

1999

## Simulation study of a kanban-controlled production system

Arvind R. Krishnappa  
*West Virginia University*

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

---

### Recommended Citation

Krishnappa, Arvind R., "Simulation study of a kanban-controlled production system" (1999). *Graduate Theses, Dissertations, and Problem Reports*. 950.  
<https://researchrepository.wvu.edu/etd/950>

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact [researchrepository@mail.wvu.edu](mailto:researchrepository@mail.wvu.edu).

**SIMULATION STUDY OF A  
KANBAN CONTROLLED PRODUCTION SYSTEM**

**Arvind R. Krishnappa**

**Thesis submitted to the  
College of Engineering and Mineral Resources at  
West Virginia University  
in partial fulfillment of the requirements  
for the degree of**

**Master of Science  
in  
Industrial and Management Systems Engineering**

**Wafik Iskander, Ph.D., Chair  
Majid Jaraiedi, Ph.D.  
Robert Creese, Ph.D.**

**Department of Industrial and Management Systems Engineering**

**Morgantown, West Virginia  
1999**

**Keywords: JIT, Production Planning, Kanbans, Simulation**

*To my parents*

## **ACKNOWLEDGMENTS**

I would like to sincerely thank Professor Wafik Iskander of the Department of Industrial and Management Systems Engineering, for providing valuable advice, guidance and encouragement throughout my graduate studies at West Virginia University. As my advisor and committee chairman, his patience, suggestions and reviews certainly made the completion of this thesis possible. I would also like to extend my thanks to my committee members Professor Majid Jaraiedi and Professor Robert Creese, both of the Department of Industrial and Management Systems who made valuable technical suggestions during the course of this research.

Special thanks to my parents without whose love and support none of this would have been possible. This work is dedicated to them with love. I would also like to thank my wife for her love, encouragement, editorial and typing assistance. Finally I wish to thank my sisters for their support and encouragement.

# Table Of Contents

	<b>Page</b>
Dedication	ii
Acknowledgements	iii
List of Tables	viii
List of Figures	ix
Chapter:	
1. Introduction and Statement of The Problem	1
1.1 Just-In-Time (JIT) Systems	1
1.2 Statement of the Problem	2
1.3 Research Objectives	5
1.4 Research Approach	6
2. Just-In-Time Philosophy	8
2.1 Key concepts in Toyota Production System	9
2.2 Just-In-Time Production	10
2.3 Autonomation	11
2.4 Flexible Work Force and Originality & Ingenuity	11
2.5 Improvement Activities	11
3. Kanban System	13
3.1 Pulling System for JIT production	13
3.2 What is a Kanban	14
3.3 Kanban Rules	15
3.4 Operation of Kanbans	15

4 Literature Review	18
4.1 Kanban Related Literature	18
4.1.1 Optimization models	18
4.1.2 Markov Models	20
4.1.3 Variations of the kanban system	22
4.1.4 Inventory focused studies	23
4.1.5 Simulation Studies in JIT production	25
4.1.5.1 Model configuration	26
4.1.5.2 Simulation Language used	27
4.1.5.3 Distribution & Random Variables used	27
4.1.5.4 Design of Experiment	28
4.1.5.5 Measures of Performance	31
4.2 Global Productivity Comparison	33
4.3 Comparison of JIT with other production control methods	34
4.4 Literature related to JIT practices in industry	36
4.5 JIT purchasing	40
5. Model Development	41
5.1 JIT System Description	41
5.2 Sequencing Products on a mixed-model assembly line	45
5.3 Model Inputs	48
5.3.1 Master Production Schedule	50
5.3.2 Processing Times	51
5.3.3 Quality Defectives & Machine Breakdowns	53
5.4 Simulation Model	54
5.4.1 Simulation Language	54
5.4.2 Simulation Model Characteristics & Assumptions	54
5.4.3 Elements of the Simulation Model	56
5.4.3.1 Initialization & Sequencing of Products	57

5.4.3.2 Demand Generation on the final assembly line	58
5.4.3.3 Machine Breakdowns & Repair	58
5.4.3.4 Production Ordering Kanbans	59
5.4.3.5 Withdrawal Kanbans	59
5.4.3.6 End of a Shift	60
5.4.4 Model Testing & Verification	61
5.5 Design of Experiment	61
6. Analysis Of Results	63
6.1 Time in System	63
6.2 Work in Process	71
6.3 Stockout	79
6.4 Kanban Waiting Times	83
6.5 Utilization and Throughput	92
7. Conclusions And Recommendations	94
7.1 Summary and Conclusions	94
7.2 Recommendations for Future Research	98
References	100
Appendices	107
Appendix-1 : Experiments and Data	107
Appendix-2 : ANOVA of time in system	128
Appendix-3 : ANOVA of work in process	130
Appendix-4 : ANOVA of stockout	132
Appendix-5 : ANOVA of kanban waiting time	134
Appendix-6 : ANOVA of utilization	136
Appendix-7 : ANOVA of throughput	138
Appendix-8 : Mixed-model sequencing algorithm	140

Appendix-9 : Simulation program listing	142
Abstract	155
Vita	156



## List Of Tables

	Page	
4-1	Classification of JIT related simulation studies	26
4-2	Distributions used in JIT simulation studies	27
4-3	Experimental factors studied in JIT simulation studies	29
4-4	Performance measures used in JIT simulation studies	32
5-1	Subcomponent requirements (BOM)	46
5-2	Master production schedule used	50
5-3	Processing times at various stages for product type 1	51
5-4	Processing times at various stages for product type 2	51
5-5	Processing times at various stages for product type 3	52
5-6	Processing times at various stages for product type 4	52
5-7	Processing times at various stages for product type 5	52
5-8	Quality rework times at various stages	53
5-9	Time between failures of the stages and their repair times	53
5-10	Experimental factors and their levels	62

## List Of Figures

	Page
3-1 Operation of Kanbans	16
5-1 System Being Simulated	42
5-2 Relationship between $X_{jk}$ and $(K*N_j)/Q$	47
6-1 Relationship between time in system and number of kanbans	64
6-2 Relationship between time in system and number of product types	66
6-3 Relationship between time in system and number of stages on feeder line	67
6-4 Relationship between time in system and processing time variability	67
6-5 Interaction between number of kanbans and number of product types	68
6-6 Interaction between number of kanbans and number of stages	69
6-7 Interaction between number of kanbans and level of processing time variability	69
6-8 Interaction between number of product types and level of processing time variability	70
6-9 Relationship between work in process and number of kanbans	72
6-10 Relationship between work in process and number of stages	73
6-11 Relationship between work in process and number of product types	73
6-12 Relationship between work in process and processing time variability	75
6-13 Interaction between number of kanbans and number of stages	76
6-14 Interaction between number of kanbans and number of product types	77
6-15 Interaction between number of stages and number of product types	78

6-16	Relationship between stockout and number of kanbans	80
6-17	Relationship between stockout and number of product types	81
6-18	Interaction between number of kanbans and number of product types	83
6-19	Relationship between kanban waiting time and processing time variability	85
6-20	Relationship between kanban waiting times and number of product types	85
6-21	Relationship between kanban waiting time and number of kanbans	86
6-22	Relationship between kanban waiting time and number of stages	87
6-23	Interaction between number of kanbans and number of stages	89
6-24	Interaction between number of product types and number of stages	90
6-25	Interaction between processing time variability and number of stages	90
6-26	Interaction between number of product types and level of processing time variability	91

# Chapter 1

## Introduction and Statement of the Problem

### 1.1 Just-In-Time (JIT) Systems

In an environment where flexibility and quick adaptability to changes in the market determines a company's survival, more and more companies are turning to a JIT based production system as a solution to remain competitive. JIT has its origins in the Toyota Production System which was developed and promoted by Toyota Motor Corporation. It did not find acceptance in American companies until after the oil crisis in 1973.

The success of JIT can be attributed to its underlying philosophy which goes far beyond just inventory control. It encompasses all aspects of production and seeks to eliminate all sources of waste. There are many sub-components to JIT such as: total quality control, preventive maintenance, multi-functioned workers, automation and inventory control with the aid of *kanbans*. The advantages of JIT cannot be realized until and unless these sub-components have been instituted. The JIT philosophy is explained in detail in Chapter 2.

JIT systems typically employ *kanbans* as a means of inventory control. A JIT system operating under kanban control is commonly called the "pull system" (in contrast to the popular MRP inventory control system which is commonly called the "push system"). This is because of the way in which succeeding stages trigger production in preceding stages. At the time the succeeding stage requires materials for its process, it withdraws materials from its input inventory location. This withdrawal leads to the detachment of a *withdrawal kanban*, which is routed to the output location of the preceding stage. At this output location the withdrawal kanban is exchanged for a *production ordering kanban*, and the output container from this location is transferred to the input location of the stage that sent the withdrawal kanban. In the meantime, the production ordering kanban triggers the

stage to produce an item to replace the container that has been withdrawn. In this manner, each stage on the production line is coupled with the preceding stage as well as the succeeding stage.

## **1.2 Statement of the Problem**

As mentioned in the previous section, a JIT system has various sub-components. However, the kanban controlled inventory system lies at the heart of JIT. It is the aim of this thesis to study the characteristics of these inventory systems.

The success of a kanban controlled JIT system depends on both the design and operational factors associated with the system. Key design factors include the number of product types that are produced on a line, the number of stages on a production line and the number of kanbans that are maintained between stages. In the age of flexible manufacturing systems (FMS), it is very easy to produce more than one product on a single production line (such production lines are referred to in the literature as mixed-model lines). However, this flexibility comes at its own risk. This is basically the risk that for a particular product type the processing time exceeds the *cycle time* associated with the line thereby causing stoppage of work at subsequent stages on the production line. This risk is all the more exacerbated in a kanban controlled JIT system because of the strong linkage that exists between stages. Understanding the behavior of a kanban controlled JIT system manufacturing different product types, which is one of the aims of this thesis, would therefore provide invaluable insights to managers making design decisions.

