SAE TECHNICAL PAPER SERIES

1999-01-1644

The Production System Design and Deployment Framework

David S. Cochran
Massachusetts Institute of Technology

SAE The Engineering Society For Advancing Mobility Land Sea Air and Space® INTERNATIONAL

International Automotive Manufacturing Conference and Exposition
Detroit, Michigan
May 11-13, 1999

400 Commonwealth Drive, Warrendale, PA 15096-0001 U.S.A. Tel: (724) 776-4841 Fax: (724) 776-5760

ISSN 0148-7191

Printed in USA

1999-01-1644

The Production System Design and Deployment Framework

David S. Cochran
Massachusetts Institute of Technology

ABSTRACT

This session keynote paper presents a framework for designing and deploying production systems. The framework enables the communication and determination of objectives and design solutions from the highest level to the lowest level of a manufacturing enterprise. The design methodology ensures that the physical implementation, called Design Parameters (DPs), meets the objectives or Functional Requirements (FRs) of the production system design. This paper presents a revolutionary approach to determine the objectives and the implementation of a "lean" production system design for a manufacturing business as guided by the design axiom of independence.

Keywords: lean production, information system design, design methodology, cell design

INTRODUCTION

The classic problem in large organizations is that top management sets the general course and the workforce will do it, but the buck seems to stop in the middle. There are a number of complicating factors involved, however, it is certain that middle management must convert the strategic, business objectives into action. The root of the problem is how to convert the corporate mission statement(s) into something that is real, tangible and properly measured on the factory floor? Two questions arise. *What* to communicate. *How* to communicate it.

The Production System Design and Deployment Framework provides a methodology to translate strategic manufacturing objectives into design and implementation actions on the factory floor. The Framework enables the decomposition of a production system design from the strategic level to the implementation level as guided by the idea of independence in design as defined by the independence axiom [Suh, 1990]. Axiomatic design provides the foundation for manufacturing system design and implementation-path dependency, which are two aspects of the Framework [Suh, Cochran, Lima, 1998].

With Axiomatic Design, determining the physical solution or Design Parameter (DP) to achieve the Functional Requirement (FR) develops a design. At each level of a design's decomposition, a design is tested for independence, according to the independence axiom.

The process of zig-zagging between the functional and physical domains to lower levels provides a complete design decomposition of the strategic objectives (FRs) to the lowest level necessary to explain and determine the physical implementation. This type of decomposition is very different from the typical, single-tree design decomposition approaches [Marca, and McGowan, 1993]. The Framework is being used to design, deploy and communicate the objectives of "lean" production systems [Cochran, et al 1998].

The motivation for developing the Production System Design and Deployment Framework is to clearly define objectives (what we want to do) and the corresponding physical implementation (how it will be done). The goal is to provide a means for communicating and deploying a system design to numerous people. The Framework uses the axiomatic design methodology to prevent confusion and the blind and rote application of rules.

System design requires understanding the customer. System objectives/ Functional Requirements (FRs) are determined based on the customers' needs. Implementation Design Parameters (DPs) are next developed to independently achieve the FRs. This approach provides the context for decision making and design-solution development. Design rules, however, may be appropriate in one context and in another context, may not be appropriate at all. Design by rules is an anathema to systems engineering.

System designs can fail under the following conditions:

a. when only the objectives of a system are thought about in the absence of a solution
b. when solutions are developed without tying them to the objectives of a system
c. when the eureka solution or buzz-word-of-the-week is blindly applied without understanding the underlying system objectives (e.g., resulting in JIT warehouses, sequence buffers and other atrocities)
d. when modeling is used to optimize a set of limiting assumptions that are the result of poor system design

The goal of the Production System Design and Deployment Framework is to prevent the occurrence of the above four problem situations, which result in non-holistic system design "solutions."

There is a tendency to believe in the "eureka" solution as captured by the buzz-word-of-the-week. However, just doing "5S", "visual factory" or "JIT" alone is not enough [Monden, 1983]. The buzzwords do not convey the objectives or solutions that are required to design production systems.

The following example illustrates the result of doing (c) at an automotive manufacturing facility that implemented "Just-In-Time" delivery and "Kanban." These were the buzzwords in manufacturing of the 80's. JIT and Kanban were implemented by this facility without addressing the true objectives or production system design problem that these tools were solving. JIT and Kanban were solutions (DPs) looking for a problem to solve. The implementation of these tools was done in the absence of an integrated production and manufacturing system design [Black, 1991]. Figure 1 illustrates a plant design resulting from implementing DPs or tools instead of a system design.

