IMD INTERNATIONAL

IMD-3-0899

v. 10.02.2004

# AUTOMATIX, INC.: MODULAR DESIGN AND SUPPLY CHAIN STRATEGY

*Professor Ron Sanchez prepared this case as a basis for class discussion rather than to illustrate either effective or ineffective handling of a business situation*

From late 1992 to mid 1994, Bob Collier, CEO of Automatix, Inc., Ed Easterbrook, vice president of product development, Janet Lorino, vice president of operations, and a few other key staff members "reinvented" the way Automatix creates and manages its products. The process began in late 1992, when Collier and his management team spent several weeks intensively reviewing the competitive situation of their firm. The review had led to unanimous agreement about the firm's future prospects.

"It's not a very pretty picture," Collier had said in summing up the team's analysis one Friday afternoon. "On the one hand, it is clear that we are as capable as any other major automation systems firm. On all key capabilities--technology, marketing, manufacturing, customer service, reputation--we are at least as good as the best in our industry, and we are probably even tops in some critical areas. And on the market side, demand for automation products is growing very rapidly. You would think that under those circumstances we would be doing great. But the truth is, we can only sell our products at profitable prices when they offer the best performance available in each specific market we serve. We have worked very hard to improve our speed in bringing new generations of higher performing products to market, but the improvement in speed to market that we've achieved so far is not good enough to make a clear competitive difference. The top four or five competitors in our industry are all bringing their new products to market as fast as we do. That means that on average we only have the top-performing product in our market about 20% of the time, and so about 80% of the time we are just not able to sell our products at profitable prices."

"I think we would all agree that Automatix has the most committed, hardest working people in our industry, and that our problem is not the result of not trying hard enough. But I believe we should be able to look forward to a better future than the picture we have in front of us now."

In the late 1980s, Automatix had made a successful transition from the "traditional" product development method (a sequence of separate development tasks performed by functional departments like R&D, engineering, manufacturing, and marketing) to a more concurrent process carried out by a multifunctional development team. The move to team-based development had reduced Automatix' product development cycle time to about 30 months.

But as Ed Easterbrook observed in the meeting in late 1992, "Adopting the team approached helped to improve our speed--that we all agree on. But it seems we have hit a plateau again. All our competitors seemed to have implemented team-based processes and again have comparable development cycles. To be honest, I just don't see how we are going to get any faster than we currently are. If we add more people to a development team to try to reduce the cycle time below current levels, that seems to break down the team process and add organizational complexity, and development times actually increase. So at this point, I just can't imagine how we could cut down development time further. We are all working flat out here, as you know Bob, but it does appear that we are not pulling ahead of our competitors."

The result of the assessment of Automatix' competitive situation was that the firm decided to take "a big step forward," as Bob Collier put it, "by stepping back from the situation we are in and taking a fresh look at what we are trying to do and how we are trying to accomplish it." The management team agreed to make a top priority during the next months of rethinking the way Automatix creates and realizes products "from the ground up."

## Industry Background

Industrial automation systems consist of various kinds of sensors, control devices, and a small programmable computer (usually called the "controller") that are configured together to provide programmable, automated control for virtually all types of production and handling processes. Sensor devices of various kinds are mounted at critical locations along a production or materials handling line and sense the condition of each stage or station in the production or handling process. Signals from the sensors are sent to the controller, which refers to an automation software program to decide what actions should be taken at upstream or downstream stages of production to maintain the desired flow of materials, parts, and goods. The controller then sends appropriate instructions to the control devices located at various stations upstream and downstream, and the resulting actions of the control devices serve to maintain the desired flows through the production system. (*Refer to* ***Exhibit 1***.)

With the advent of low-cost, high-speed microprocessors in the 1980s, it became possible to sense, compute, and control many kinds of industrial processes in real time. The steadily increasing speed and capabilities of microprocessors in the

1980s and 1990s made it possible to increase the speed and improve the precision (and thus yields) of many industrial processes. By the early 1980s, industries in the advanced economies had begun the large-scale installation of automation systems, and the market for industrial automation products began to grow explosively, quickly becoming a multibillion dollar global market by the late 1980s.

