished-goods inventory. With lead times of three weeks for assembly and test, planning of back-end starts is difficult, requiring information on both the backlog of orders and demand forecasts. One complexity involves the issue of device speed. Although Xilinx understands the expected fraction of dies that will yield to each speed level, the actual fraction for any given die is different. Thus, planning using the expected fraction of dies at each speed level will often result in a mismatch of supply and demand. To meet the demand, Xilinx will start more material in the back end and pick wafers intelligently using measurements collected in the fabrication-and-sort step.

Most Xilinx customers are serviced through distributors who maintain inventories of Xilinx finished-goods parts. The advantages distributors provide to Xilinx are that they have cost-effective means for handling large numbers of small to medium-size customer orders and they offer such value-added services as inventory consolidation, inventory management, and procurement-program support. The cost of Xilinx is that they add an extra link to the supply chain, causing a potential distortion in demand information. The lack of end-demand visibility can be partially offset when distributors provide Xilinx with systematic data regarding point of sale (POS), bookings, backlog, and inventory. Most Xilinx customers are original equipment manufacturers (OEMs) that put one or more Xilinx parts on a circuit board and then assemble a large system using the board and other components. The OEMs then sell these systems to other customers using various marketing and distribution channels. The Xilinx supply chain is further complicated by the practice of many OEMs of subcontracting the board assemblies to specialized vendors.

On-time delivery is emphasized at Xilinx. As a result, Xilinx has often resolved the trade-off between inventory and on-time delivery by adding inventory. One of the key goals of the supply-chain-management initiative is to achieve the same levels of customer service with lower inventory costs throughout the supply chain.

Product Postponement: The Programmable Logic Devices

Before recent developments in programmable logic, logic devices were primarily ASICs in which the logic was built in during wafer fabrication. Typically, the OEM customer would design an ASIC as part of

a larger design of the system board on which the ASIC would be mounted. The OEM customer submitted a design for the ASIC to a semiconductor manufacturer, who fabricated a prototype of the device according to design specifications. The characteristics of ASICs were fully determined during fabrication, and hence the OEM customer receiving an ASIC could use it only for the intended design. Yet, because of changes in the system specifications or design flaws, design iterations were very common in such product-development projects in the high-technology industries (Figure 2). Any change in the design of an ASIC required both modifying the semiconductor-fabrication process and manufacturing additional prototype ASICs using the modified process. A change in the fabrication process could cost hundreds of thousands of dollars and manufacturing prototype ASICs could take over three months. As a result, design iterations in systems using ASICs were very time consuming [Trimberger 1994].

With programmable logic devices, the OEM customer receives a "generic" device. These devices are not completely generic—each type has features that cannot be customized. Thus, once a customer chooses a generic die type, the customer can customize within a certain range of parameters. The features that create these hard design limits include die packaging, speed grade, maximum number of logic gates, voltage, power, maximum die input and output, and software programming methodology:

—The customer chooses from a set of possible package types and lead-pin counts. Different packages have different thermal and protective properties and have different maximum electrical input and output characteristics.

—The customer chooses from a set of speed grades, each of which produces a different clock rate. Higher speeds may be required for some applications.

—The customer chooses from a set of possible device sizes, specified by the number of logic gates. The number of logic gates determines the size and complexity of the logic design that can be implemented.

—The customer selects from a variety of voltages used to power the device (usually 2.5 V, 3.3 V, or 5V).

—Each generic device type has different power constraints.

—Each generic device has different maxi-

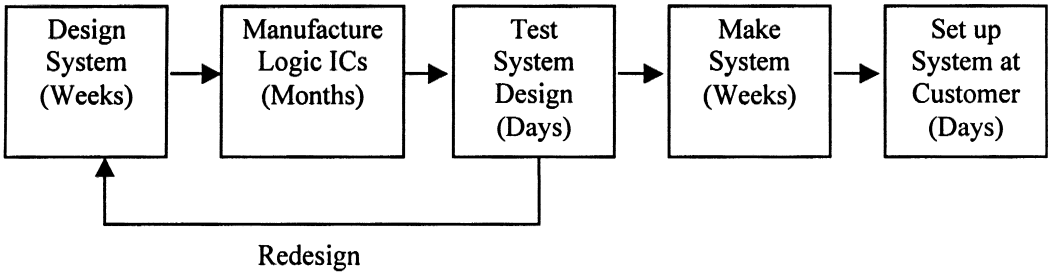

Figure 2: When building a system using an ASIC, the manufacturer incorporates the logic when the integrated circuit is manufactured. Thus, the designer must wait for a new integrated circuit to be manufactured to make design changes.