The number of stages on a production line is the result of the combination of the precedence relationships that exist between jobs and clustering of related jobs. Each such cluster then becomes a stage on the production line. While clustering of related jobs may be an issue by itself (time and motion study, flexible manufacturing, etc.), it begins to occupy an important role in a JIT system as it determines the *length* of the production line

in terms of the number of stages on it. From an intuitive perspective, owing to the strong chaining that exists between stages on a kanban controlled JIT system, one would anticipate a relationship between the length of the production line and its performance. This thesis explores this relationship and will study if the length of the production line (i.e. number of stages on the line) indeed has a bearing on its performance.

Another interesting design factor that is studied is the number of kanbans maintained between stages. As was briefly explained in the previous section, when a *production ordering kanban* is presented at a stage, it signifies the consumption of an item produced by that stage by a subsequent stage, and authorizes that stage to produce an item. When a stage needs to produce an item, it withdraws raw-materials from the preceding stage's output storage area with the help of a *withdrawal kanban*. Withdrawal of raw-materials from the preceding stage will trigger a production ordering kanban to be presented at that stage thereby chaining the production line. To understand the importance of the number of production-ordering and withdrawal kanbans, consider the following situation: Suppose there is a production line with two workstations A and B. Workstation A produces the final part and is therefore considered to be at the *head* of the line; workstation B feeds A and since it is the last station on the line it is considered to be at the *tail* of the line. Let us also suppose that the number of production-ordering and withdrawal kanbans at both stations A and B are equal to one. Under this configuration of the line, imagine that an item is withdrawn from the output storage area of station A. This will cause the production ordering kanban at station A to trigger production at station A to replace the item that was withdrawn. This will in turn cause station A's withdrawal kanban to be used to withdraw an item from station B's output storage area and subsequently cause the production ordering kanban at station B to be used to trigger production at station B to replace the item that was withdrawn by station A. While all this is in progress, if a demand for another item produced by station A occurs, then there is a stockout situation. This stockout could lead to loss of revenue in terms of good-will or much worse: cause stoppage of work on the assembly line if station A feeds into one. Thus it is important to

analyze the behavior of the JIT system by varying the number of kanbans maintained at the stages. These relationships become all the more interesting when one considers the interactions that might exist between the number of kanbans maintained and the number of product types produced on the line, or the number of kanbans maintained and the number of stages on the line, etc.

There are many operational factors associated with a production line, for example, processing time variability, quality fallout, worker training, etc. However, for the purpose of a focused study, the only operational factor that is considered in analyzing the behavior of the kanban controlled system in this thesis is the processing time variability. The underlying premise of JIT systems is that materials will be available at the time the demand is made. However, processing time variability introduces an element of stochasticity which could have an impact on the performance of the JIT system. Quality fallout is assumed to be part of the system but is not considered as an explicit factor.

Often there is no single metric that captures in entirety the conflicting aspects of measuring the performance of a kanban controlled production line. This thesis uses the following set of metrics to measure the performance of a kanban controlled production line:

1. ***Time in system***, defined as the time between the placement of an order and its completion at a station on the production line.
2. ***Inventory***, defined as the time-average of the inventory items at various storage locations on the production line
3. ***Throughput***, defined as the number of items assembled on the assembly line during a shift.
4. ***Stockout***, defined as the number of times an order was placed with no inventory item present to satisfy the request.

5. *Waiting Times of the Kanbans*, defined as the time that the production-ordering and withdrawal kanbans spend waiting for start of service.
6. *Utilization*, defined as the average utilization of all stages on the production line.

### **1.3 Research Objectives**

The overall objective of this research is to characterize the behavior of kanban controlled JIT production lines under the different design and operational factors described in the previous section. This research will aim to determine the relationships between each of the following performance metrics:

1. Time in system
2. Average inventory
3. Average throughput
4. Average stockout
5. Waiting times of the kanbans
6. Utilization of the production line

and the following experimental factors:

1. Number of kanbans maintained at the stages
2. Number of stages on the production line
3. Number of product types assembled on the production line
4. Level of processing time variability



## **1.4 Research Approach**

In order to quantitatively analyze the relationships defined in the previous section, a model that abstracts the key elements of the kanban controlled JIT production system needs to be developed. This model can then be used as a basis for drawing inferences regarding the relationships.

Various alternatives exist for developing the model, ranging from physical models to mathematical models. Physical models are extremely rigid and expensive and very soon lose their usefulness to the modeler. Hence, they are not typical of the kind of models that are usually of interest in Operations Research and System Analysis. Mathematical models on the other hand, represent a system in terms of logical and quantitative relationships and provide the modeler, quite inexpensively, with the capability of observing the system reactions to changes in its underlying relationships. If the system being modeled is simple, then it is possible to construct a closed-form analytical solution which can then be used to answer the questions of interest about the system. However, most real-life systems, of which the kanban controlled JIT system is an instance, are extremely complex and intractable, thereby precluding precise and valid analytical solutions. The literature survey conducted as part of this research (detailed in Chapter 4), indicates that researchers have tackled this problem using analytical approaches, but their usefulness has been limited due to the underlying assumptions. In such situations, where analytical approaches to modeling are ineffective, the model is best studied by means of *computer simulation*. Pritsker (1993) defines *computer simulation* as the process of designing a mathematical-logical model of a real system and experimenting with this model on a computer.

Simulation is an extremely appealing tool to model complex real world situations without building, disturbing or destroying them. It can be an effective technique to assess alternative system designs. It also provides an environment in which better control over experimental factors can be maintained. In addition, simulation results can easily be

explained and understood by management. While all these attributes add to the appeal of simulation, it is not without drawbacks. Simulations are costly to construct, run and validate. Since each run of a simulation model produces only estimates of the performance measures, it is necessary to either make several independent runs of the model or run the simulation for a sufficiently “long” period of time. Simulations also hide critical assumptions. They may be imprecise and the degree of imprecision cannot be measured easily.

The research objectives described in the previous section will be accomplished by developing a simulation model to represent the operation of the kanban controlled JIT production system. The JIT production system used in the simulation model is explained in detail in Chapter 5. Due to the lack of real-life input data related to processing times at workstations, master production schedule, etc., specific distributions have been assumed for the input parameters. It should be pointed out that the model, in spite of being developed based on these distributions, is still a very flexible and adaptable system which can be used to model other JIT systems with very little modification. The model will be fully tested and verified before using it in the experiments to generate statistics for the performance measures of interest. General conclusions regarding the performance of the JIT system under various simulated conditions will then be drawn based on a detailed analysis of the simulation output.

Chapter 2 represents a brief introduction to the JIT philosophy, Chapter 3 represents how kanbans are used in JIT systems for the purpose of inventory control, Chapter 4 contains the literature survey performed as part of the research, Chapter 5 is concerned with the model development, Chapter 6 is concerned with analysis of the output and Chapter 7 is a summary of the conclusions and recommendations.

## Chapter 2

### Just-In-Time Philosophy

The popular Just-In-Time (JIT) concept of production control came into being with the development of the Toyota Production System. It has been so widely accepted by manufacturing firms across the world, that it is no longer associated with Toyota but has been identified as an entity by itself.

The concepts involved in Just-In-Time, are underlined in the Toyota Production System. The Toyota production system was developed and promoted by Toyota Motor Corporation in the aftermath of 1973 oil crisis. The main purpose of the system is to eliminate “waste” through various improvement activities. The other tenets of the system are quantity control, quality assurance and respect for humanity.

Some of the typical “wastes” observed in manufacturing plants are due to excessive production resources. Excessive production resources include *excessive work force, excessive facilities and excessive inventory*. When these elements exist in amounts more than necessary, whether they are people, equipment, materials, or products, they only increase costs and add no value. For instance, having excessive work force leads to superfluous personnel costs, having excessive facilities leads to superfluous depreciation costs and having excessive inventory leads to superfluous financial charges. Moreover, excessive production resources create the secondary waste - *overproduction*. Overproduction is to continue working when essential operations should be stopped. Overproduction causes the third type of waste found in manufacturing plants - *excessive inventory*. Extra inventory creates the need for more manpower, equipment, and floor space to transport and stock the inventory.

Given the existence of excessive resources, overproduction, and excessive inventory over time, demand for the fourth type of waste would develop. This fourth type - *unnecessary capital*

*investment*, includes such things as warehouse construction for stocking excess inventory, hiring extra workers for inventory handling and many other overhead costs.

As mentioned above, “waste” reduction is just one of the objectives of Toyota Production System. But before embarking on waste reduction, three subgoals need to be achieved:

1. *Quantity control*, which enables the system to adapt to daily and monthly fluctuations in demand of quantity and variety
2. *Quality assurance*, which assures that each process will supply only good units to subsequent processes
3. *Respect for humanity*, which must be cultivated while the system utilizes human resources to attain its cost objectives.

It should be emphasized that these three goals cannot exist independently nor can they be achieved independent of each other. All goals are outputs of the same system; with productivity as the ultimate and guiding purpose.

## **2.1 Key Concepts In Toyota Production System**

A continuous flow of production, or adaptation to demand changes in quantities and variety, is created by achieving two key concepts: Just-In-Time and Autonomation. JIT basically means to produce the necessary units in the necessary quantities at the necessary time. Autonomation may be loosely interpreted as autonomous defects control. It supports JIT by never allowing defective units from a preceding process to flow into and disrupt a subsequent process.

## **2.2 Just-In-Time Production**

An example of JIT in the car part assembly process is for the necessary types of subassemblies from the preceding processes to arrive at the product line at the time needed and in the necessary quantities. If JIT is realized in the entire firm, then unnecessary inventories in the factory will be completely eliminated making stores or warehouses unnecessary. In the Toyota System, the people of a certain process go to the preceding process to withdraw the necessary units in the necessary quantities at the necessary time. The preceding process produces only enough units to replace those that have been withdrawn. This method is called the *pull system*.