By the early 1990s, Automatix and four other major automation firms provided approximately 60% of all major industrial automation systems sold globally. Although some smaller firms served niche markets globally for highly specialized automation systems (e.g., for controlling chemical refineries), the five major providers of automation systems provided families of automation systems that could be configured to serve a diverse array of assembly, treating, extruding, forming, finishing, packaging, materials handling, and other industrial processes. The balance of the market for automation products was served by local firms using "generic" automation components to configure unsophisticated control systems on a customer's site.

The technology for automation systems advanced rapidly in the 1980s and 1990s, carried forward by the continuously falling cost and exponentially improving performance of microprocessors, by the development of thousands of different kinds of sensors and control devices for automating virtually every kind of industrial process, and by the development of programming tools for quickly configuring a set of sensors, controls, and the controller to suit the control requirements of a given process. Although many production and materials handling lines had at least some processes in common, the many differences in factory layouts, materials processed, products made, and other customer-specific characteristics meant that each automation system installed was a unique configuration of sensors, controls, and controller routines customized to the specific requirements of each process site.

Competition among the major companies making industrial automation systems is focused on five competitive dimensions:

- Increasing the *speed* of automated processes, which permits higher rates of output from a given production line, and which requires use of the latest (fastest) microprocessors, sensors, and control devices available;

- Offering *new sensors and controls* that extend the reach of automation systems to new kinds of processes, and that improve the precision with which a process can be controlled, resulting in higher yield rates for a given level of output from a process;

- Offering greater *reconfigurability, upgradeability, and scalability (expandability)* of an automation system, thereby enabling faster, easier reconfiguration of processes when plant layouts or equipment change, as well as more precise "fine tuning" of existing processes;

- *Customer support and training* in the use of the automation system;

- *Price.*

Pricing varies critically with the relative performance of an automation system. In general, during the lifetime of a given production system, the economic benefits of increased output and yields that users may realize from an automation system with higher speed and greater precision can be very significant. For this reason, the firm that can offer the highest performing automation system for a given type of production line can usually command a substantial price premium for its product. Firms that can only provide “me-too” automation systems--i.e., systems comparable in performance to those provided by competing firms for a given application--typically face severe price competition to win orders.

**Automatix Analyzes Its Product Creation and Realization Processes**

During early 1993 Bob Collier, Ed Easterbrook, and Janet Lorino led a company-wide, “ground-up” rethinking of the way Automatix creates and realizes products. The result was the emergence and adoption of a radical new approach to creating and realizing products.

After the strategic review meetings in late 1992, Ed Easterbrook began to analyze Automatix’ product creation process to try understand more clearly what aspects of the product development process required the most time and resources to complete. Easterbrook undertook a detailed analysis of four development projects recently completed by the firm--two projects completed before the adoption of the team approach and two projects completed after implementing the team approach. Daily time records representing from 100 to 250 person-years of development time per project were analyzed in detail for each day and for every person to determine how much time was being spent by each employee on each kind of activity in each development project. Development staff involved in the various projects were also interviewed to compile a detailed qualitative and quantitative description of how people had spent their time in the four development processes.

The results of the analyses provided some stunning insights into how Automatix’ staff actually spent their time in developing new products. In *both* the two team-based projects and the two projects following the sequential-functional approach, at least 50%--and in one project over 80%--of total development time was spent making revisions to designs of components that were necessitated by changes made in the designs of other components. Further investigation and interviews revealed why this was happening.

When a product development project was started at Automatix, a set of desired product performance goals was defined and agreed on by management for each new generation of product. Based on those defined performance goals, product development engineers would then create a high-level system design that began with a “block diagram” indicating the kinds of functional components that will be needed in a new product and the performance levels that must be provided by each component to meet overall product performance goals. Flow lines indicating the basic interactions between functional components that have to take place in the eventual product design were then added to create a high-level, schematic “system design” of the product (*refer to **Exhibit 2***).

Automatix’ controllers typically consist of eight basic functional components:

- a microprocessor mounted on a motherboard (printed circuit board) that jointly provide the electronic circuitry that functions as the "brain" of the automation system;
- memory cards (SRAMs and DRAMs) for storing control programs and data;
- ports for receiving inputs from sensor devices and for sending output signals to control devices;
- the programmer for programming the controller to work with a given array of sensors and control devices;
- an operating system (Windows);
- a power supply;
- a bus for mounting and connecting the printed circuit boards containing the microprocessor, memory, and programmer;
- a steel or plastic box containing and protecting the other components.