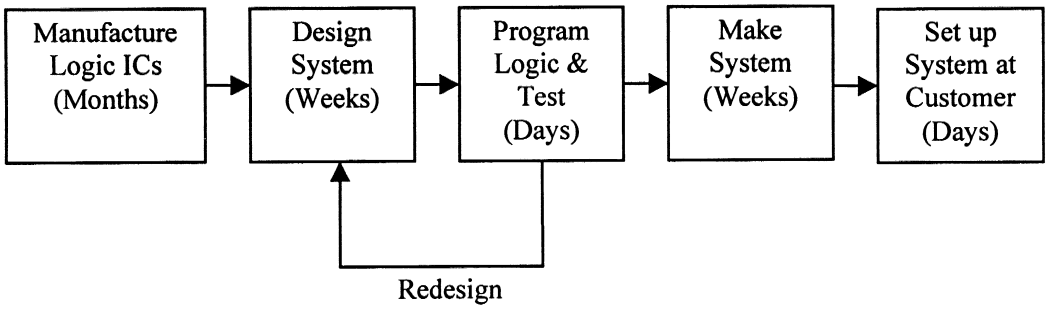

Figure 3: When building a system with a programmable logic device, the customer incorporates the logic using software after the integrated circuit is manufactured. Thus, design changes can be made quickly using software. In contrast to Figure 2, the steps "manufacture logic IC" and "design system" are reversed.

mum input and output electrical characteristics, for example, the maximum level of current that the device can put out.
—The customer may select a device that uses a familiar programming methodology.

Although the customer must decide on some characteristics in advance, the essential characteristic of the final device, the logic function of the device, is not defined in physical processing. Instead, the OEM customer programs, in minutes or hours, the programmable logic device using software running on a personal computer. The user downloads the information into the generic die and thus completes a fully customized logic device. With such a programmable logic device, the process for designing an end system is now dramatically different (Figure 3). Each design iteration takes less time as does the overall design and development process.

Besides shortening the design-process time, product postponement can improve the operational efficiency of the supply chain by reducing the procurement lead times. ASIC suppliers often operate under a build-to-order system, not maintaining finished-goods inventory (but they may have some in-process inventory). As a result, the procurement lead times for OEM customers are sometimes two to three months long. Since accurate forecasting of demand at the specific ASIC device level over such a long horizon is difficult, OEM customers using ASICs often keep large inventories of the ASICs. Programmable logic suppliers can afford to keep inventory in finished-goods form or in the die bank because programmable logic devices are more generic with more predictable demand. Thus, lead times for procuring programmable logic devices are in days or weeks so OEM customers who use them need less inventory.

In-system programming (ISP) allows even greater product postponement. With this capability, customers can easily program or reprogram the logic even after the device is installed in the system (Figure 4). For example, electronic systems such as multi-use set-top boxes, wireless-telephone cellular base stations, communications satellites, and network-management systems, can now be fixed, modified, or upgraded after they have been installed.