JIT is managed by the *Kanban System*. In short, the Kanban system is an information system which controls the production processes in every process. A Kanban is a card which is usually placed in a rectangular vinyl envelope. Two kinds of Kanbans are mainly used: the *withdrawal* Kanbans and the *production-ordering* Kanbans. A withdrawal Kanban details the quantity which the subsequent process should withdraw, while a production-ordering Kanban shows the quantity which the preceding process must produce.

Kanbans circulate within Toyota factories, between Toyota and its many cooperative companies, and within the factories of cooperative companies. In this manner, the Kanbans can convey information on withdrawal and production quantities to achieve JIT production. A detailed description of the operation of Kanbans is given in Chapter 3.

The Kanban System is ably supported by the following subsystems, which are integral to the Toyota Production System:

1. Smoothing of Production
2. Reduced Setup Times
3. Standardization of Jobs

4. Improvement Activities
5. Design of Machine Layout

### **2.3 Automation**

As noted previously, the two pillars which support the Toyota Production system are JIT and Automation. To realize JIT, 100 % of defect free units must flow to the subsequent process, and this flow must be rhythmic without interruption. Therefore, quality control must coexist with the JIT operation throughout the Kanban system. Automation means to build in a mechanism to prevent mass-production of defective work in machines or product lines. In Toyota factories, almost all machines are autonomous (an autonomous machine is a machine to which an automatic stopping device is attached, and this device would stop the machine in case of a bad product being produced), thereby preventing mass production of defective parts.

### **2.4 Flexible Work Force and Originality and Ingenuity**

Two concepts key to the success of Toyota Production System include flexible work force and encouraging workers to think creatively. Flexible work force enables varying the number of workers according to demand changes, thereby avoiding excessive work force. By encouraging workers to think creatively, the management is instilling the human respect which is of utmost importance to keep the morale and involvement of the work force at a high level.

### **2.5 Improvement Activities**

The Toyota Production System integrates and attains different goals while pursuing its ultimate goal of cost reduction. Improvement activities are a fundamental element of the Toyota Production System. Each worker has the chance to make suggestions and propose improvements via a small group called a *quality control circle*. Such a suggestion-making process allows for

improvements in quality assurance by preventing recurrence of defective works and machines, and in respect for humanity by allowing each worker to participate in the production process.

In summary, the basic purpose of the Toyota production system is to increase profit by reducing costs through completely eliminating waste such as excessive stocks or work force. The kanban system has been developed as a means of dispatching production and managing JIT. Production smoothing is achieved by reducing production lead times and having small lot sizes. Autonomation (autonomous defects control systems) support the JIT production by ensuring a high level of quality in products. Finally, improvement activities will contribute to the overall process by standardizing operations, remedying certain defects and increasing worker morale.

## **Chapter 3**

### **Kanban system**

The *Kanban System* is an information system that controls the *Just-In-Time* (JIT) production. In this chapter, the various types of kanbans, their usages, and rules are described. The way kanbans connect supporting routines in production lines is also discussed.

#### **3.1 Pulling System For JIT Production**

The first requirement for JIT production is to enable all processes to know the accurate timing and required quantity of demand. In the ordinary production control system, this requirement is met by issuing various production schedules to all of the processes: parts-making processes as well as the final assembly line. These parts processes produce the parts in accordance with their schedules, employing the method of the preceding process supplying the parts to its following process, or, the *push system*. However, this method will make it difficult to promptly adapt to changes caused by demand fluctuations or trouble at some other process. For adapting to these changes under the ordinary system, the company must change each production schedule for each process simultaneously, and this approach makes it difficult to change the schedules frequently. As a result, the company must hold inventory among all processes in order to absorb the demand changes. Thus, such a system often creates an imbalance of stock between processes, which often leads to dead stock, excessive equipment, and surplus workers when model changes take place.

By contrast, in the Toyota system (JIT), the subsequent process will withdraw the parts from the preceding process. Such a mode of operation is termed as *pull system*. Since only the final assembly line can accurately know the necessary timing and quantity of parts required, the final assembly line goes to the preceding process to obtain the necessary parts in the necessary quantity at the necessary time. The preceding process then produces the parts withdrawn by the subsequent process. Further, each part-producing process withdraws parts or materials from



preceding processes further down the line.

Thus it is not required during any time period to issue simultaneous production schedules to all the processes. Instead, only the final assembly line can be informed of its changed production schedule when assembling each vehicle one by one.

### **3.2 What is a Kanban?**

A *Kanban* is a tool to achieve JIT production. It is a card which is usually put in a rectangular vinyl envelope. Two kinds of kanbans are mainly used: a *withdrawal* kanban and a *production-ordering* kanban.

A *withdrawal kanban* specifies the kind and quantity of product which the subsequent process should withdraw from the preceding process, while a *production-ordering* kanban specifies the kind and quantity of product which the preceding process must produce. The *production-ordering* kanban is often called an in-process kanban or simply a production kanban.

Several other kinds of kanbans exist. For making withdrawals from a vendor, a *supplier* kanban is used. The supplier kanban contains instructions which request the subcontracted supplier to deliver the parts. The next type of kanban is called a *signal* kanban. To specify lot production in production processes like diecasting, punchpress, or forging, such kind of kanbans finds common use. An *express* kanban is issued when there is a shortage of a part. Although both the withdrawal kanban and the production-ordering kanban exist for this type of problem, the express kanban is issued only in extraordinary situations and should be collected just after its use. An *emergency* kanban is issued temporarily when some inventory is required to make up for defective units, machine troubles, extra insertions or a spurt in an operation. Various other kanbans which find limited use are described in Monden (1993).

### **3.3 Kanban Rules**

In order to realize the JIT purpose of kanban, the following rules (Monden, 1993) must be followed:

**Rule 1.** The subsequent process should withdraw the necessary products from the preceding process in the necessary quantities at the necessary point in time. Any withdrawal without a kanban should be prohibited. Any withdrawal which is greater than the number in the kanbans should be prohibited. A kanban should always be attached to the physical product.

**Rule 2.** The preceding Process should produce its products in the quantities withdrawn by the subsequent process. Production greater than the number in the kanbans must be prohibited. When various kinds of parts are to be produced in the preceding process, their production should follow the original sequence in which each kind of kanban has been delivered.

**Rule 3.** The number of kanbans should be minimized. Since the number of kanbans expresses the maximum inventory of a part, it should be kept as small as possible. Final authority to change the number of kanbans is delegated to the supervisor of each process.

**Rule 4.** Kanbans should be used to adapt to small fluctuations in demand

**Rule 5.** Defective products must never be conveyed to the subsequent process.

### **3.4 Operation of Kanbans**

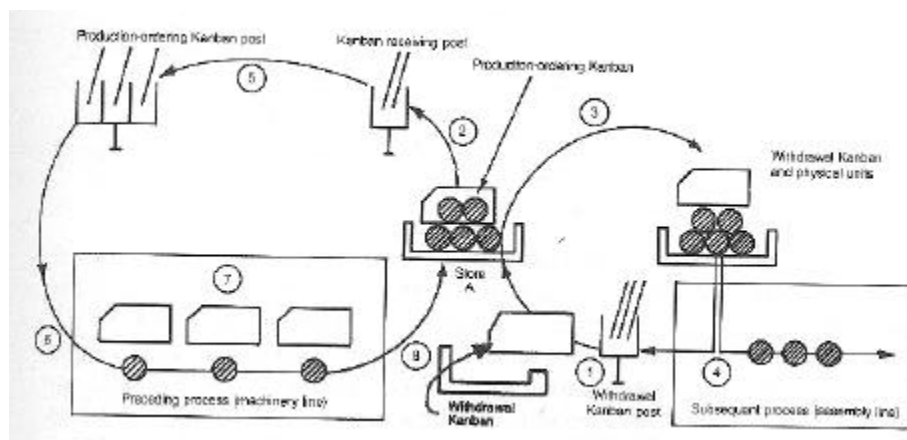
Figure 3-1 shows how the withdrawal kanban and the production-ordering kanban are used. Starting from the subsequent process, the various steps utilizing the kanban are:

**Step 1.** The carrier of the subsequent process goes to the store of the preceding process with the withdrawal kanbans kept in his withdrawal kanban post and the empty pallets on a forklift or jeep. He does this at regular predetermined times.

**Step 2.** When the subsequent process carrier withdraws the parts at store A, he detaches the production-ordering kanbans which were attached to the physical units in the pallets and places these kanbans in the kanban receiving post. He also leaves the empty pallets at the place designated by the preceding process people.

**Step 3.** For each production-ordering kanban he detached, he attaches in its place one of his withdrawal kanbans. When exchanging the two types of kanbans, he carefully compares the withdrawal kanban with the production-ordering kanban for consistency.

**Step 4.** When work begins in the subsequent process, the withdrawal kanban must be put in the withdrawal kanban post.



**Figure 3-1:** Operation of kanbans (courtesy of Monden, 1993)

**Step 5.** In the preceding process, the production-ordering kanban should be collected from the kanban receiving post at a certain point in time or when a certain number of units have been produced and must be placed in the production-ordering kanban post in the same sequence in

which it had been detached at store A.

**Step 6.** Produce the parts according to the ordinal sequence of the production ordering kanbans in the post.

**Step 7.** The physical units and the kanban must move as a pair when processed.

**Step 8.** When the physical units are completed in this process, they and the production-ordering kanban are placed in store A, so that the carrier from the subsequent process can withdraw them at any time.

Such a chain of two kanbans must exist continuously in many of the preceding processes. As a result, every process will receive the necessary kinds of units at a necessary time in the necessary quantities, so that the JIT ideal is realized in every process.