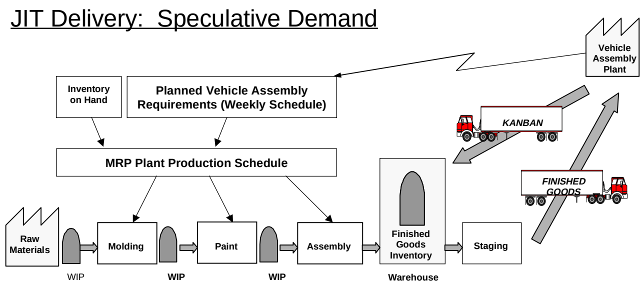

Figure 1. "JIT Warehouse" Due to Implementing Tools Not a Production System Design

The implementation of a Kanban delivery cycle between an automotive components plant and a vehicle assembly plant was done to accomplish the JIT objective (which means to produce the right product, at the right time, with the right quantity). In this case, the automotive components plant had to build a "Just-in-Time Warehouse." This situation explains why many people in industry have claimed that JIT "forces the inventory back on to the supplier." Looking at the above situation, this claim is true. In fact, the above approach has added cost. Why did the components plant have to build a JIT Warehouse?

The tools or how's (DPs) of a system design were copied and partially implemented. The total system design objectives (FRs) were not identified, understood and communicated. The corresponding solutions (DPs) of a complete production system design (which Toyota calls the Toyota Production System, renamed, "lean" production) became the objectives of an implementation program [Womack, Roos, and Jones 1990]. The true objectives or problems that were being solved (the FRs) by implementing the JIT program were lost.

What would have to be changed to truly implement: "producing the right product at the right time, in the right quantity?" First, the equipment must be able to predictably produce at the right pace and the right mix of products as demanded by the customer, with perfect quality.

In this case, the machine for painting had to be re-designed. People were re-integrated into the production system. Paint was re-designed to no be longer a high-speed "island of automation" in the factory. The changeover time of paint was reduced from 1 hour to less than one minute. Furthermore, the painting machine now operates at the pace or takt time of its (single) customer.

In fact, paint was integrated into a volume-flexible, assembly cell. Figure 2 illustrates that paint is now one operation that is integrated with assembly, which operates at the takt time, based on the demand from one customer. The ultimate result of designing the equipment to meet the manufacturing objectives was higher quality, less scrap, less automation, less inventory, on-time delivery without a warehouse and lower investment cost.

Secondly, to achieve the minimum inventory objective of true JIT there must be information to define what product to produce and when to produce it. This information must be based on actual consumption demand from the customer.

The information system, as shown in Figure 1, does define what and when to produce. The question is "based on what?" and "how often is the information sent?" In Figure 1, the plant's MRP (Material Requirements Planning) computer system defines what to produce based on the planned build of forecast build from vehicle assembly. Then based on the state (inventory counts) in the production that are collected on a daily, weekly and sometimes monthly basis, the computer *automatically* calculates a time window (the when) quantity and mix of products to produce.

The problem is as soon as a new MRP plan is generated; the information on which it is based has changed. In fact, the planned build will most certainly be wrong. The plan is obsolete the minute it is generated since the state of the inventory count will change.

A system is not controllable when its information is erroneous or late. In most MRP environments, both of these conditions are true. The time interval to sample the state of the system is much greater than the time interval necessary to control the system. This condition is analogous to driving a car and a car's braking system. What if, when driving a car, it took one day for the brake to engage the wheel's rotor after pressure is applied to the brake? One would most certainly crash! Attempting to control a production system with MRP/ERP is analogous to this situation.

To summarize, there are two major problems with the plant information system design represented by Figure 1. First, production is not based on actual consumption from vehicle assembly. Second, the system is not controllable.

The JIT Warehouse in Figure 1 is a superficial solution to deal with the inadequacies of the information system, the sub-system (cell) design/definition and the equipment design. Figure 2 illustrates the new production system design. The information system, the definition of sub-systems and the equipment design have changed.

To control any system, the state of the system must be sampled in a time interval that is fast enough to initiate controllability. Therefore, an objective of production system design is to be able to sample the system state based on a minimal time interval. Toyota calls its system state sampling time interval the "pitch."

The idea is to make the pitch or system sampling time interval equal to Takt Time times the Container Size.

Pitch = Takt Time * Container Size

Therefore, the pace of production and problems in production are fed back on a container-by-container basis (e.g., in the time interval that corresponds to the time to produce one container). When the brake is applied in this system, problems are recognized at the time interval corresponding to the pitch of the system. Instead of daily or weekly recognition of problems, problem recognition corresponds to the time required to make a standard container size of parts. Since the system's recognition of problem conditions does not significantly lag the need to make changes, the system is controllable. Production, therefore, can eventually achieve perfect quality, predictability in output (rate) and production based on actual demand… which are high-level FRs of the Production System Design.

By implementing "Heijunka" in combination with pull information in the reverse direction of material flow, and installing capacity for a finite set of customers based on takt time, a total systems conversion took place with the new plant design shown in Figure 2. A new, slower takt time was established based on one or a limited set of customers. Volume flexible cells, where the workers are separated from the machines were designed. The JIT warehouse was eliminated!