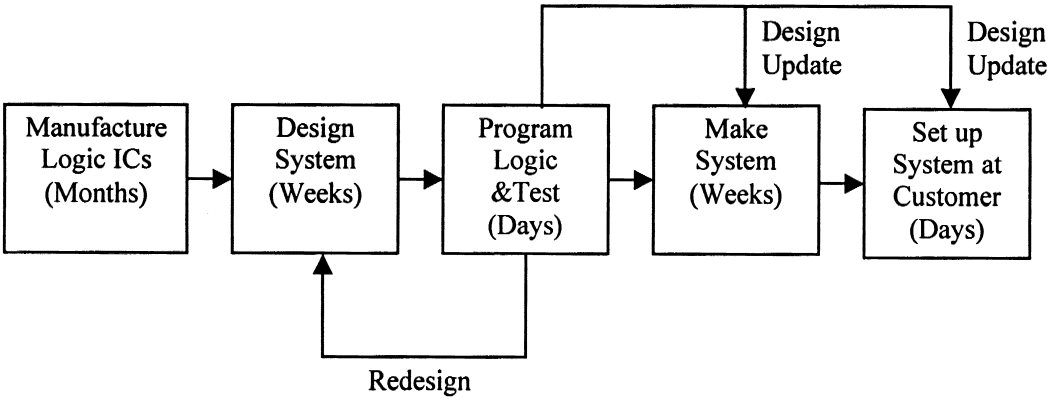

Figure 4: In a system built with a programmable logic device with in-system programming capability, the logic can be incorporated after the system is set up with the customer.

Process Postponement: The Die Bank as the Push-Pull Boundary

The use of product postponement allows Xilinx's customers to create a near-infinite number of different products (different logic designs) from a few thousand types of physical products (Xilinx finished goods). However, since demand for each finished good is usually very uncertain and manufacturing takes around three months, achieving excellent service with reasonable overall inventory levels was a challenge with this many different finished goods. Since many of the finished goods use the same type of die, Xilinx recognized an opportunity to implement process postponement to simultaneously reduce inventory and increase service responsiveness.

Its revised process using postponement works as follows. Instead of using the projected demands for individual finished goods to determine production at the front end, Xilinx aggregates the demands for finished goods into die demands and uses the projected die demands to determine the front-end production starts. After completing the front-end stage, it decides how to customize the dies into different finished goods in the back-end stage. It thus postpones product differentiation, moving it from the beginning to the end of the front-end stage. It still bases customization in the back-end stage on demand forecast (push), with inventory being held in finished-goods form. Thus, the push-pull boundary remains at the end of the process. Since the point of product differentiation moves forward but the push-pull boundary is still at the end of the process, we refer to this approach as partial postponement. Eppen and Schrage [1981] initially proposed this approach in a multilevel distribution setting; it is equally applicable to this manufacturing setting.

Although partial postponement provides benefits, moving the push-pull boundary to an earlier point in the process can increase them. In full die-bank push-pull postponement, the generic dies are held in inventory (the die bank) immediately after the front-end stage, and this die bank becomes the new push-pull boundary. No inventory is held in finished-

goods form; instead, the dies are customized according to customer orders.

We compared die-bank push-pull postponement and the no-postponement approach by analyzing the inventory and service trade-off for each approach using data from a family of finished goods derived from the one die type. We assumed independent and normally distributed demands and a weekly periodic review base-stock policy. For the no-postponement approach, we modeled the system as independent inventory nodes, each representing a finished goods part. We calculated the minimum inventory required to meet a service constraint (maximum expected back orders) for each node and summed the inventory across nodes. For a given level of safety stock, we estimated the expected back orders for each node using the demand uncertainty and the planning lead time [Nahmias 1993]. For the die-bank push-pull postponement approach, we modeled the system as a single inventory node at the die bank. We estimated expected back orders at this node using the demand uncertainty of the aggregated die demand. We showed that the die-bank push-pull strategy offers significant improvements (Figure 5).

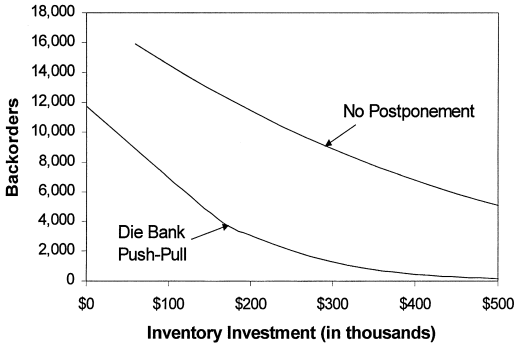

Figure 5: The graph shows the expected number of back orders as a function of the total inventory for two approaches: the no-postponement approach and the die-bank push-pull approach. For the same level of inventory investment, the expected number of back orders is much lower under the die-bank push-pull approach.