## **Chapter 4**

### **Literature review**

The first article on the JIT implementation in manufacturing appeared in the 1970s (Sugimori et al. 1977). Since then, a lot of research articles have been published in various journals. This review was based on previous literature surveys by Golhar and Stamm (1991), and Huang et al. (1985). The review here is not an exhaustive one, but encompasses most of the major work done in JIT. As the research being proposed deals mainly with kanbans, the first section in this review is devoted to the research that has been conducted in the kanban related area. This is then followed by a review of articles in areas pertaining to JIT.

#### **4.1 Kanban related literature**

As mentioned in the previous chapter, kanban is a method of controlling inventory. Many articles are devoted to the study of kanbans. Typically, most papers deal with the determination of the number of kanbans either through optimization models or through simulation models. Some of these articles that deal with kanbans are listed below according to the category they fall under.

##### **4.1.1 Optimization models**

Kimura and Terada (1981) were among the first to model a kanban system. They describe the system by means of three relations. These relations assume production and order quantities to be multiples of the quantity represented by a kanban card, that there is no time lag between the removal of a card and the start of production, and that production in one period is available to the following workstation in the following period. They did not directly create an optimization model of the behavior of the system described but they used simulation to study the amplification at workstation “n” of fluctuations of demand at the final workstation. Their simulations permitted them to conclude that it is desirable to reduce the number of units represented by a kanban card as

much as possible.

Bitran and Chang (1987) extended the work of Kimura and Terada and offered a mathematical programming model for the kanban system in a multi-stage production setting. Their deterministic model is designed to assist in the choice of the number of kanbans to use at each stage and thus control the level of inventory. The model was solved as an ILP and the solutions were aimed at giving managers in a JIT environment an idea of the operational control. Philipoom, Rees, Taylor and Huang (1990) used an integer programming approach in solving a different problem. They considered the case where it is not possible to reduce all machine setup times to the level required for smooth operation of the kanban method. In the case that they considered, a bottleneck workstation contains more than one machine and produces more than one type of part during a production cycle. Setup times are long with respect to production times. This situation may arise in a smaller shop that uses general purpose machine tools or in a larger plant where cost does not always allow the purchase of dedicated machines. They present evidence showing that in a shop where a limited number of workstations retain long setup times, the use of the unmodified kanban method will, in general, lead to bottlenecks resulting from too frequent and too long setups. The suggested solution is to use *signal* kanbans at the interfaces with the workstations in question.

Bard and Golany (1991) examined the problem of determining the number of kanbans in a multi-product, multistage manufacturing system. They describe the example of a printed circuit board assembly plant where five workstations produce three end products. A workstation is, in principle, able to undertake more than one operation and so to produce more than one in-process item. The objective of the model is to determine the number of kanbans to use at each workstation so as to minimize the total cost, including setup costs, holding costs, and shortage costs, over a known time horizon.

Price, Gravel, Nsakanada and Cantin (1992) describe an adaptation of the model of Bitran and Chang (1987) to the assembly job-shop situation previously modeled as a simulation (Gravel and

Price, 1988, 1992). The model is similar in structure to that of Bitran and Chang (1987), although further constraints and variables are required to take into account the fact that machines are used for multiple tasks. The objective function seeks to minimize the makespan. In a recent paper (1995) they report the numerical results obtained from their optimization model and indicate that there is an upper bound on the number of cards that should be used with a given kanban lot-size.

Li and Co (1991) used dynamic programming to determine the number of kanbans to use at each stage of a production process in order to minimize the inventory holding cost. Their formulation allows the treatment of both serial and tree-structured production processes. They assume that capacity will always allow requirements to be satisfied without backorders. Bielecki and Kumar (1988), have shown that there are ranges of parameter values describing an unreliable manufacturing system for which zero inventory policies are exactly optimal even when there is uncertainty in manufacturing capacity. This result may be surprising since it runs counter to the argument that inventories are buffers against uncertainty and therefore one must maintain a strictly positive inventory as long as there is uncertainty.

#### **4.1.2 Markov models**

Various studies of the kanban system have been done using markov modelling concepts. Deleersnyder et. al. (1989), developed a discrete time Markov model to describe a single card kanban system and to quantify the relationship between cost and service level. They included machine availability and uncertainty of demand as the two sources of randomness in the system. A model for a general N-stage production system was developed and experimental results were obtained with a 3-stage model version of this general N-stage model. They demonstrated that incremental improvement may be achieved by iteratively lowering the inventory level and increasing the machine reliability. The key role of the downstream kanban loop in improving the multistage system performance is also demonstrated. Siha (1994), developed a continuous time markov model for the pull production system. Various allocation patterns of kanban capacity and mean production time over the system stations were studied. Chao et al. (1995) addressed the

prioritization of kanbans in the case of a single station serving multiple downstream stations. They used the markovian decision process (MDP) and dynamic programming concepts and arrived at a solution to this prioritization problem. The solution was found to outperform the first-come-first-served (Berkeley and Kiran 1991) and longest queue rules in experimental tests. Wang and Wang (1990) applied queuing concepts suggested by Hiller and Boling (1967) and then applied a markov process approach to determine the number of kanbans for three production configurations: one station to one station, multiple stations to one station and multiple stations to multiple stations. They also discussed a method for adjusting the number of kanbans in a production system with unreliable machines.

Hodgson and Wang (1991a, b) studied the concept of hybrid push/pull systems in an iron and steel manufacturing unit. They used a markov decision process in their analysis. In their first article, they studied the case with particular emphasis on the above mentioned manufacturing system. In part-2 they extended their work to the general parallel/serial multi-stage production system. Their study showed that it is beneficial to operate all upstream stages of each branch of production line with push policy and to operate all other stages in the pull mode.

Berkeley (1991) showed that the two-card kanban controlled line can be viewed as a generalization of the tandem queue. Many of the kanban systems described in the production literature, and used in practice, are equivalent to a tandem queue. The concepts of tandem queues may be used to obtain the performance measures of two-card systems. A tandem queue approximation can be used to find the minimum number of kanbans required to achieve a desired production level.

Graham (1992), describes a steady-state markovian model for calculating the number of kanbans required to control single-stage processes feeding assembly lines. A markovian model of an alternative just-in-time system, in which the off-line process is triggered by the passage of vehicle bodies past a point prior to the assembly area, is also described.



Chaudhury and Whinston (1990) present a control methodology for flow shops that is decentralized and has low data handling and computational requirements. The methodology is based on stochastic automata methods for modeling learning behavior. It is proposed that such a methodology can be used with kanban type control technique to make flow shop systems more flexible and adaptive in nature. Relationship of the model to computational models such as neural computing is discussed.

#### **4.1.3 Variations of the kanban system**

As an alternative to a kanban method, a periodic pull system (PPS) was developed by Kim (1985) as an operating policy of practicing a pull system. Due to the fact that, in a PPS, the manual information processing time of a kanban method is replaced with an instant on-line computerized processing, the material lead time is much shorter than that of a kanban system. This results in better system performance such as less lead time and faster system response. Kim formulated a PPS mathematically and a solution approach was provided for target stock levels, as well as the analysis of the fluctuations of in-process material flow and on-hand stock levels. Chang and Yih (1994) proposed a modified kanban system called *generic kanban system* which is applicable to dynamic environments. Different production control systems were compared with the generic kanban system and the generic kanban system was observed to perform better than the dedicated kanban.

In a JIT production system with kanbans, as used by the Japanese, the number of kanbans employed at each work center remains relatively constant from month to month, even though demand may change. This occurs because of the unique conditions inherent in the production environment of Japanese firms. However, a number of American firms have considered or attempted implementation of a JIT system with a different production environment, thus necessitating that the number of kanbans at each work center be adjusted periodically. Rees et. al. (1987) developed a procedure for dynamically adjusting the number of kanbans at work centers in an unstable production environment. The methodology utilized forecasted demand and estimates

of container lead time probability density functions.

In a simulation study conducted by Rees, et. al. (1989), they researched the adaptability of the JIT system in an American production operation. A hypothetical, ill-structured production operation was used to compare the effectiveness of kanban with the traditional MRP lot-for-lot approach. Philipoom et. al. (1987) analyzed two aspects of the JIT technique that would confront the production managers implementing it for the first time. The factors that influence the number of kanbans required at work centers and the impact of these factors on the efficiency of a JIT system were demonstrated via a simulation model. In order to identify the factors influencing the number of kanbans, the authors looked at a single work center operating under varied simplifying assumptions. Based on this treatment they identified throughput velocity, coefficient of variation of the processing time at the work center, utilization of the machine and whether or not auto correlated processing times exist.

Hiller and Boling (1967) showed that deliberate unbalancing of production lines leads to improved throughput. Several such unbalancing methods were analyzed by Villeda et. al. (1988). The output rates with unbalanced stations were always superior to the output rates of the perfectly balanced configurations. The extent of improvement over the output rate of balanced systems increased directly with the variability of operation times in final assembly and the subassembly stations, and was also affected by the interstage buffer capacity allowed in the system. Sarkar and Harris (1988), have studied the effect of variation of operation times at different locations on the production line.

#### **4.1.4 Inventory focused studies**

Funk (1989) has evaluated several inventory cost reduction strategies in a JIT manufacturing system, using an existing PCB assembly system as an example. The article compared the effect of reducing the number of work centers in the manufacturing system, the response time between work centers, the number of different parts produced in the system and the lot sizes on the

inventory costs. The analysis found that the number of work centers has the strongest effect followed by response time. However, a reduction in the lot size and the number of different boards if achieved, has a much stronger effect than response time.

Landers and Mohammed (1989), considered the problem of determining the lot sizes on a single-machine flexible work cell or machining center, supplying sub assemblies to a JIT mixed model assembly line. The lot size was determined by considering planned utilization, availability, efficiency and yield (quality) of the work cell. Lee (1987) performed a parametric analysis of the JIT system and concluded that the choice of scheduling rules has a major effect on the system performance. He also observed that the range of product mix does not have a large effect on the system effectiveness. A large variety would cause a marginally better process utilization and gives a lower mean job tardiness.