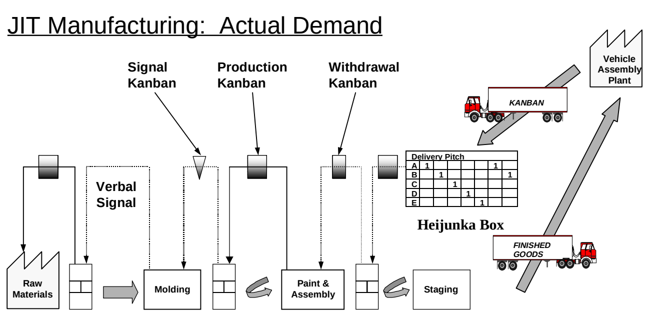

Figure 2. Information Flow and Equipment Re-design to Achieve Production System Design Objectives

Credit for this conversion goes to the true knowledge and understanding of the objectives of the Toyota Production System (TPS) design, not buzz-word implementation. This change required a new information system combined with a totally new approach to machine and cell design based on takt time.

So why haven't people who have seen the TPS in Japan understood its many facets? Why has lean manufacturing been partially implemented or implemented as a set of best practices? It has been explained to me that, "you understand what you see." In this case, people implemented only facets of a system design.

Partial implementation and best practice implementation (like set up time reduction) are analogous. Organizationally, few people really understand the problem that is being solved by the best practice. The general consensus is that it (the best practice) is generally regarded as "a good thing," but is not taken too seriously.

WHY USE THE FRAMEWORK?

Why is the Production System Design Framework proposed in this paper so important? It communicates the elements of a multi-faceted system design in a logical and systematic way that can be communicated throughout an organization. Most importantly, the axiomatic design foundation on which it is based provides the scientific and theoretical basis to design and re-design any production system. Figure 3 illustrates the Production System Design and Design Deployment Framework. The objective of this framework is to provide a production system design and deployment approach that is ultimately product independent. This means that this framework can reflect a production system design that is applicable to any discrete-part production environment, independent of production volume and product type.

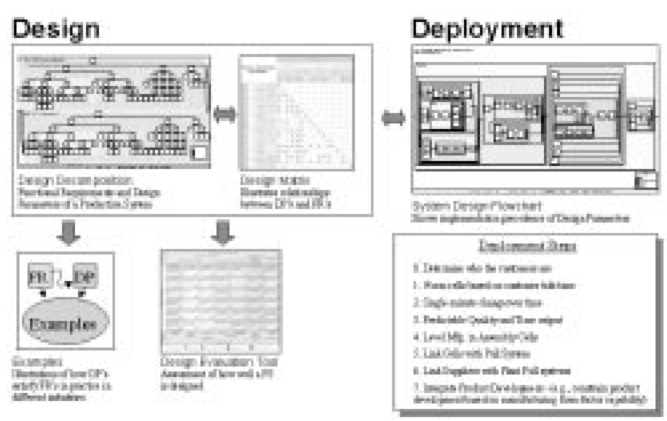

Figure 3. Production System Design and Deployment Framework

THE ELEMENTS OF THE FRAMEWORK

The Framework consists of the following elements:

- The Production System Design (PSD) Decomposition
- The PSD FR-DP Examples
- The PSD Matrix

- The PSD Flowchart and Deployment Steps for implementation
- The PSD Evaluation Tool

The PSD Decomposition (see Figure 4) identifies the design relationships to achieve a “lean” production system design. It provides a systematic means of identifying the objectives (FRs) and the corresponding implementation (DPs) from the strategic business level to the subsystem (cell) and machine design levels. In other words, the decomposition identifies the why of production system design, not just the how that Toyota has implemented. Shingo, a key engineer who taught and developed the Toyota Production System (TPS), emphasized to know why, not just how [Shingo, 1989]. The PSD Decomposition approach provides the foundation to develop a next generation system design.

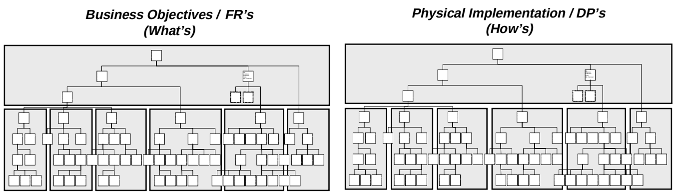

Figure 4. Functional Requirement - Design Parameter Decomposition Hierarchy [Cochran, et al., 1998]

The PSD FR-DP Examples (see Figure 5) illustrate the design relationships identified by the PSD Decomposition. The FR-DP examples illustrate design relationships that are postulated to be product independent in the automotive components industry and other industries. The FR-DP design relationships have been found to be equally applicable to the automotive, aircraft and consumer products industries [Reynal and Cochran, 1998].

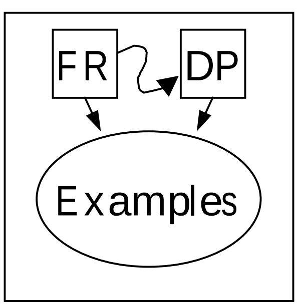

Figure 5. Illustration of How DPs satisfy FRs in Practice in Different Industries

The PSD Matrix that is illustrated in Figure 3 is a composite FR-DP design-relationship matrix. The matrix is the result of illustrating the design relationships to the fourth level of the design decomposition. The axiomatic design foundation used to develop the PSD Decomposition requires that functional independence is maintained [Suh, 1990]. Independence is determined by the form of the design relationship matrix at each level of the design. The goal of design is complete independence, which is represented a diagonal FR-DP relationship matrix.