Although this die-bank push-pull postponement strategy offers performance improvements, it is not acceptable for customers that require fast deliveries. Thus, if the back-end lead time is two weeks and the customer needs delivery in one, Xilinx could not meet the customer's requirement. Xilinx wanted to move from a partial postponement approach to the die-bank push-pull approach and still satisfy such customer requirements. Thus, it has adopted a hybrid approach. Xilinx has been reducing back-end lead times, and the times for the majority of products are now shorter than customers usually require. It builds these products for the die bank according to customer orders (the die-bank push-pull strategy). It builds finished goods with longer back-end lead times and shorter delivery time to forecast (the partial-postponement strategy).

To determine the distribution of inventory between finished goods and die bank, we used the same number of finished goods as in the previous analysis. We assumed each finished-goods part had one of two back-end lead times: a time equal to the customer-response time and one longer than the customer-response time (set at the average for the parts with lead times greater than the customer-response time). We increased the percent of parts with the short lead time from 0 to 100 percent to generate the results. To avoid concerns about the order in which we selected

finished goods for back-end lead time reduction, we assumed equal demands for all finished goods. So that we could use Eppen and Schrage's [1981] model to analyze the partial postponement approach, we assumed all parts had the same coefficient of variation.

For parts with the short back-end lead time, we used the die-bank push-pull approach and determined the minimum die-bank inventory to maintain the desired level of service. For the parts with the longer back-end lead time, we used the partial-postponement approach. For these parts, we determined the inventory levels required for the given service level using Eppen and Schrage's results [1981]. Their results are for just such a partial-postponement structure (under a different name), and they allow us to calculate the effective demand uncertainty as a function of the individual finished-goods uncertainty levels and the front-end and back-end lead times. Using these results, we calculate the total safety stock in finished goods for a maximum level of expected backorders.

When few parts have short lead times, we must manage most parts using the partial-postponement approach, keeping most inventory in finished goods. As the number of products with short back-end lead times increases, we can build more parts from the die bank to meet customer orders, decreasing inventory in finished goods and increasing that at the die bank. The decrease in finished-goods inventory is much more rapid than the increase in die-bank inventory. Thus, moving towards the pure die-bank push-pull approach reduces inventory and dramatically reduces

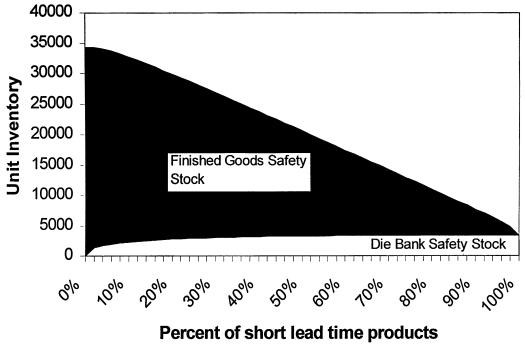

Figure 6: The figure illustrates the inventory distribution between die bank (white) and finished goods (black) when adopting a hybrid strategy. The horizontal axis is the proportion of finished goods that have back-end lead times within the customer-response-time window. As this proportion grows, more of the products can be built to order. Thus, the total inventory decreases significantly and the mix of inventory becomes more heavily weighted to die bank. Results are shown for a constant service level (as measured by expected back orders).

cost since the cost of finished goods is 40 percent more that the die cost.