Sometimes it is hard to reduce set-up times at work centers. So, in such cases, a special type of kanban, referred to as a “signal kanban” is used to produce larger than normal lots at such work centers. Philipoom et. al. (1990) have reasoned as to why the traditional EOQ lot-sizing is inapplicable for signal kanbans. They developed integer mathematical programming for determining the optimal lot-sizes. Two versions of IP models were developed and solved. One of these models minimizes inventory at a work center while the other version minimizes inventory and setup costs. The impact of the results obtained from the models was determined using simulation. The simulation results while not offering any universal guidelines or conclusions, did offer several implications for the manager. In general, these implications are that it may be prudent to consider inventory and setup costs rather than simply seeking to reduce inventory to its minimum level in a JIT system; and under certain conditions a signal kanban system may be more cost effective than a feasible standard kanban system.

In a related paper, although not concerning JIT per se, Porteus (1985) examined the tradeoff between the investment costs needed to reduce the setup cost and the operating costs identified in the EOQ model. An extension of the EOQ model in which the setup cost is viewed as a decision

variable, rather than as a parameter, was developed. Miyazaki, Ohta and Nishiyama (1988), derived two formulae to calculate the average inventory yielded by fixed interval withdrawal kanban and supplier kanban. They used these formulae to determine the number of kanbans needed. They also proposed an algorithm to obtain the optimal order interval that minimizes the total operation cost.

A method of estimating the amount of safety stock needed at each station to achieve a certain predetermined level of performance (percentage of demand backlogged) was developed by So and Pinault (1988). The analysis was based on systems with only one station. The method was then applied to the general model by decomposing the production line into separate stations and analyzing each station separately. Experimental studies show that their method of buffer size estimation matched the results obtained by simulating the same system. South (1986) developed a minimum lot-size formula for a JIT operating system. Spence and Porteus (1987) modeled the increased effective capacity that results from reduced setup times. They used the multi product capacitated EOQ model. However a major limitation of the model is that it ignores uncertainty.

A key question for deciding upon inventory levels in manufacturing systems is the holding cost. Several authors argue that the holding costs are in general overstated. The article by Wacker (1986) examines the logic behind these arguments and describes the effect of holding costs on JIT manufacturing system performance.

#### **4.1.5 Simulation studies in JIT production**

Simulation has been widely used as a tool to identify and study internal and external factors that affect the success of JIT implementation; to determine the number of kanbans required at each work station (Changchit and Terrel (1988)); and to investigate the effect of demand, processing time variations and other factors.

Simulation has been used in JIT studies for two main reasons. First, it was used to evaluate the

relative performance of JIT production with other types of production systems such as MRP (Krajewski et al. 1987), order point systems and push-type systems (Kimura and Terada 1981, Sarker and Fitzimmons 1989). The second objective was to identify factors detrimental to the success of JIT implementation (Gupta and Gupta 1989, Huang et al 1983, Krajewski et al. 1987, Philipoom et al. 1989, Sarker and Harris 1988).

**4.1.5.1 Model Configuration**

Configuration of a conceptual model used may affect the simulation accuracy and efficiency. The model structures used in JIT simulations can be categorized into two major groups: multi-line, multi-stage; and single-line, multi-stage models. The following table classifies papers according to the category they belong to:

Table 4-1 : Classification of JIT related simulation studies

<b>Model Configuration</b>	<b>One kanban</b>	<b>Two kanbans</b>	<b>References</b>
<b>Multi-line multi-stage</b>	X	X	Gupta and Gupta (1989)
		X	Huang et. al. (1983)
		X	Sarker and Harris(1988)
		X	Villeda et. al. (1988)
<b>Single-line, Multi-stage</b>	X	X	Changchit and Terrell (1988)
		X	Kimura and Terada(1981)
		X	Sarker and Fitzimmons (1989)
		X	Schroer et. al. (1984)
X	X	Schroer et. al. (1985)	

Most of the models are relatively small in scale. The largest model is due to Sarker and Fitzimmons (1989), where a nine stage model is used. Only Schroer et. al. (1984) have evaluated the relative performance of systems that use a single kanban over those systems that use dual-card kanbans.

The assumptions of the simulation model affect the realism of the results. They also may affect the way results are interpreted. The review shows that most studies(Changchit and Terrel 1988, Gupta and Gupta 1989, Huang et. al. 1983, Sarker and Fitzimmons 1989, Sarker and Harris 1988, Schroer et. al. 1984, 1985, Villeda et. al. 1988) assume a perfect production process; that is, no scrap or defect will occur. Some studies also assume that machines never break down (Sarker and Harris 1988, Villeda et. al. 1988) and that a fixed daily production is available(Schroer et. al. 1984, 1985).

**4.1.5.2 Simulation language used**

SLAM and SIMAN were the most widely used simulation languages. Gupta and Gupta (1989) used a continuous type language, DYNAMO, in their studies. Krajewski et al. (1987) developed the MASS program in-house using the FORTRAN programming language.

**4.1.5.3 Distribution and Random Variables used in the simulation**

The following table summarizes the processes and corresponding distributions used in the studies:

Table 4-2 : Distributions used in JIT simulation studies

<b>Process Variables</b>	<b>References</b>
<p><b>Processing (service) time</b></p> <p><b>Exponential distribution</b></p>	<p>Changchit and Terrell(1988), Huang et. al. (1983)</p>

<b>Process Variables</b>	<b>References</b>
<b>Normal distribution</b>	Changchit and Terrell(1988), Huang et. al. (1983), Gupta and Gupta (1989), Huang et. al. (1983), Sarker and Fitzimmons (1989), Villeda et. al. (1988)
<b>Constant</b>	Huang et. al. (1983)
<b>Random</b>	Krajewski et. al. (1987)
<b>Setup time</b>	
<b>Uniform distribution</b>	Sarker and Fitzimmons (1989)
<b>Lead Time</b>	
<b>Normal distribution</b>	Krajewski et. al. (1987)
<b>Time between arrival</b>	
<b>Constant</b>	Schroer et. al. (1985)
<b>Time between machine repair</b>	
<b>Exponential distribution</b>	Sarker and Fitzimmons (1989)
<b>Time between machine failure</b>	
<b>Normal distribution</b>	Krajewski et. al. (1987), Sarker and Fitzimmons (1989)

#### ***4.1.5.4 Design of Experiment***

Some of the factors that have been used in various studies include the type of inventory policy, variables related to the process, product structure, influence of the customer, supplier/vendor

influence and various buffer mechanisms. The following table is a listing of the factors studied and the corresponding reference.

Table 4-3 : Experimental factors studied in JIT simulation studies

<b>Factors</b>	<b>Related References</b>
<b>Inventory Policy</b>	
<b>Lot sizing at end item level</b>	Kimura and Terada (1981), Krajewski et. al. (1987)
<b>Lot sizing at component level</b>	Krajewski et. al. (1987)
<b>Inventory record inaccuracy</b>	Krajewski et. al. (1987)
<b>Process</b>	
<b>Setup time</b>	Krajewski et. al.(1987), Villeda et. al. (1988)
<b>Stock-out level</b>	Gupta and Gupta (1989)
<b>Scrap rate</b>	Krajewski et. al. (1987)
<b>Worker flexibility</b>	Changchit and Terrell(1988), Krajewski et. al. (1987), Villeda et. al. (1988)
<b>Equipment failure(breakdown)</b>	Gupta and Gupta(1989), Krajewski et. al.(1987), Sarker and Fitzimmons (1989)
<b>Line imbalance</b>	Gupta and Gupta(1989), Huang et. al.(1983), Krajewski et. al. (1987), Sarker and Fitzimmons (1989), Schroer et. al. (1984, 1985), Villeda et. al. (1988)



<b>Factors</b>	<b>Related References</b>
<b>Lead time</b>	Gupta and Gupta(1989), Huang et. al. (1983) Krajewski et. al. (1987), Villeda et. al. (1988) Sarker and Fitzimmons (1989)
<b>Processing time</b>	Changchit and Terrell (1988), Gupta and Gupta (1989), Huang et. al. (1983), Kragewski et. al. (1987), Sarker and Fitzimmons (1989) Sarker and Harris (1988), Villeda et. al. (1988)
<b>Bottleneck</b>	Changchit and Terrell(1988), Huang et. al. (1983),
<b>Time between arrival</b>	Schroer et. al. (1984,1985)
<b>Final assembly variability</b>	Villeda et. al. (1988)
<b>C. Product structure</b>	
<b>Shape</b>	Krajewski et. al. (1987)
<b>BOM Level</b>	Krajewski et. al. (1987)
<b>D. Customer influence</b>	
<b>Forecast error for products</b>	Krajewski et. al. (1987)
<b>Proportion of specials</b>	Krajewski et. al. (1987)
<b>Demand rates</b>	Gupta and Gupta (1989), Huang et. al. (1983) Kimura and Terada (1981), Sarker and Fitzimmons (1989)

<b>Factors</b>	<b>Related References</b>
<b>E. Vendor influence</b>	
<b>Average lead time</b>	Krajewski et. al. (1987)
<b>Lead time variability</b>	Krajewski et. al. (1987), Schroer (1984, 1985)
<b>Vendor quality</b>	Krajewski et. al. (1987)
<b>Supply rates</b>	Gupta and Gupta (1989), Schroer et. al. (1984, 85)
<b>F. Buffer mechanisms</b>	
<b>Buffer size</b>	Gupta and Gupta (1989), Kimura and Terada (1981), Schroer et. al. (1984,85)
<b>Buffer stock and lead time</b>	Krajewski et. al. (1987), Villeda et. al. (1988)
<b>Capacity slack</b>	Gupta and Gupta (1989), Krajewski et. al. (1987)

#### ***4.1.5.5 Measures of Performance***

The performance of JIT systems can be evaluated from several directions. The measures may be classified into three categories: overall, inventory related and due-date related measures. As shown in the following table, three criteria - utilization of facility, output(production rate), and work-in-process (WIP) - have been most frequently used. Because different measures have been used in different studies with different model assumptions and experimental factors, the comparison of simulation results between studies becomes very difficult.