After defining the basic functions and performance levels to be provided by a new product and the system design for that product, development engineers at Automatix would next focus on developing new components that could provide the functions and performance characteristics required in the new product design. However, Ed Easterbrook's detailed analyses of Automatix development projects established that two factors tended to make component development a complicated, time-consuming process. First, on the market side, during component development new marketing goals or needs would be identified that required (i) a change in the performance level required of a component, (ii) the use of a new technology to meet a new component performance requirement, and/or (iii) the addition of a new type of component to the product (most commonly in the form of a new kind of sensor or control device). Second, on the technology side, engineers would often have ideas for new or improved components that they would want to include in a new product. Some ideas for new components would come from component suppliers who offered Automatix the opportunity to use new kinds of sensors, control devices, or other components based on new technologies.

Easterbrook's analysis established that such changes in market and technology factors typically led to decisions to change either product specifications or specific component designs "mid-stream" in the development process. The decisions would result in changes in the designs of some components, which would then typically require compensating changes in the designs of other interacting components, which might then lead to the need to revise designs of yet other components, and so on in an expanding "chain reaction" of component design changes. Easterbrook's analysis showed that mid-stream changes in product specifications accounted for about half of the total time and resources spent in redesigning components.

The other half of development time spent in revising component designs resulted from the desire of design engineers to include leading edge technologies in their new product designs, typically in the form of new, higher performing component designs based on new technologies. When the system design of a new product included a new component design based on a new technology, however, subsequent full development of the component would often lead to the discovery of unanticipated kinds of interactions between the new component and other components in the product design. For example, in some cases engineers would discover that there was a need to provide a new or revised communication interface between the new component and other components in order to assure their proper functioning together. In other cases, designers would discover that the functioning of a new component design would affect or be affected by the functioning of another component in some unintended--and usually undesired--way. A new microprocessor design might be found to be more sensitive to the heat generated by the power supply in the controller box than the microprocessors the firm had used in earlier product designs. As the technical uncertainty inherent in using a new component design based on a new technology was discovered and resolved during the product development process, designs of interacting components would typically also have to be modified, and the "chain reaction" of redesigns of other interacting components would then take place.

Easterbrook's analysis also helped Automatix to develop new insights into how its management resources and time were being consumed in product development. In general, development staff with expertise in all critical areas of a new product worked as a team to create the initial system design at the beginning of the product development process. Subsequent development work required more specialized expertise to develop new designs for each type of component in the system design. Component development was done by component development groups responsible for the detailed development and design of a specific component. Both the permanent organization of the development and engineering staff in Automatix and the organization of specific development projects reflected these technical component specializations.

The need to redesign a component as the result of a design change in another component raised a number of administrative and budgetary issues that required substantial management attention during product development processes. Typical issues involved deciding which development group's budget the redesign of a component should be charged to--the budget of the development group whose decisions to change their own component design precipitated the need to change the design of the other component, or the budget of the other development group that would then have to make the compensating change in their component design? Similarly, questions were sometimes raised as to how the additional time spent on modifying a component design to satisfy a request from another component development group would affect the performance evaluation of the group making the requested design modification. Further, because the specialized component development groups were often involved in developing new components for several different product development projects at any time, there was often a question of whether priority in the use of development staff should be given to a new product development project currently underway or to revising a component design created for an earlier project. Such issues had come to be know as "interface issues" in Automatix because the chain reactions of design revisions resulted from the way each component has to "interface" with the other components in an overall product design. Easterbrook's analysis of four development projects established that adjudicating interface issues was the principal demand on management time and resources during product development at Automatix.

A similar pattern of time and resources demands was also discovered by Janet Lorino in analyzing the way her operations staff spent their time in creating new processes or modifying existing processes to produce, distribute, and service Automatix' new products. Her analysis showed that the "chain reaction" of design changes in product components precipitated not just changes in the designs of other components in a new product design, but also changes in the design of the process activities or "process components" that would be needed to manufacture, assemble, and service products based on the revised component designs.