Table 1 summarizes the four process-postponement approaches. The primary driver of the benefits of process postponement is the risk pooling or statistical pooling that occurs when aggregating demands for many finished goods into demand for fewer dies. The aggregate demand is less uncertain, and thus the firm can hold less inventory to provide the same level of service. The risk-pooling effect is large when the number of finished goods for each die type is large and the correlation between finished-goods demands is small. A large correlation between two finished goods means that if demand is larger than expected for one finished good, it will likely also be larger than expected for the second finished good. Fortunately, at Xilinx, there are a

Strategy	Postponement of product decision	Inventory at die bank	Inventory at finished goods
No postponement			■
Partial postponement	■		■
Die bank push-pull	■	■	
Hybrid	■	■	■

Table 1: For each of four approaches to managing Xilinx's process, the table indicates whether or not postponement is used and where inventory is held—in the intermediate and generic form at die bank or in the final form at finished goods. Xilinx adopted the hybrid model, allowing it to reduce inventories and maintain a high level of customer service.

large number of finished goods for each die (50 to 150) and the average correlation between the finished goods was found to be only 0.018.

Using postponement and holding most inventory at die bank has a number of additional benefits. Inventory held at the die bank is less costly than that at finished goods. About 30 to 50 percent of each product's total value is added in the back-end stage. Inventory held at the die bank also has a lower risk of obsolescence. Many finished goods have just a few customers. If demand drops unexpectedly, Xilinx may be left with inventory of these goods that it cannot sell to anyone else. Die inventory, however, has not yet been customized, and its flexibility greatly reduces the risk of obsolescence. Obsolescence costs in the industry are often about five percent of gross inventory per year, nearly all for finished goods. Postponement makes inventory management easier. In practice, inventory cannot be managed solely by a model-based system. Its decisions must be adjusted for issues beyond the model's scope. With process postponement, management can focus on managing the inventory of the 100 dies rather than trying to make decisions for 10,000 finished goods.

Implementing Process Postponement

Implementing process postponement often requires redesigning current products while trying to keep the changes transparent to the customer. Fortunately this can be done fairly easily in high-technology manufacturing because of the short life of products. To redesign a product to enable process postponement, a manufacturer can simply wait the short time until the next product-generation release when many customers will convert their designs to take advantage of speed and price benefits.

Xilinx designs products to allow for the use of process postponement, keeping the degree of customization low through the front-end stage. For a few general product categories, the die options (for example, many options for logic cell count) are numerous but packaging options are few. Thus, process postponement provides minimal advantage, and little can be done from a design perspective because some features (such as logic cell count) can be created only during the front-end stage.

Xilinx has pursued three process-related initiatives to make process postponement more effective—inventory modeling, supply-mix prediction, and back-end cycle-time reduction. It uses inventory

modeling to determine the appropriate push-pull boundaries for finished goods and to determine inventory levels at various stocking locations. For parts in finished-goods stock, it is optimal to keep inventory in the die bank for quick replenishment instead of using pure partial postponement. Xilinx uses inventory models to improve the hybrid strategy and to determine the optimal level of inventory to hold in the die bank to replenish finished goods and to fill orders for build-to-order parts. It currently uses a multi-echelon model developed jointly with IBM [Ettl et al. forthcoming; Brown et al. 1999].

In the supply-mix-prediction initiative, Xilinx uses statistical models to predict the speed mix of the die-bank inventory. Customer orders specify the desired speed. To customize dies from the die bank to meet customer orders or to replenish finished-goods stock, Xilinx must know how many dies are in each speed yield in the die-bank inventory. Xilinx can easily predict the average fraction of die per wafer that will be of each speed. However, due to slight perturbation in the wafer-fabrication process, the actual fraction for each individual wafer will be different. The objective of the supply-mix initiative is to predict this fraction. Although the true speed of a device is not known until it completes the assembly and test stages, Xilinx can get initial data using a test on die-bank inventory called wafer sort. Using this data, Xilinx applies regression and other statistical methods to estimate speed yield distributions quite accurately [Ehteshami and Petrakian 1998]. This knowledge enables a planner to choose wafers from the die-bank inventory that closely match the order requirements, thus reducing the wasted dies and improving response times.

> Xilinx reduced its inventory levels without harming overall customer service.