Table 4-4 : Performance measures used in JIT simulation studies

<b>Measures</b>	<b>Related References</b>
<p><b>A. Overall measures</b></p> <p><b>Total cost</b></p> <p><b>Waiting time</b></p> <p><b>Idle time</b></p> <p><b>Production rate</b></p> <p><b>Facility Utilization</b></p> <p><b>Throughput</b></p> <p><b>Length of queue</b></p> <p><b>Overtime</b></p> <p><b>Service level</b></p>	<p>Huang et. al. (1983)</p> <p>Sarker and Harris (1988), Schroer et. al. (1984,85)</p> <p>Villeda et. al. (1988)</p> <p>Gupta and Gupta (1989)</p> <p>Changchit and Terrell (1988), Sarker and Fitzimmons (1989), Sarker and Harris (1988), Schroer et. al. (1984), Villeda et. al. (1988)</p> <p>Changchit and Terrell (1988), Sarker and Fitzimmons (1989), Sarker and Harris (1988), Schroer et. al. (1984), Villeda et. al. (1988), Gupta and Gupta (1989)</p> <p>Schroer et. al. (1984,85)</p> <p>Sarker and Fitzimmons ('89), Sarker and Harris ('88)</p> <p>Changchit and Terrell (1988), Huang et. al. (1983)</p> <p>Krajewski et. al. (1987)</p>
<p><b>B. Inventory Related Measures</b></p> <p><b>Raw material inventory</b></p> <p><b>Work-in-process (WIP)</b></p>	<p>Huang et. al. (1983)</p> <p>Gupta and Gupta (1989), Huang et. al. (1983)</p>

<b>Measures</b>	<b>Related References</b>
<b>Finished goods inventory</b>	Sarker and Fitzimmons (1989), Villeda et. al. (1988) Huang et. al. (1983)
<b>Total inventory</b>	Krajewski et. al. (1987)
<b>C. Due date related measures</b>	
<b>Shortage of final product</b>	Gupta and Gupta (1989)
<b>Past due demand</b>	Krajewski et. al. (1987)

#### **4.2 Global productivity comparison**

Wheelwright (1981) and Abernathy et. al. (1981) attribute the success of Japanese manufacturing firms to the willingness of managers to pay close attention to details. For example, the production related issues (workers, quality control, product assurance, production planning, and materials control) are treated as strategic factors by Japanese managers, while managers in Western countries consider these issues as operational factors. According to these researchers, the strategic approach has caused Japanese management to take a long-term view of business and this has been instrumental in fostering the evaluation of the JIT philosophy. Productivity advantages of Japanese industries suggest that JIT implementation leads to profitability. The productivity improvements in the Japanese automotive industry can be attributed to operational improvements and managerial systems. Burnham (1985) observed that in order to be competitive, industries must focus on workforce involvement in decision-making and management commitment to viewing production related issues as strategic related issues instead of operational.

Cooper (1984) argues that culture has been critical to the successful JIT implementation by Japanese firms. He classifies Japan's cultural issues into two broad categories: worker related and management related. Japanese workers are loyal, flexible, well educated, co-operative and willing

to work long hours. Key elements of the Japanese management approach are: life-time employment, consensus and bottom-up approach to decision making, respect for workers, seniority based pay scale and a paternalistic approach towards workers. This distinction between worker and managerial cultural issues is helpful because Japanese management approaches have been imported successfully by other cultures. For example, Japanese companies with subsidiaries in the USA are providing the same working environment as in Japan and getting US workers involved in implementing the JIT philosophy. Thus for a successful JIT implementation in the West, a strategic view of the production related issues and employee oriented Japanese management approach seem critical.

### **4.3 Comparison of JIT with other production control methods**

A number of papers have been devoted to a qualitative comparison of production control concepts. Gelders and Wassenhove (1984) compared Materials Requirement Planning (MRP), kanban, and Optimized Production Technique (OPT), while focusing on their properties. Aggarwal (1985) confronted an assessment of the same concepts with the expected impact of flexible manufacturing systems (FMS). Grunwald et. al. (1989) studied the influence of demand characteristics and characteristics of product structure on the various production control techniques. Their analysis suggests that when uncertainty and complexity are small, kanban control is the best control policy method. However under growing uncertainty, production on order will become difficult. For relatively small non-stationary uncertainty, MRP can be used. For growing non-stationary uncertainty, the disadvantage of MRPs is that safety stock levels should be adopted more frequently. If complexity increases, MRP will increasingly fail to handle capacities. In such a situation a control concept like OPT is required.

Using data collected from case studies, Plenert, Gerhard and Bert (1986) conclude that JIT and OPT are more productive than MRP, and OPT system is more complete than the JIT system. In their opinion, the OPT system develops a detailed operating philosophy, not just an operating procedure and it includes many of the features of the JIT system and additional benefits as well.

According to some simulation and mathematical studies on inventory system comparisons, the manufacturing environment is most crucial in reducing inventory. Krajewski et. al. (1987), guided by a panel of production and inventory managers from diverse plant environments, identified manufacturing environmental factors relevant to United States that are necessary for better performance. They then assessed the robustness of the kanban system in various environments. Their first study identified inventory, process, buffer mechanism and customer influence, to be the significant influencing factors in the manufacturing environment. When the kanban system was tested in various environments, it led to impressive performance, in the settings that had already been identified to be conducive for improved manufacturing. To determine if kanban, intrinsically had qualities which lead to the performance or if the environment resulted in the performance, the authors compared kanban system with Reorder Point (ROP). Surprisingly, kanban and ROP performed at the same level and the authors concluded that rather than kanban being intrinsically better, the environment stipulates whether it would result in improved manufacturing performance.

Another critical factor in successful implementation of an inventory system is human involvement. Since workers are responsible for making the kanban system work, it is relatively simple and easy to implement (Rice and Yoshikawa 1982, Aggarwal 1985). MRP on the other hand has more human involvement problems because employees are more excluded from decision-making and fail to see how their work contributes to reduction in inventory. OPT requires managers to make procedural and work method changes prior to its implementation. These changes increase employee involvement. As a result, OPT has fewer employee problems in controlling inventory than MRP.

The articles cited above focus on contrasting kanban, MRP and OPT. However, some studies argue that kanban and MRP systems are complementary (Karmakar 1989). According to these researchers, the strength of MRP is in long-term planning and scheduling, while kanban is better at daily operations, controlling production by reducing inventory, and providing visible controls. Thus, integration of MRP and kanban would allow a manufacturer to improve productivity and

customer service level.

#### **4.4 Literature related to JIT practices in industry**

Success of JIT philosophy depends on the implementation of four basic tenets:

1. Elimination of waste
2. Employee involvement in decision-making
3. Supplier participation
4. Total quality control

There are numerous articles in literature related to the practice of JIT in the industry. Ebrahimpour and Schonberger (1984) have focused on Japanese management techniques relevant to JIT production with total quality control (TQC). They have briefly explained JIT and TQC conceptually. An exposition is made which argues that by technology transfer of JIT techniques many pit-falls in the manufacturing environment prevalent in developing countries may be overcome.

Finch and Cox (1986) looked at the potential of JIT for small manufacturers. They argue that repetitive environment is not a requirement for a successful JIT approach. Implementation of a JIT system in a small bottling factory is described to support their argument. In their opinion, most small factories are focused factories by account of producing few products. Reduction in set-up times, indispensable for successful JIT implementation, may be achieved by small manufacturers as well. The other foundations of JIT such as group technology, preventive maintenance and cross training can be achieved by small manufacturers as well as by large manufacturers. However there are a couple of difficulties associated with JIT implementation in small firms. They cannot order JIT delivery of raw materials and components from vendors. Also, they do not have a uniform demand rate/uniform work-load. A realistic approach for the small manufacturer is to strive to attain those JIT requirements that can be easily implemented and not

to move towards a total JIT system at one shot.

Crawford et. al. (1988), conducted a survey to identify implementation and operating problems experienced by implementers of JIT. According to the survey, the major reasons cited for JIT implementation were to regain competitive position, to increase profit, to improve quality, and to lower cost. Reduced inventories, reduced manufacturing costs, reduced lead times, and improved product quality were identified to be some of the benefits of implementing JIT. Difficulties in implementing fell into two broad categories: people problems and operating problems. People problems included cultural resistance to change, lack of top management support and lack of organizational communication. Operational problems included lack of vendor support, inability to meet schedule, machinery breakdown, poor forecasting, and data inaccuracy.

Another survey was conducted by Gilberth (1990). He conducted a broad based survey of U.S. firms to determine their state in the implementation of JIT philosophy. Lessons learned from this survey were that companies should start with in-house programs of lot-size reduction, quick set-up and total quality management. External programs requesting suppliers to speed delivery and reduce batch sizes should follow and not lead the implementation. Several important elements of JIT such as buffer stock removal, group technology, overlapped scheduling, consistency in the master production schedule, standardization of component items, and plant-wide involvement in work improvement projects should come later in the JIT developmental effort. Companies that are implementing JIT must realize that JIT is not just an inventory reduction technique. It is a system whose underlying philosophy is continuous production process improvement. It is not possible for most companies to jump directly into a JIT system. Miltenberg and Wigngaard (1991) propose a three phase implementation of JIT systems which will not disrupt the existing system.