A workable situation is a quasi-coupled (or a decoupled) design. This type of design is path dependent. The FR-DP matrix at each level of the design decomposition is either an upper <u>or</u> lower triangular matrix. In a path-dependent design, the DP that affects the most FRs is implemented first. The less interaction the better. The knowledge of path-dependent designs leads to a controlled design implementation path.

The composite matrix in Figure 6 shows that a Production System Design is a highly path-dependent design. Therefore, the implementation steps are said to be path dependent. The result of a path-dependent design is that the order of implementation does matter (and that other steps can be taken simultaneously). The implication, of course, is that if the proper steps aren’t taken in the system design implementation, the implementation work is just muda itself.

Figure 6. Composite Design Matrix

The PSD Flowchart (see Figure 7) is a graphical representation of the system design architecture [Suh, Cochran, Lima, 1998]. The flowchart is derived from the design matrix. The flowchart graphically represents the path-dependent design information shown in the design matrix. The implementation precedence is graphically displayed and is clear to everyone. Furthermore, the logic and reasoning for this implementation path is easily understood as a result of its derivation from the design decomposition and the design matrix.

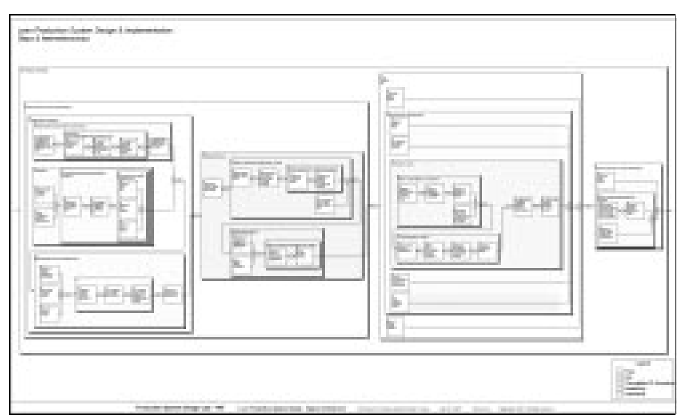

Figure 7. The Production System Design Flow-Chart

The PSD Diagnostic (see Figure 8) is derived from the PSD Decomposition. It evaluates to what degree a manufacturing concern has really designed or implemented the DPs of the PSD Decomposition. In this case, the fourth-level FR-DP relationships are evaluated. A six level rating scale is used. Level 1 represents ugly, “mass” production. Level 3 represents the Ford system circa 1920. Level 5 represents very good “lean” design implementation. Level 6 represents implementation perfection. Therefore, the diagnostic provides a mechanism to measure the implementation adequacy of new system designs based on the system design represented by the design decomposition. On one sheet of paper, it is possible to know where you are and where you want to be.

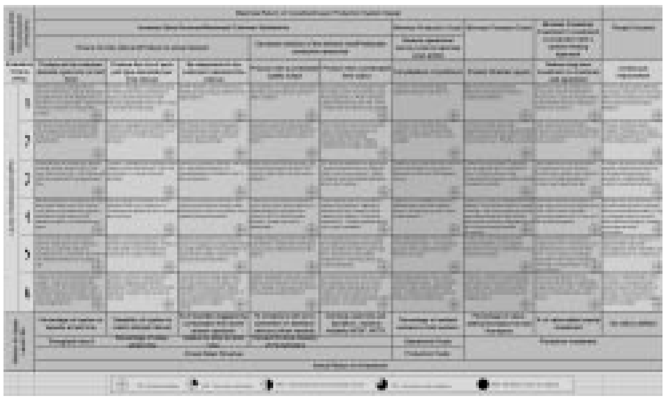

Figure 8. Production System Design Diagnostic Tool

SYSTEM DESIGN - INTRODUCTION

In order to distinguish the difference between a Production System and a Manufacturing System, it is first necessary to define what is meant by a system. A system has definite inputs and outputs and acts on its inputs to produce a desired output [Parnaby, 1979]. Furthermore, a system is comprised of many sub-systems. The interaction between these sub-systems affects the output of the system as a whole. The sub-systems must act as an integrated whole to produce the desired result.

A Manufacturing System consists of the arrangement and operation of machines, tools, material, people and information to produce a value-added physical, informational or service product whose success and cost is characterized by measurable parameters. The Production System consists of all of the elements and functions that support the manufacturing system.

The Production System Design (PSD) includes the design of the performance measurement system and supporting elements of the manufacturing system. The production system defines the measurable parameters that the manufacturing system must achieve. Production system design, therefore, must consider the methodologies that are needed to cost-justify new equipment. The PSD encompasses and includes the Manufacturing System Design (MSD) and predicates overall design effectiveness.