## Automatix "Reinvents" Its Product Creation Process

Ed Easterbrook and Janet Lorino's analyses established a new understanding among Automatix' development staff about the root causes of Automatix' current time requirements for bringing new products to market:

(i) The marketing specifications for Automatix' new products were typically a "moving target" that led to mid-stream changes in component specifications and precipitated a chain reaction of time-consuming component redesigns during development.

(ii) Similarly, mid-stream changes in component designs by engineers wanting to include "leading edge technology" also precipitated chain reactions of component redesigns.

(iii) Developing products that include new kinds of components based on new technologies commonly led to "downstream" revisions in component designs, because unanticipated and undesired interactions between the new kind of component and other components would usually only be discovered at some late "downstream" stage in the development process.

(iv) Most management time and resources consumed in product development were spent adjudicating interface issues because Automatix lacked a clear and consistent policy for managing changes in designs of interacting components during product development.

In subsequent discussions, some engineers argued that Automatix' development processes were basically "out of control," and that the substantial development time and resources consumed by frequent reworking of component designs were evidence of that. Other engineers, however, argued that chain reactions of product and process component redesigns were just an inevitable part of product development processes and, although unfortunate, nothing could be done about them.

Eventually, Bill Bright, a senior design engineer at Automatix, made a startling proposal. Bright suggested that once a project team agreed on the system design for a new product, the interface specifications (that define how each component will interact with every other component in the product design) should be "frozen"--i.e., no aspect of a component design that would require a change in the design of another interacting component would be allowed to change after that point. In essence, after freezing the interface specifications for the components in a system design, component development groups would then have to follow a simple "design rule" that all component designs developed by each group have to conform to the interface specifications.

Bright's proposal elicited a number of objections and questions.

One project manager objected that freezing the interfaces would make it difficult or impossible to accommodate changes in product performance specifications requested by the marketing staff. Bright replied that that by trying to be responsive to a changing set of marketing needs *during* each development project, Automatix might actually be slowing down each development project and therefore compromising its overall ability to respond to the market quickly. Bright then took his proposal one step further and proposed another design rule: No changes would be allowed in product specifications after freezing the interface specifications for components in the system design of a new product.

Bright explained his thinking in this way: "Maybe if we didn't allow changes in product specifications once we start a development project, we could finish each development project more quickly. Then we could immediately get started on the next development project to meet a new set of product performance specifications, and then finish that development project a lot more quickly, and so on. In other words, maybe we could respond more effectively to changing market requirements by freezing interface specifications and then carrying out each new development project more quickly, rather than by making product specification changes *during* development that slow down our development processes and make them difficult to manage. In other words, if each development project could become a lot faster and consume fewer resources, we could 'cycle through' a larger number of more frequent development projects, rather than spending a lot of time and resources in relatively infrequent, slow, inefficient development projects."

Some engineers had objected that freezing the component interface specifications would block introductions of new components based on new technologies. Bright then argued that it might be better *not* to try to develop new components based on new technologies during product development, but rather to develop proven designs for new kinds of components in a technology development process that would be separate from product development processes. The object of the technology development process would be to understand fully the "system behavior"of a new component, so that design engineers could specify with confidence exactly how the new component would interact with the other components in Automatix' kind of products *before* the new component is included in a new product design. Bright suggested that managing development of significantly new components "off line" from product development might help Automatix cycle through development projects much faster. If the increase in product development cycle speed was great enough, then the time required to research the system behavior of a new component and then include it in a next-generation product development project might be less than the time required to develop next generation products based on new components whose system behaviors were not yet fully understood.

Bill Bright's proposals created a swirl of debate among Automatix' managers and development teams. Nevertheless, Ed Easterbrook recognized that Bill Bright's suggestion of a radically different way of developing new technologies and new products represented a new logic for managing product development processes--and so might offer benefits that simply might not be obtainable through the firm's current approach to managing development. After consultations with CEO Bob Collier and others, in May 1993 Easterbrook asked Bill Bright to organize a development team that would work in the new way proposed by Bright. The team was given responsibility for developing Automatix' new FPC-100 series of controllers for industrial automation systems--and also for charting a new process for developing new products in Automatix.