In a study by Ahmed et. al. (1991), they explore whether there are differences among the companies at various states of JIT implementation based on factors which were considered relevant in the literature. The factors were categorized as internal and external. Internal factors relate to questions involving product, equipment and process, work-force and top management



commitment. External factors deal with supplier and customer issues. Responding firms were categorized into five groups based on their states of JIT implementation ranging from totally non-JIT to partial and full implementation. The findings of this study suggest that some of the issues such as supplier proximity and lead-time of the supplier, which have been traditionally considered to be important, are not major impediments to JIT implementation. However, there are some issues such as top management commitment which may facilitate or inhibit the move towards JIT implementation.

Crawford and Cox (1990) conducted a study to understand how performance is evaluated in JIT operations. JIT requires a continuous effort to improve the manufacturing process. Thus the performance measurements used in this environment must be constructive rather than punitive. In this article, various issues regarding the setting up of performance measurements in a JIT environment are addressed.

Richmond and Blackstone (1988), have looked into the plastics processing industry to determine how JIT concepts are being implemented. Elimination of waste is taking place through the implementation of quality control at the source. Use of Statistical Process Control (SPC) to maintain quality at desired levels, setup time reduction through advanced technology, JIT purchasing practices are being implemented with the aim of adopting JIT. Parnaby (1988), has emphasized that the introduction of JIT manufacturing systems requires the simultaneous and integrated development of many elements in a business. He has elucidated the concept of a systems approach by describing JIT implementation in Lucas Industries. Wildemann (1988), has described the efforts towards JIT implementation in German manufacturing units.

Flapper et. al. (1991) proposed a method by which JIT can be embedded into the currently existing MRP. By following a three step framework as proposed by these authors, the number of changes that need to be made to the MRP database are relatively few. Making these changes to MRP will simply allow JIT to operate within MRP. Lockmay and Cox (1991) have studied the potential of V-A-T analysis in the implementation of JIT. V-A-T analysis is a procedure which

categorizes manufacturing operations based upon the nature of product flow via the structure of the routings and bill of materials. The letters V, A, and T stand for the shape of the dominant product routings and bill of material structures within a manufacturing facility.

Just-in-time production systems operate effectively if resources are correctly sized in terms of performance and availability. In JIT production lines in particular, performance can be considerably affected by machine reliability and the attendant maintenance policies. In a paper by Albino (1992), a single-product multi-stage manufacturing line with resource failures and controlled by a two card kanban system was modeled using discrete-event simulation. Several performance measures were evaluated to determine optimum operating policies given resource failures. In addition, different maintenance policies were implemented in order to better understand their impact on overall system performance. Several case examples were developed to further provide insight into the problem. Abdalnour et. al. (1995) have also identified the effect of different maintenance policies on the JIT production system.

Sequencing of products on an assembly line in a JIT production system is another area which has an impact on the successful implementation of the system. This is because it is necessary that the parts usage be as constant as possible. Many researchers have developed sequencing rules for sequencing mixed models on the assembly lines. Sumichrast et. al. (1992), compare several procedures for sequencing products on a mixed-model assembly line in a just-in-time production system using simulation analysis. The procedures include two “goal chasing” heuristics developed at Toyota that focus on constant component usage, an algorithm developed by Miltenberg (1989) designed to achieve uniform production rates for each model, a time spread method developed by the authors that smooths the work load at each assembly line station, and batch sequencing procedure which is frequently used in practice. These two sequencing procedures are evaluated according to four measures of assembly line inefficiency: work not completed, worker idleness, worker station time and a measure of variability in uniform component usage.

## **4.5 JIT purchasing**

JIT purchasing has been an important factor in improved product quality and productivity. Ansari and Modarress (1988) studied four US manufacturers who are leaders in using JIT purchasing. The results of their study indicate that quality inspection at the source of supply, small lot-size with frequent deliveries, a drastically reduced number of suppliers, and evaluating and selecting suppliers based on quality and mutual relationships are major activities of JIT purchasing which have significantly contributed to the improvement of product quality and productivity.

Chapman (1989) reports on an empirical field study exploring the relative importance of several JIT based independent variables on the total level of supplier linked inventory in a JIT environment. The results from the study suggest that the customer gains substantial benefits in inventory reduction by focusing on lot size and lead time. The study also supports the contention that an integral part of any JIT implementation is the establishment of close communication with suppliers and a complete education program that informs suppliers what to expect and how to respond to JIT. A just-in-time (JIT) buyer demands frequent deliveries of small lots of certain products. However, there is a perception among the suppliers that participation in a JIT delivery system is economically disadvantageous for them. To test this conjecture, a generalized inventory model was developed for a supplier who has to meet a deterministic demand at fixed intervals. The generalized total cost model developed was found to be a piecewise convex function. A simple algorithm was developed to compute the optimal batch size. When production uptime and cycle time are each equal to an integer multiple of the shipment interval, a perfect matching of shipment size occurs, and for such a situation, the generalized model specializes to more traditional inventory models. The solution approach for such models becomes much easier. It was found that, under certain conditions, the total cost decreases linearly with reduced shipment size and the suppliers benefit from this. Economic impact of ordering and setup cost reduction were also investigated.

## Chapter 5

### Model Development

#### 5.1 JIT System Description

The overall aim of this thesis is to study the behavior of a kanban controlled JIT production line by varying the design and operational parameters associated with the production line and analyzing the various performance metrics through a simulation model. Since a simulation model is a computer representation of the system that is being modeled, it is important to define the system being modeled in explicit terms in order to establish the boundaries and the level of detail that needs to be included in the simulation.

The system being modeled is best explained by means of the graphical representation in Figure 5-1. The figure shows a production system with an assembly line manufacturing various products. The assembly line contains various workstations (Workstation-1, Workstation-2, etc.), and each of the workstations assembles a sub-component of the final product. The assembly line may be thought of as a conveyer system in which all workstations preceding Workstation-1 (if any) maybe thought of as the *upstream* side of the assembly line, and all workstations subsequent to Workstation-1(if any) maybe thought of as the *downstream* side of the assembly line. The first workstation on the upstream side of the assembly line is considered as the *head* of the assembly line and the last workstation on the downstream side of the assembly line is considered as the *tail* of the assembly line.

Assembly lines in general could be part of a repetitive manufacturing facility wherein production is scheduled according to a master production schedule (MPS), or it may be the case that the assembly line is directly exposed to the market demand. In the latter case, orders for production of an item are placed at the same time that the demand for the item occurs. Typical examples for this kind of assembly lines are the fast food counters at

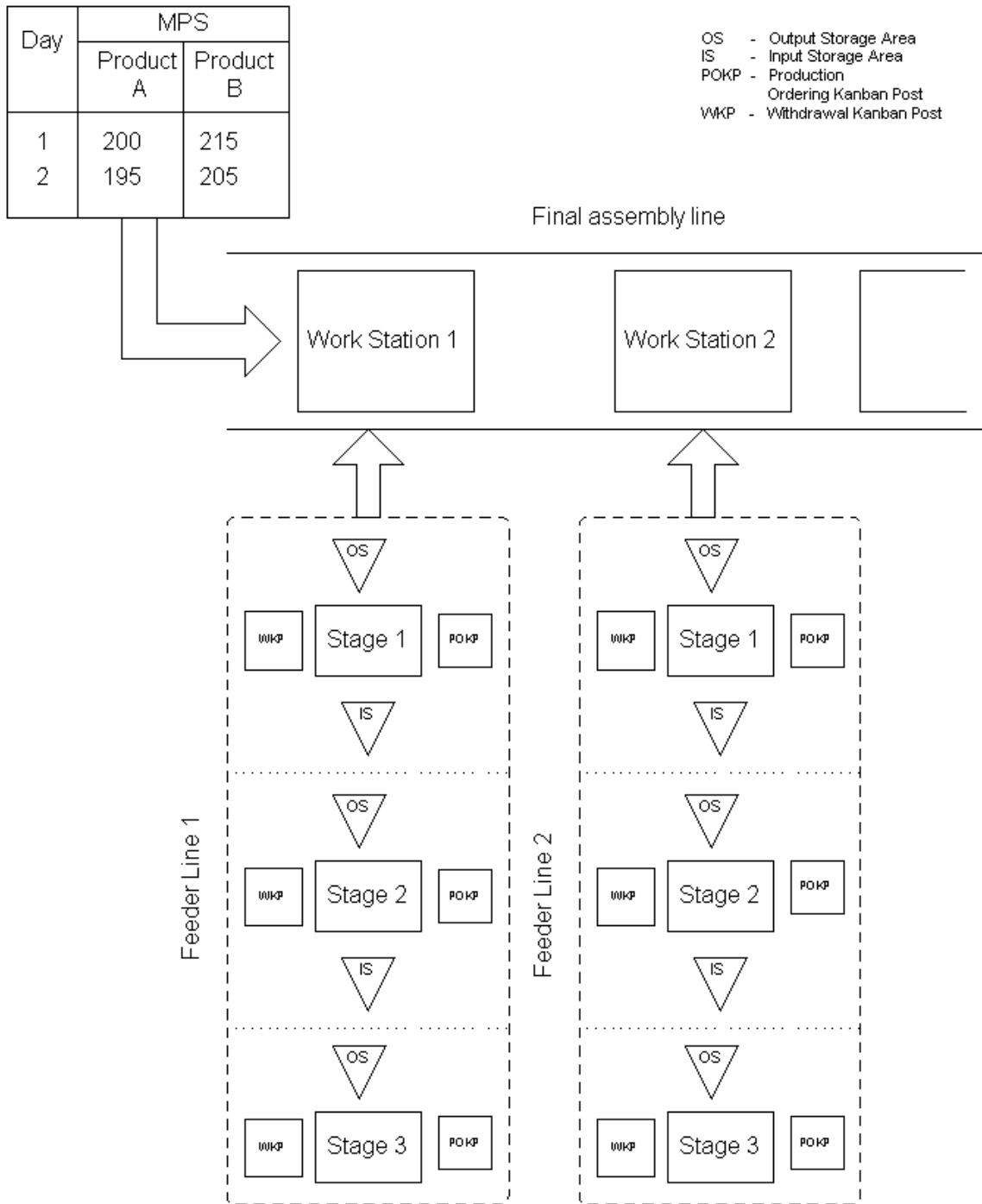


Figure 5-1: System used in simulation

various restaurants. When a customer places an order at the counter, that order is communicated to the head of the assembly line and production begins to complete that order. On the other hand, in repetitive manufacturing facilities, a master production schedule (MPS) specifies the number of products to be produced during a specified period of time (e.g. duration of a shift). The specified period of time divided by the number of products to be produced during that period of time, yields a quantity commonly referred to as the *cycle time*, and it signifies the time between placement of orders at the head of the assembly line to meet the MPS specifications. For e.g., if the shift length is 480 minutes and if 60 products are required to be produced during the shift, the cycle time is 8 minutes and this then becomes the time between placement of orders at the head of the assembly line. Examples of such kinds of assembly lines are numerous and the most famous is the automobile assembly line, which needless to say, has had a great impact on our day-to-day life. Due to the proliferation of such assembly lines and their relevance to the industry, this thesis considers only this kind of assembly lines. In addition, the thesis considers the mixed-model kind of assembly lines due to the fact that most assembly lines these days are of the mixed-model kind, i.e., have the capability of assembling different but related products. The single-model assembly line would be a special case of the assembly line considered in this research.