There are four types of operations in any manufacturing system: transport, storage, inspection and processing. Optimizing operations means to improve one of these elements at a time. Improvement of operations in most cases does not lead to improvement of the system [Shingo, 1989]. Improving system performance requires understanding the value of each operation and its interaction with other operations.

Few operations are value adding. Transport, storage, and inspection do not add value to a product. However, sometimes these operations are necessary. These non-value-adding operations must be reduced and eliminated in the context of the entire production system design.

Most factory optimization work is limited to the design and improvement of operations [Sohlenius, 1998]. Very little emphasis is placed on the design and improvement of the manufacturing system, called the value stream [Womack and Jones, 1996]. In addition, most equipment design is operation-improvement focused, not system-improvement focused. Many factories brag about having the world’s fastest and largest machine of a certain type, which evidences this thinking. System design requires designing equipment (which is part of the processing operation) in the context of the system in which it is operated.

A central theme of production system design is that, you get what you measure. There is strong evidence based on the operations-focused cost justification approach in mass production plants that causes a departmental plant structure. The cost justification approach results in the layout of equipment into departments and the development of the highest speed machinery possible [Cochran, et al 1998].

Production system design is the anti-thesis of optimization. Design requires changing variables that are heretofore thought to be unchangeable in many optimization models. System-wide improvements are not made since there has not been a way to rationally design production systems [Van Brussel, et al., 1993]. In some instances, limiting assumptions are made regarding the nature of manufacturing problems. Consequently, some of the manufacturing research has not made much impact on industry.

AXIOMATIC DESIGN AND THE FRAMEWORK

The development of the Production System Design Decomposition is based on the power of axiomatic design. Three elements of axiomatic design that are significant in the development of the Production System Design Decomposition are the concept of design domains, the Independence Axiom, and the idea of “zig-zagging” [Suh, 1990].

Design involves a continuous interaction between what we want to achieve (objectives) and how we achieve it (physical solution) [Suh, 1990]. The objectives of a design are stated in the functional domain and are called functional requirements (FRs), whereas the physical solutions are generated in the physical domain and are called design parameters (DPs), as illustrated in Figure 9. The design process involves selecting DPs that independently satisfy the FRs.

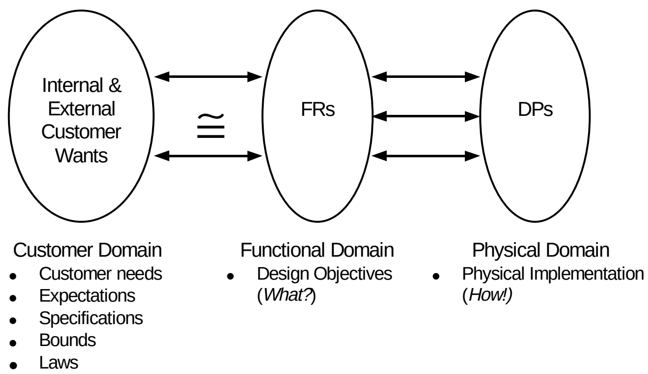

Figure 9. The Axiomatic Design Domains [Suh, 1990]

The Independence Axiom guides the design process. The FRs must be stated independently of other FRs. An adjustment of a design parameter should only affect its corresponding functional requirement. In this way, the independence of the FRs is determined. Therefore, the design approach requires a designer to find one and only one solution (design parameter) to meet a given objective or functional requirement.

The idea of zig-zagging means that any design, no matter how complex, may be decomposed into its constituent levels. In the design decomposition, the process of zig-zagging establishes a design hierarchy of objectives and solutions as shown in Figure 10. The zig-zagging implies that the selection of a design parameter at a higher level establishes the context for the determination of the next lower level of functional requirements. The benefit of a design hierarchy is to enable the decomposition of higher-level requirements in such a way that it is possible to define the relevance of the lower-level design in meeting the higher-level requirements.

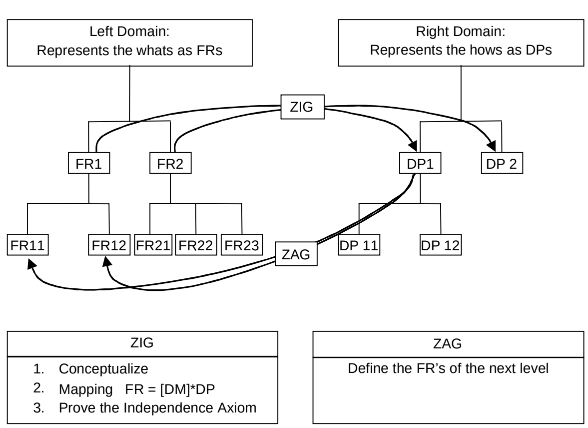

Figure 10. Zig-Zagging

The decomposition of a production system design provides the fundamental thinking and methodology for developing effective system designs.