The third initiative to improve process postponement is a continuing process to reduce back-end lead times. Xilinx has worked with its manufacturing partners to reduce the wafer-fabrication time from three to one-and-a-half months. For Xilinx to make the die bank the push-pull boundary, the back-end lead time must be short. With a shorter back-end lead time, Xilinx can satisfy a larger proportion of customer orders using the die bank as the push-pull boundary instead of finished goods. Much of the back-end lead time is administrative time. Thus, Xilinx has been able to streamline the process and reduce the lead time through information technology and closer supplier (for assembly and testing) involvement. Internal planning and order fulfillment systems have been made more responsive and electronic data interchange or Extranet web-based tools have been used to expedite the exchange and processing of information between Xilinx and its worldwide vendors.

Conclusion

Xilinx has created tremendous values through product and process postponement. In the case of product postponement, it has found the value of ISP and IRL to be tremendous. For example, Hewlett-Packard Company used a Xilinx field-programmable gate array, a powerful variety of programmable logic devices,

when it designed the LaserJet Companion, reducing its design cycle by an estimated six to 12 months [Rao 1997]. For the electronics industry, Reinertsen [1983] estimated that a six-month delay in the development time of a product reduces the profits generated over the product's life cycle by a third.

Firms are only beginning to realize the potential of product postponement. Rao [1997] describes how IBM designed asynchronous transfer mode (ATM) networking switches when the industry had not yet fully developed standards and protocols. Using programmable logic devices with ISP capabilities, it was able to deliver systems to its customers that could easily be upgraded to the latest standards with no hardware changes. With more recent technological advances, firms can even provide these upgrades through the Internet for systems that are online. Villasenor and Mangione-Smith [1997] describe how FPGAs are changing the field of computing, possibly resulting in major technological breakthroughs. They envision computing devices that adapt their hardware almost continuously in response to changing input. They also predict that configurable computing is likely to play a growing role in the development of high-performance computing systems, resulting in faster and more versatile machines than are possible with either microprocessors or ASICs. With such technology, firms can postpone the definition of a product without limit, an ultimate form of product postponement.

Process postponement has also significantly improved financial performance at Xilinx. Although Xilinx has not kept performance metrics since it first introduced process postponement, its refinement of the process-postponement hybrid from the third quarter of 1996 to the third quarter of 1997 helped it to reduce corporate inventory from 113 dollar days to 87 dollar days (dollar days is the net inventory divided by the cost of goods sold for the quarter times 90 days per quarter). This translates directly into cost savings and improvements in the company's return on assets. At the same time, customer service, measured by the percentage of times that customer orders are filled on time, has remained the same. This is particularly impressive because during that period, Xilinx released an unusually large number of new products. Despite the proliferation of product variety and the increase in service back orders associated with technical problems with the new products, Xilinx reduced its inventory levels without harming overall customer service. During this time period, the inventory levels at the key competitors increased to well over 140 dollar days.

Gaining acceptance of the models took time and effort.

Currently, Xilinx is working closely with its partners to further reduce lead times at both the front-end and back-end stages. Clearly, reducing front-end lead times will result in even less safety stock needed in the die bank; while reducing the back-end lead times will enable Xilinx to satisfy more customer orders by using the die bank as the push-pull boundary.

Implementing postponement at Xilinx requires tremendous organizational sup-

port. The change from stocking primarily in finished goods to stocking primarily in die bank initially created some nervousness among the sales and logistics personnel who dealt with customers' orders. Although the company realized that it needed to use scientific inventory models to manage inventory levels effectively, gaining acceptance of the actual models took time and effort. We ran extensive computer simulations to demonstrate the effectiveness of the model and conducted intensive training and education programs with various functions within the company to create confidence in the model and acceptance of this new approach. The results showed that all these efforts were worthwhile, and postponement is now a key part of Xilinx's overall supply-chain strategy.

Acknowledgments

We thank Chris Wire, a key figure in driving the demand-and-supply-chain initiative at Xilinx, for his general input. We also thank John McCarthy and Dean Strausl for their support and vision in the projects and Donald St. Pierre for providing the engineering details of in-system programming for logic devices.

References

Brown, A. O.; Ettl, M.; Lin, G. Y.; and Petrakian, R. 1999, "Implementing a multi-echelon inventory system at a semiconductor company: Modeling and results," IBM Watson Labs technical report, Yorktown, New York.