## Integrating the Supply Chain into the Modular Development Process

The "FPC" in FPC-100 stands for "flexible process controller," and the FPC-100 product line was intended to provide high levels of flexible reconfigurability in automating smaller production lines that were likely to grow in complexity, undergo modernization, or otherwise be modified during the lifetime of the controller. The FPC-100 series was intended to be a "breakthrough" product that would provide high performance, flexibility, and ease of use at a price low enough to extend the market for sophisticated industrial automation systems to smaller firms and low-budget production operations. As they began their project, the FPC-100 development team recognized the need to make several important decisions, each of which would require inputs from participants in the future supply chain for the FPC-100.

### Considering the Supply Chain when Specifying Component Interfaces

One of the most critical decisions facing the FPC-100 team was the amount of "flexibility" that should be designed into the component interfaces that would be frozen or "standardized" in the FPC-100's system design. The standardized interfaces would have to be flexible enough to allow the ready substitution of all the component variations needed to support FPC-100's strategy of providing flexible reconfigurability. In effect, the standardized interfaces would have to support "plug and play" compatibility among the many different kinds of components needed to configure the FPC-100 to suit the specific requirements of each customer's application. The development team began to refer to the FPC-100 as a "modular" product because it would be designed to allow plug-and-play compatibility among a large set of "mix and match" modules (components).

Specifying interfaces in the modular FPC-100 to achieve the required level and forms of flexibility required that the development team gather some new kinds of marketing and technology information from potential participants in the FPC-100 supply chain. The development team asked their marketing colleagues to consult with customers and the value-added vendors likely to handle the FPC-100 to try to project the full range of functions, features, and performance levels that would be important in meeting the longer-term needs of customers for the FPC-100 series.

Automatix' marketing staff, however, was unused to providing such information to development engineers. In fact, many of the marketing staff considered that their job was to gather information about various market needs and then to interpret those needs to determine the "best" specifications for a new product--i.e., the specifications of a product that would appeal to the largest possible number of customers. In effect, the FPC-100 team was asking the marketing staff to give them a broad picture of customer requirements--without the "value added" of deciding specifically what single set of product specifications would "best" serve those needs. Some of the marketing people even felt that the development team was trying to bypass their expertise and "take over" marketing's traditional function. However, with encouragement from top management, marketing staff provided the requested market information to the development team.

The development team also needed new information about upcoming technologies in order to determine the forms of technical flexibility that should be designed into the modular interface specifications of the FPC-100. Because some useful new kinds of components (mainly microprocessors and new kinds of sensors and control devices) were always under development by Automatix and its various suppliers, the development team asked engineers from internal development projects and from Automatix' key suppliers to provide them with technical descriptions of new or improved components that could become available for use with the FPC-100 in the next three years, which was the expected commercial lifetime of the first generation FPC-100 controllers.

As the FPC-100 team evaluated the forecasts of market needs and technology opportunities for new and improved components, as well as the need to hold costs for the FPC-100 at a low level, they began to speak about the "strategic role" of each kind of component in the product. Each strategic role for components would have to be supported with appropriate kinds of interface specifications. The team identified three strategic roles for components in the FPC-100:

*Performance Drivers.* Some components were referred to as "performance drivers"--components whose performance levels would have to be increased as quickly as possible in the future in order to maintain performance leadership in the product market. The microprocessor in the controller was seen as a key performance leader, for example, because using faster microprocessors allowed both higher speeds and greater precision in controlling production lines and resulted in higher outputs and higher yields in customers' factories.

The development team recognized that interfaces between performance components and other components should be specified to support direct introduction of higher performing components, like faster next-generation microprocessors. This required defining interface specifications that would require component developers to "design in" higher performance capabilities in some components of the FPC-100 in anticipation of their need to work with upgraded, higher performing components in the future. In some cases, interfaces were specified that required components to be designed to interface with components based on more than one kind of technology. Establishing interface specifications that would require components with "redundant" functions and/or performance capabilities beyond those required for the initial release of the FPC-100 added extra material and production costs to those components. However, the development team considered these initial extra costs to be an investment that would enable rapid upgrading of the FPC-100 in the future by permitting direct introduction of new and improved components (like new microprocessors) into the FPC-100 without having to redesign other components. In effect, standardizing interfaces that would require designing future functions and performance capabilities into components today would enable greater speed in bringing new technology to market in the future and would potentially extend the commercial lifetime of the product line.