Benefits of the Production System Design Decomposition:

- Ability to concretely describe and distinguish between various production system design concepts
- Adaptability to different products and manufacturing environments
- Ability to design or create new system designs to meet new requirements (e.g., to determine a new design when the FRs or DPs change)
- Portability of a Production System Design Methodology across industries (e.g., auto to aircraft to food industry)
- Indicates the impact of lower-level design decisions on total system performance
- Provides the foundation for developing a new set of manufacturing performance measures from a system-design perspective
- Makes the connection between machine design requirements and system objectives.

The power and postulate of the production system design approach is that the decomposition is applicable to various product types and is volume independent. Therefore, it is proposed that the design decomposition be equally apropos to automotive or lower-volume aircraft production, for example.

The scope of a production system design must consider the manufacture of a variety of products, at the lowest total cost, with the highest quality, delivered on time to the customer within the customer's expected lead time. The Production System Design Decomposition captures these requirements for repetitive, discrete-part manufacturing environments.

CHOOSE FRs IN THE FUNCTIONAL DOMAIN – The first step in designing a production system is to determine the highest level (FRs). The customer attributes must be mapped to the functional design domain and represented by (FRs). There can be many different possible sets of (FRs) one may wish to satisfy in designing a production system.

The Production System Design Decomposition starts with the highest-level FR stated as:

FR1 = Maximize the return on investment (ROI).

FR1 is derived based on the viewpoint of the owners or shareholders of a manufacturing company. It is interesting to note that there are three very different customers of production systems: the stockholders (owners), the employees (internal customers) and the final customers of the products produced by the system (external customers). The design decomposition starts with the point of view of the shareholders as the decomposition is developed the lower-level FRs reflect the wants of the internal and external customers.

It should be noted that all FRs are stated with a verb_noun structure. DPs are stated with a noun_verb structure. FRs describes something that must be accomplished, while DPs represent physical objects.

DETERMINATION OF DPS IN THE PHYSICAL DOMAIN – The second design decomposition step is to determine the design parameters (DPs) that can satisfy the FRs at the corresponding level in the decomposition tree. For a given set of FRs, there can be many different design solutions, as defined by the DPs. The following example represents this design and selection process. For example, to satisfy FR1, we may choose one of the following two DPs as the corresponding DP1:

DP1a = Minimum cost production

DP1b = “Lean” production system design

The consequence of choosing DP1a rather than DP1b (or vice versa) is quite significant.

RETURN TO THE NEXT LOWER LEVEL OF THE FUNCTIONAL DOMAIN – Having defined the functional requirements and design parameters FR1 and DP1 at the highest level, the next step in axiomatic design is to go back (i.e., "zigzag") to the functional domain from the physical domain. If the chosen DP can be implemented without further detailed design, there is no need to go back to the functional domain.

The lower level FRs must be determined by decomposing FR1, which is equivalent to determining the functional requirements of the DP1 chosen (i.e., either DP1a or DP1b). If the designer had chosen DP1a, the corresponding FRs of the next level may be different than when DP1b is chosen. FR1 may be decomposed after selecting DP1a or DP1b by:

FR11 = Increase the sales revenue

FR12 = Minimize the manufacturing cost

FR13 = Minimize manufacturing investment

These functional requirements are derived from the formula that calculates return on investment (ROI).

$$ROI = \frac{Sales - Cost}{Investment} \qquad (1)$$

FIND THE CORRESPONDING DP1x's BY MAPPING FR1x's IN THE PHYSICAL DOMAIN – Now we have to find DP1x's that correspond to FR11, FR12 and FR13. These DP1x's are also the decomposition products of DP1a or DP1b. Therefore, the DP1x's may be different depending on whether DP1a or DP1b are chosen.

For DP1a, the decomposition of the next level of the DP hierarchy may be stated based on the manufacturing situation in 1915 when mass production systems were evolving:

DP1a1 = Production output as high as possible

DP1a2 = Unit cost minimized

DP1a3 = Machines run all the time (high utilization)

The design above was developed when there was unlimited demand for a low cost automobile. To control cost, the mass production system started with these DPs. Further decomposition of DP1a2 explains why mass production systems are measured and configured the way they are today.

DP1b reflects today’s environment. Today’s source of “economic authority” is no longer the producer but the consumer [Petzinger, 99]. The corresponding DP1bx's must be chosen in line with the customers’ desires:

DP1b1 = Production to maximize customer satisfaction

DP1b2 = Target production cost

DP1b3 = Investment with a systems-thinking approach

This second type production system is designed to increase the sales revenue while decreasing cost and investment by making products that customers want to have -- when they want to have them. These are key elements of customer satisfaction in today’s manufacturing environment.

DETERMINING THE DESIGN RELATIONSHIPS – Having determined FR1x's and DP1x's for two different DP1s, the design matrix must be determined to establish whether the proposed design satisfy the Independence Axiom. The Independence Axiom requires that functional independence be maintained. This statement means that, ideally, DPs are chosen to affect only one FR. A diagonal matrix defines this condition and is called an uncoupled design.