Dapiran, P. 1992, "Benetton—Global logistics in action," *Asian Pacific International Journal of Business Logistics*, Vol. 5, No. 3, pp. 7–11.

Ehteshami, B. and Petrakian, R. 1998, "Speed yield prediction," Working paper, Xilinx, Inc., San Jose, California.

Eppen, G. D. and Schrage, L. 1981, "Centralized ordering policies in a multi-warehouse system with lead times and random demand," in *Multi-Level Production/Inventory Systems: Theory and Practice*, ed. L. B. Schwarz, North-Holland, Amsterdam and New York.

Ettl, M.; Feigin, G. E.; Lin, G. Y.; and Yao, D. D. forthcoming, "A supply network model with base-stock control and service requirements," *Operations Research*.

Feitzinger, E. and Lee, H. L. 1997, "Mass customization at Hewlett-Packard: The power of postponement," *Harvard Business Review*, Vol. 75, No. 1, pp. 116–121.

Lee, H. L. 1993, "Design for supply chain management: Concepts and examples," in *Perspectives in Operations Management: Essays in Honor of Elwood S. Buffa*, ed. R. Sarin, Kluwer Academic Publishers, Boston, Massachusetts, pp. 45–65.

Lee, H. L. 1996, "Effective inventory and service management through product and process redesign," *Operations Research*, Vol. 44, No. 1, pp. 151–159.

Lee, H. L.; Billington, C.; and Carter, B. 1993, "Hewlett-Packard gains control of inventory and service through design for localization," *Interfaces*, Vol. 23, No. 4, pp. 1–11.

Lee, H. L.; Feitzinger, E.; and Billington, C. 1997, "Getting ahead of your competition through design for mass customization," *Target*, Vol. 13, No. 2, pp. 8–17.

Lee, H. L.; Padmanabhan, V.; and Whang, S. 1997, "The bullwhip effect in supply chains," *Sloan Management Review*, Vol. 38, No. 3, pp. 93–102.

Lee, H. L. and Sasser, M. 1995, "Product universality and design for supply chain management," *Production Planning and Control: Special Issue on Supply Chain Management*, Vol. 6, No. 3, pp. 270–277.

Lineback, R. J. 1997, "The foundry/fabless model could become dominant," *Semiconductor Business News*, Vol. 1, No. 5, p. 1.

Nahmias, S. 1993, *Production and Operations Analysis*, second edition, Richard D. Irwin, Inc., Homewood, Illinois.

Rao, S. S. 1997, "Chips that change their spots," *Forbes*, Vol. 160, No. 1, pp. 294–296.

Reinertsen, D. G. 1983, "Whodunit? The search for new-product killers," *Electronic Business*, Vol. 9, No. 7, pp. 34–39.

Trimberger, S. M. 1994, *Field-Programmable Gate Array Technology*, Kluwer Academic Publishers, Boston, Massachusetts.
Ulrich, K. 1995, "The role of product architecture in the manufacturing firm," *Research Policy*, Vol. 24, No. 3, pp. 419–440.
Villasenor, J. and Mangione-Smith, W. H. 1997, "Configurable computing," *Scientific American*, Vol. 276, No. 6, pp. 66–71.
Whitney, D. E. 1995, "Nippondenso Co. Ltd.: A case study of strategic product design," Working paper, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Randy Ong, Vice-President, Operations, Xilinx Inc., 2180 Logic Drive, San Jose, California 95124–3400, writes: "This is to certify that the supply-chain efforts at Xilinx as described by the authors . . . have indeed been carried out. We have observed tremendous payoffs via such efforts, improving the efficiencies and effectiveness of our supply-chain and order-fulfillment processes. As a fabless semiconductor company, Xilinx has to rely on tight integration with our supply partners, distributors, and customers to remain competitive. Demand and supply-chain management is a cornerstone of our manufacuring strategy, and we are pleased to see such efforts creating great values for the company. I am also pleased to report that we are continuing our efforts to build supply-chain excellency so that Xilinx can become the leading edge supply-chain company in the semiconductor industry."