*Configuration Drivers.* Some components were characterized as "configuration drivers" because they would be configured in many different combinations of components to customize control systems to meet different customer requirements. Sensors and control devices and the programmer for on-site programming of the controller were considered to be key configuration drivers by the development team.

Specifying interfaces for configuration components was especially challenging, because many different kinds of sensors and control devices were likely to become available for use with the FPC-100 during its commercial lifetime. However, there was no uniform "industry standard" interface for connecting sensors and control devices to controllers. Rather, there were three major alternative standards or "protocols" for connecting sensors and control devices made by most firms, as well a number of proprietary protocols for connecting the sensors and controls made by other firms. Moreover, it was impossible to predict exactly the kind of interfaces various suppliers of sensors and controls would use for their components in the future.

Two approaches to managing this dilemma were adopted by the FPC-100 team. The first approach was to adopt as the standard interface for the FPC-100 one of three major "protocols" for connecting sensors and control devices, and then to advise all suppliers of sensors and controls that some important future products under development at Automatix would base their interface specifications for sensors and controls on that protocol. In this way the development team hoped they would be able to influence suppliers of sensors and controls to adopt the protocol and make their components compatible with the sensor and control interface specifications to be used in the FPC-100. The team also decided that, as a last resort, they could create "interface adapters" for connecting important non-conforming sensors and adapters to the FPC-100 in the future.

*Cost Drivers.* Some components were characterized as "cost drivers" because their main contribution to the FPC-100 would be to provide reliable performance at the minimum possible cost. The strategy of the development team was to define "common component designs" for all components that did not require different designs in the various planned versions of the FPC-100. Memory cards, the Windows operating system, the power supply, the bus, and the box would all be used in common in all variations of the FPC-100 shipped. Interfaces between common components could be specified and standardized straightforwardly, because those components would not vary or change during the commercial lifetime of the first generation FPC-100 controllers.

Defining the Process Capabilities of the Supply Chain

A second critical issue facing the FPC-100 team was how to improve coordination between product and process development to significantly speed up time to market after completing product development. In this regard, the team recognized the need to develop the overall process design for producing the FPC-100 concurrently with developing the FPC-100 product design.

Sarah Swift, a production design engineer and member of the FPC-100 development team, defined this issue in this way: “Just as we have to define in the interface specifications how all the components are going to interact in the FPC-100 product design, we also need to define how each component in the FPC-100 is going to interact with each step in the production process. For example, when the microprocessor and its motherboard are designed, we need to be sure that the design can be produced either on our own PCB [printed circuit board] assembly line or on one of our PCB suppliers’ assembly lines. Alternatively, if we really need to have a motherboard that is beyond our current PCB production capabilities, then we need to define as early as possible what new process capabilities we need to start developing now in order to be sure we can assemble the FPC-100 motherboard when development is done and it is ready to be manufactured. If we want to be fast to market, we have to avoid unforeseen delays in starting up manufacture of the FPC-100, and that means we have to systematically define our own current production capabilities and those of our key suppliers, as well as identify the new process capabilities we and they will have ready by the time the FPC-100 is ready to go to production. Only by knowing what we are actually capable of doing now can we get a clear idea of the new process capabilities, if any, we will have to put in place to be able to produce the product you are designing.”

Sarah Swift’s suggestion that the FPC-100 team should ‘look forward’ into Automatix’ and its suppliers’ production capabilities to be sure that they could actually produce all possible configurations that the FPC-100 modular product design would allow was accepted by the development team as a necessary step in assuring speed to market once development was completed. The development team subsequently spent several months working with key internal and external suppliers to define all current process capabilities that could be used to produce, ship, and assemble all components in the FPC-100, as well as the capabilities of downstream logistics and service providers who would have to deliver and support the FPC-100 in the field.

The team then agreed to establish a design rule governing the interactions of the modular components in the FPC-100 with the process capabilities of the supply chain that would have to realize the FPC-100. All component designs created for the FPC-100 were constrained to be manufacturable, shippable, and serviceable using only Automatix’ and its suppliers’ *current* process capabilities. By adopting and following this design rule, the FPC-100 team assured that the progression from final product design to manufacture to shipping and supporting the new product would be fast, with no unexpected delays due to the need to create or redesign a process capability. Subsequently, the FPC-100 team referred to this process as defining and standardizing the "product/process interfaces" governing the interactions between component designs and existing capabilities at each step in Automatix supply chain.