The design equation for alternative (a) in 1915 was as follows:

$$\begin{Bmatrix} FR11 \\ FR13 \\ FR12 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} DP1^a1 \\ DP1^a3 \\ DP1^a2 \end{Bmatrix} \qquad (2)$$

This design represents the mass production system design for the automotive industry in 1915, during the mechanical era. The X signifies a strong relationship between the FRs and DPs.

A triangular matrix defines this design relationship. This design is called a decoupled or quasi-coupled design; it manifests either an upper or lower-triangular matrix. This type of design is path-dependent design. Path-dependent designs mean that the implementation order of the DPs is relevant and significant.

In 1915, the order of implementation was DP1a1, DP1a3, then DP1a2. To increase sales revenue, the factory DP was to produce more products. This DP resulted in division of labor, scientific management and labor unrest due to the de-skilling of jobs. Also, the now classic statement that, "you can purchase any color vehicle as long as it is black," was actually the result of the tremendous pressure to produce more products (i.e., decrease the operation cycle time) in the Ford factory [Womack, Roos, and Jones, 1991; Arnold and Faurote, 1915]. Black paint dried faster than any other paint color.

Secondly, high machine utilization (DP1a3) justified large capital investments to eliminate direct labor as a result of economies of scale. The thought was that investment efficiency was achieved by keeping the machines running. Of course, when the demand for your product is unlimited, this premise is true. In addition, DP1a3 had a strong affect on manufacturing cost. The machines were designed to reduce as fast possible to reduce direct labor content per unit (which was over 80% in 1915). The design assumed that one person tended one station or one machine.

A second design relationship matrix reflects the relationship between the alternative (a), FRs and DPs in today's manufacturing environment:

$$\begin{Bmatrix} FR11 \\ FR13 \\ FR12 \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} DP1^a1 \\ DP1^a3 \\ DP1^a2 \end{Bmatrix} \qquad (3)$$

This design matrix illustrates that system designs change over time. DP1a1 no longer satisfies FR11. Producing more products today does not guarantee that a product will be bought. When the FRs are not satisfied, the design is said to be incomplete.

Similarly, it can be shown that the second design, alternative (b), represented by DP1bx's satisfies FR1x.

$$\begin{Bmatrix} FR11 \\ FR12 \\ FR13 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} DP1^b1 \\ DP1^b2 \\ DP1^b3 \end{Bmatrix} \qquad (4)$$

The comparison of design matrix (3) and (4) illustrates that DP1bx's implements a relevant design solution in today's manufacturing environment.

After the design is completed, the X's in the design matrices can be replaced with precise expressions or constants through modeling of the physics or geometry of the design. The modeling is done for the lowest level of the decomposition (called "leaves"). The higher level design equations are made up of the lower level design parameters and matrices [Kim, et al, 1991].

Figure 4 shows the Production System Design Decomposition of the "lean" production system design that resulted from the design process described above.

At each level in the Production System Design Decomposition, the Design Matrix is used to determine whether the design is adequate before further decomposition can occur. An uncoupled design is the best design, although a decoupled or quasi-coupled design is acceptable. These two classes of designs allow further decomposition.

Figure 6 illustrates a composite of the design matrix for four (4) levels of the decomposition. From Figure 6 we see that the design is a decoupled design, which is, by definition, implementation path-dependent design. A path-dependent design means that there is a specific order of DP (design parameter) implementation. This is a significant result as it tells us what items must be worked on first when implementing a production system design.

CONCLUSIONS

This paper has presented a Production System Design and Deployment Framework that is based on the scientific design foundation of axiomatic design. The Framework has 5 elements: the design decomposition, the design matrices (which are part of the decomposition process), the implementation flowchart (which is derived from the design matrix), FR-DP relationship examples to describe the decomposition's applicability to various discrete-part manufacturing industries, and the plant design diagnostic (derived from the design decomposition). The Production System Design approach has the purpose of eliminating the buzz-word-ology that plagues communication and understanding in industry. Just as there are scientific axioms, laws and relationships, the Framework seeks to provide a methodical way to think, make decisions, and to communicate and convey ideas. The Framework provides the foundation for Production System Design and Deployment with many industries.

Next steps in the research are to determine the applicability of the FR-DP relationships for various industries and applications and to interpret the implementation flowchart for various industries.

This paper has presented the foundation for the five (5) design and application papers that follow in this conference. These papers are testimony to the applicability and flexibility of this approach in production systems engineering.

ACKNOWLEDGMENTS

Thanks go to Professor J Black for introducing me to the dynamic field of Integrated Production and Manufacturing Systems Engineering. I would also like to thank Professor Nam P. Suh for his continuing support, collaboration and amplification of axiomatic design theory with me. I would also like to express my gratitude to Prof. Paulo Lima, Y. S. Kim, and Andrew Wang for their help with the figures and to all of my PSD students for their work to, “change the world one plant at time” with me.

Special thanks to Mr. Charles Szuluk, Mr. Bob Womac, Mr. Ray Schaffart, Mr. Bill Flatt, Mr. Al Tervalon, Mr. Cliff Dawson, Mr. Juergen Weber, Mr. John Barkley, and Mr. Steve Delaney of Visteon for sponsoring, supporting and testing this work. I also thank the leadership in the Aircraft/Aerospace industries for helping us to refine this work, as well.

REFERENCES

1. Arnold, H.L. and Faurote, F. L., 1915, Ford Methods and the Ford Shops, J. J. Little and Ives Co., New York.
2. Black, J. T., 1991, The design of the Factory with a Future, McGraw Hill, Inc., New York.
3. Carrus, B. J., and Cochran, D. S., 1998, Application of a Design Methodology for Production Systems, Annals of the 2nd International Conference on Engineering Design and Automation, Maui, HI
4. Cochran, D. S., 1994, The Design and Control of Manufacturing Systems, Ph.D. Thesis, Auburn University
5. Cochran, D. S., Dawson, C., (PSD Lab - Duda, J., Wang, A., Dobbs, D., Reynal, V., Kim, Y., Linck, J.) and (Visteon - Jordan, N., Boewe, C.), MIT-Visteon “Lean” PSD Course, Dearborn, June 22-24, 1998.
6. Cochran, D. S. and Lima, P.C., 1998a, Production System Design: Theory, Evaluation and Implementation, (in preparation)
7. Cochran, D. S. and Lima, P.C., 1998b, The Production System Design Decomposition, Version 4.2, unpublished report, MIT Production System Design Group
8. Cochran, D. S., Roschmann, H., Weidemann, M., and Kim, Y., 1998, “Production System Design Guideline”, internal document, Bosch Corp. Charleston.
9. Ham, I., Hitomi, K., and Yoshida, T., 1985, Group Technology: Application to Production Management, Kluwer Academic Publishers, Hingham, MA
10. Kaplan, R. S. and Cooper, R., 1997, Cost and Effect Using Integrated Cost Systems to Drive Profitability and Performance, HBS Press, Cambridge, MA.
11. Kim, S. J., Suh, N. P., and Kim, S. K., 1991, Design of Software Systems Based on Axiomatic Design, Annals of CIRP. Vol. 40, pp165-170.
12. Marca, D. A., and McGowan, C. L., 1993, IDEF0/SADT, Eclectic Solutions, San Diego, CA. ISBN 0-9638750-0-0
13. Monden, Y., The Toyota Production System, IIE Press, Norcross, GA, 1983.
14. Parnaby, J., 1979, Concept of a Manufacturing System, International Journal of Production Research, Vol. 17, No. 2, pp. 123-135.
15. Petzinger, T., Wall Street Journal, Feb 26, 1999
16. Reynal, V. A., D. S. Cochran supervisor, Production System Design and Its Implementation in the Automotive and Aircraft Industry, M.S. Thesis, MIT, 1998.
17. Shingo, S., 1989, The Toyota Production System from an Industrial Engineering Viewpoint, Productivity Press, Portland, OR.
18. Sohlenius, G., 1998, The Productivity of Manufacturing through Manufacturing System Design, KTH working paper.
19. Suh, N. P., 1990, The Principles of Design, Oxford University Press ISBN 0-19-504345-6.
20. Suh, N. P., 1995, Axiomatic Design of Mechanical Systems, Special 50th Anniversary Design Issue Transactions of the ASME, Vol. 117, pp. 2-10
21. Suh, N. P., 1995(a), Designing-in of Quality Through Axiomatic Design, IEEE Transactions on Reliability, Vol. 44, No. 2, pp. 256-264
22. Suh, N. P., 1995(b), Design and Operation of Large Systems, Journal of Manufacturing Systems, Vol. 14, No.3, pp. 203-213
23. Suh, N. P., 1997, Design of Systems, Annals of 47th General Assembly of CIRP, Vol. 47/1/1997.
24. Suh, N. P., Cochran, D. S. and Lima, P.C., 1998, Manufacturing System Design, Annals of 48th General Assembly of CIRP, Vol. 47/2/1998, pp. 627-639.
25. Van Brussel, H., Peng, Y. and Valckenaers, P., 1993, Modeling Flexible Manufacturing Systems Based on Petri Nets, CIRP Annals, Vol. 42, No. 1, pp. 479-484.
26. Womack, J. P. and Jones, D., 1996, Lean thinking: banish waste and create wealth in your organization, Simon & Schuster Inc., New York.
27. Womack, J. P., Roos, D. and Jones, D., 1990, The Machine that Changed the World, Rawson Associates, New York
28. Yien, J. T. S., 1998, Manufacturing System Design Methodology, Ph.D. Thesis, Hong Kong University of Science and Technology

CONTACT

Professor Cochran is an Assistant Professor of Mechanical Engineering at MIT. He has established the Production System Design Lab group, which is part of the Laboratory for Manufacturing and Productivity at MIT. Professor Cochran teaches students and industry alike in Production System Design. The lab’s web site is http://web.mit.edu/psd/www/. His email is dcochran@mit.edu.