**The Launch of the FPC-100**

Following the formation of the FPC-100 development team in May 1993, the team spent four and a half months working with Automatix' manufacturing people and subcontractors to define the process capabilities available for producing and assembling the components to be used in the FPC-100 series. Development of components for the FPC-100 subsequently took place within the constraints imposed both by the standardized component interface specifications for the product design and by the standardized product/process interface specifications for Automatix' supply chain.

Development of components for the FPC-100 required seven months. In total, design development of the FPC-100 was completed in just eleven and a half months after the formation of the development team. Because all new components were designed to be manufactured and assembled using existing production capabilities, the first FPC-100 units were produced and ready for testing within a few days of the completion of the design phase, just as Sarah Swift had promised. From the beginning of the development project to the production of the first units for testing had taken just a few days less than one year.

Benchtesting of the first FPC-100 units, however, revealed that some component variations did not appear to work as expected when certain other component variations were used in the controller. In short, most--but not all--component variations planned for the FPC-100 proved to be "plug and play" compatible with the other planned component variations.

The engineers on the development team recognized that the unexpected behaviors of some components in the FPC-100 were the result of their own incomplete understanding of how the various components in the FPC-100 would interact when brought together in a physical product. In other words, the unexpected behaviors resulted from a failure of the standardized component interface specifications to adequately anticipate and control the interactions of all the component variations that could be used in the FPC-100. The engineers therefore analyzed each case of unexpected component behavior to determine its cause, and then added or revised component interface specifications to prevent similar problems from occurring in future assemblies of such components. For example, certain kinds of sensors were found to delay sending their signals to the controller by small fractions of a second. These delays led to incorrect representations of the "real time" condition of a step in the production process and resulted in the controller sending inappropriately timed instructions to control devices. The signal interface between such sensors and the controller had to be respecified to compensate for such timing delays.

Benchtesting and debugging the FPC-100 took nearly six months of additional time, but all members of the development team agreed that this process provided important benefits beyond simply improving the reliability of the current FPC-100 design. The process of discovering unexpected component interactions, analyzing the causes of those interactions, and creating new or modified component interface specifications to prevent such unexpected behaviors from happening again was an intensive learning experience for the design engineers. They all believed that analyzing and correcting unexpected component behaviors by defining new interface specifications increased their understanding of the "system behavior" of products like the FPC-100, and greatly improved their ability to control the system behaviors of such products by writing more complete interface specifications for the next generation of such products.

At the end of the debugging process, the FPC was released for full production and marketing. Eighteen months had transpired from the beginning of the FPC-100 development process. Compared to the 30 or so months required to complete previous product development projects of comparable product complexity, the FPC-100 development project achieved a major reduction in time to market for Automatix.

The FPC-100 quickly became a major commercial success for Automatix, establishing both a new market segment for industrial automation systems and a solid reputation for Automatix as the innovator that brought automation to small-scale production systems.

In early 1995, four months after the introduction of the FPC-100, General Controls Corp. (one of the "big five" global industrial automation firms) announced that they would begin shipments before the end of the year of their version of a low-cost automation system for small production lines. Although the General Controls product did not offer the degree of reconfigurability provided by the FPC-100, the product would be brought to market with aggressive pricing that signalled General Controls' intention to compete for market share in the new market segment established by the FPC-100. General Controls' announcement was not entirely unexpected.

"We expected to have competition in this new market segment," Ed Easterbrook explained to CEO Bob Collier, "but we were a little surprised at the timing. We didn't expect such a fast product announcement from General. What we don't know, however--and this is the critical issue for us now--is when General started development of their product. If they saw this product opportunity when we did back in 1993, and if they have taken 30 months to develop their product like we typically did before the FPC-100 process, then that would be consistent with their plans to ship product near the end of this year. On the other hand, it may be possible that they have somehow started to develop products the way we developed the FPC-100, and that they may have started development later--maybe about the time we did."

"What do you think we should do now?" Collier asked.

"It's clear that the new approach we used on the FPC-100 really cut development time significantly," Easterbrook replied, "and part of the original concept was that we would maintain market leadership by 'cycling through' successive development processes quickly. It looks like we better put that theory to the test now. I suggest that we start the second development cycle on the next generation of FPC-100 immediately. We did a lot of groundwork in the FPC-100 project, like defining our process capabilities in detail and really fine-tuning the standardized interfaces. I expect that we can get through the second development cycle several months faster than the first one. It took us about 18 months on the FPC-100, but it's possible we could be closer to 12 months on this next development cycle. If so, that would give us an improved FPC series in the market within three or four months of General's product release. Then we'll see how long it takes General to come to market with their next generation product. That should tell us a lot about how their current product and process management capabilities compare to ours."

**Exhibit 1**
**Configuration of Industrial Automation System**

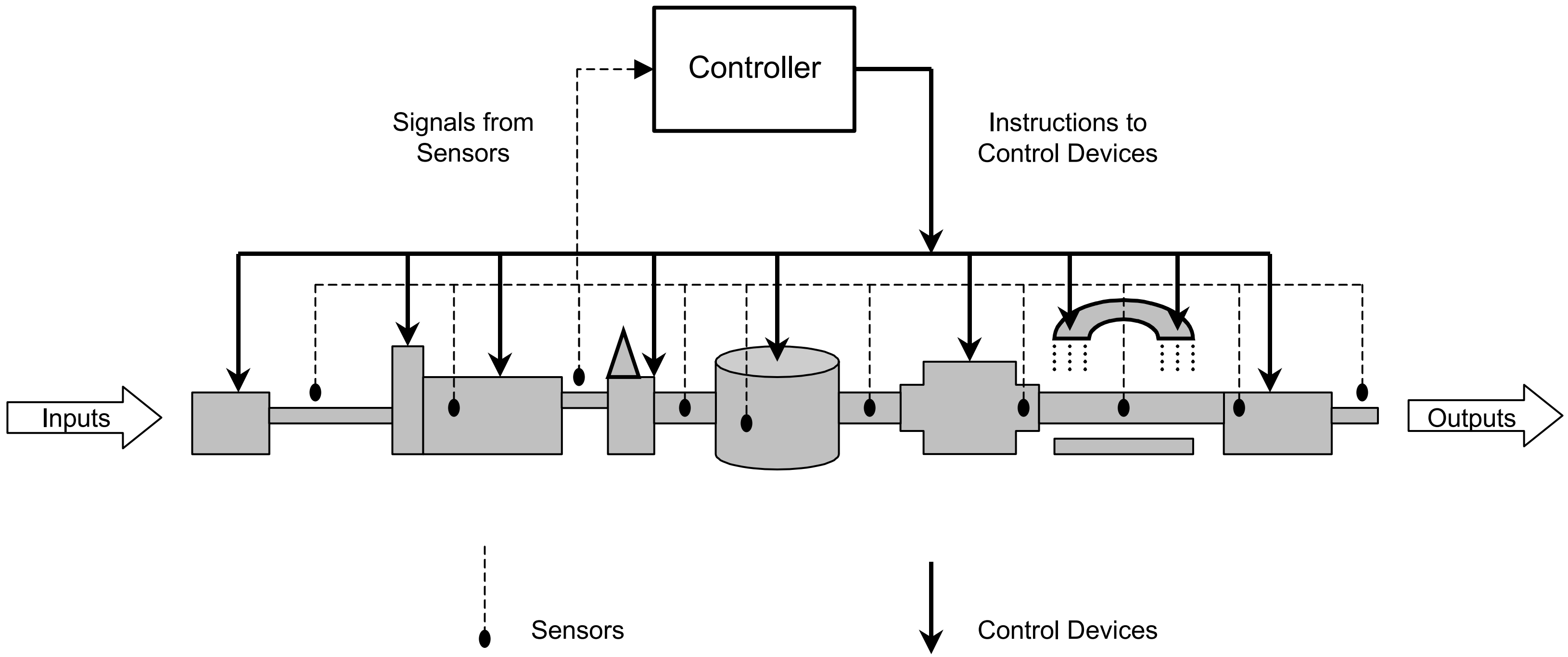

**Exhibit 2**
**Schematic System Design for Controller**