A separate feeder line feeds each of the workstations on the assembly line with the corresponding sub-component that goes into the assembly of the final product on the assembly line. The feeder line might itself be an assembly line whose final product is the sub-component which is fed to the workstation on the assembly line assembling the final product. In order to distinguish between the references to the assembly line on feeder lines and the assembly line producing the final product, the latter is henceforth referred to as the *final assembly line*. As another point of distinction between feeder lines and final assembly lines, workstations that may exist on the feeder lines are referred to as *stages*. An example of where this configuration of a final assembly line and feeder line combination might exist, is the following: Consider a final assembly line manufacturing automobiles. On the

final assembly line there might be a station that assembles the engine-system onto the automobile. The engine-system, while being a sub-component of the final product, may be the final product of another assembly line.

The feeder line used in this research is a JIT production system and uses kanbans for inventory control. The study of such feeder lines under varying design and operational conditions is the underlying objective of this thesis. As Figure 5-1 indicates, the feeder line has various components and characteristics. Just like the final assembly line, the feeder line is also mixed-model in nature and it outputs a different kind of product depending on the kind of product requested by the final assembly line. The feeder line has various stages which are the sites where processing occurs. These stages may be envisioned as human operators, robots or some kind of a flexible machine. Each stage has an output storage area and an input storage area. The output storage area is where the stage's production is stored while the input storage area is the place where the inputs to the stage (maybe output from a preceding stage) are stored. Each stage also has a production ordering kanban post and a withdrawal kanban post.

The feeder line is tightly integrated with the final assembly line. When a demand occurs at a workstation on the final assembly line, it retrieves an item from the output storage area of the tail of the feeder line. At the time of retrieving the item, it detaches a production ordering kanban from the item and places the kanban in the production ordering kanban post associated with that stage. The kanbans are placed in the post in an ordinal sequence. At some specified time, the operator of that stage on the feeder line will poll the production ordering kanban post and if there are any kanbans in it, the operator will start production corresponding to the kanban first in the ordinal sequence. At start of production, the operator will use an input item from the stage's input storage area and will detach a withdrawal kanban and place it in the stage's withdrawal kanban post. At some specified time, the withdrawal kanban post will be polled and if there are any kanbans in it, then the operator will go to the preceding stage's output storage area and withdraw an

item corresponding to the kanban. At the time of withdrawal, a production ordering kanban is detached and placed in the preceding stage's production ordering kanban post. Preceding stages on the feeder line will then follow a similar approach to trigger production and withdrawal. To make the feeder line more realistic, the stages will be subject to breakdowns and the quality of products produced by the stages will have a certain percentage of defectives. While it is the goal of this research to draw general conclusions about such feeder lines which might be of use to a manager designing such a system, for the sake of simplicity, certain aspects of a real system such as: scheduled maintenance of stages, worker fatigue, sick time, etc., have been excluded.

## **5.2 Sequencing Products on a Mixed-Model Assembly Line**

Determining the sequence of products to be assembled on a mixed-model assembly line is an important issue in JIT production systems. Since different models have different sub-component requirements at each of the workstations along the final assembly line, it is important to minimize this variation by trying to maintain a constant rate of consumption of sub-components at each of the workstations. This is done in order to reduce the variation in production quantities ordered from the feeder lines, to reduce work-in-process on the feeder lines and to maintain a constant rate of utilization. To understand the importance of the sequence of products to be assembled, consider the following case: Suppose that a final assembly line produces related product types A, B and C. Suppose also, that the final assembly line has three workstations Workstation-1, Workstation-2 and Workstation-3 each of which is fed by an associated feeder line and that these feeder lines produce sub-components in lot sizes of 3, 4 and 4 respectively. In addition, suppose that the difference in the product types lies in the number of sub-components required, also known as *bill of materials* (BOM), at the three workstations. These requirements are shown in the Table 5-1.



Table 5-1: Subcomponent requirements (BOM)

Products	WS1	WS2	WS3
A	2	1	4
B	3	4	2
C	1	2	1

Consider that during a shift, the number of A, B and C required is 10, 8 and 7 respectively. If a sequence in which all products of type A are followed by all products of type B followed by all products of type C is used, then during the assembly of all A at Workstation-1, for every order placed on the feeder line, 1 sub-component is going to remain unused thereby increasing the work-in-process inventory. When it comes time for all the C product types, the order frequency to the feeder line at Workstation-1 is reduced to one for every 3 products assembled, thereby reducing the utilization of the feeder line. Similar arguments apply at the remaining workstations also. Thus it is important that an appropriate assembly sequence be determined.

This research utilizes the “goal chasing method” as described in Monden(1993), and which is also the approach followed by the original Toyota Production System for determining an appropriate sequence of mixed models for assembly. It must be pointed out that this method produces only a heuristic solution which may or may not be an optimal sequence. To understand this sequencing method, the following notation and values need to be defined:

$$Q = \text{Total production quantity of all products } A_i \text{ (} i=1, \dots, \mathbf{a} \text{)}$$

$$= \sum_{i=1}^{\mathbf{a}} Q_i, \text{ (} Q_i = \text{production quantity of each product } A_i \text{)}$$

$$N_j = \text{Total necessary quantity of part } a_j \text{ to be consumed for producing all products } A_i \text{ (} i=1, \dots, \mathbf{a}; j=1, \dots, \mathbf{b} \text{)}$$

$$X_{jk} = \text{Total necessary quantity of the part } a_j \text{ to be utilized for producing the products of the determined sequence from first to } K \text{th.}$$

With these notations in mind, the following two values can be developed:

$$\frac{N_j}{Q} = \text{Average necessary quantity of part } a_j \text{ per unit of a product}$$

$$\frac{K \cdot N_j}{Q} = \text{Average necessary quantity of part } a_j \text{ for producing } K \text{ units of products.}$$

In order to keep the consumption speed of a part  $a_j$  constant, the amount of  $X_{jk}$  must be as close as possible to the value of  $(K \cdot N_j / Q)$ . This is the basic concept underlying this sequencing algorithm and is depicted in the Figure 5-2.

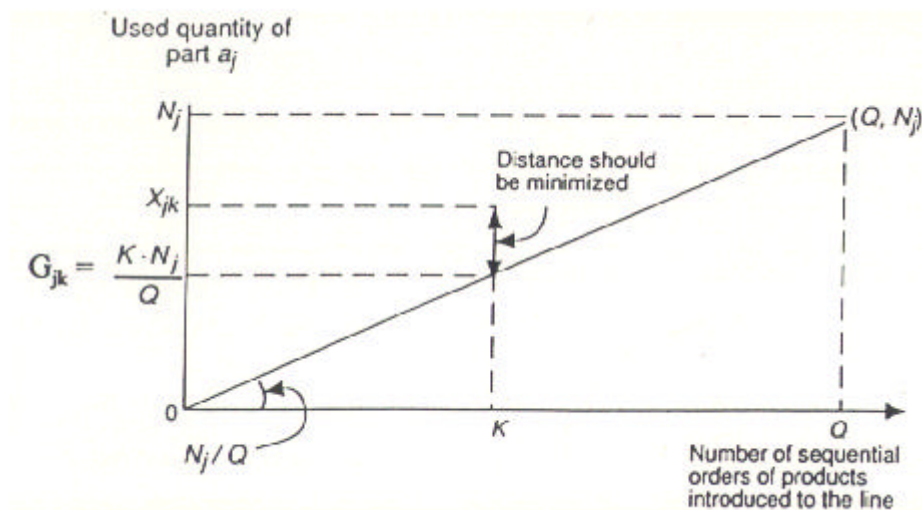


Figure 5-2: Relationship between  $X_{jk}$  and  $(K \cdot N_j) / Q$

It can now be further defined that:

$$\text{A point } G_k = (K \cdot N_1 / Q, K \cdot N_2 / Q, \dots, K \cdot N_\beta / Q),$$

$$\text{A point } P_k = (X_{1k}, X_{2k}, \dots, X_{\beta k}).$$

In order for a sequence schedule to assure the constant speed of consuming each part, the point  $P_k$  must be as close as possible to the point  $G_k$ . Therefore, if the degree is measured for point  $P_k$  approaching point  $G_k$  by using the distance  $D_k$ :

$$D_k = \|G_k - P_k\| = \sqrt{\sum_{j=1}^b \left( \frac{K \cdot N_j}{Q} - X_{jk} \right)^2}$$

then, the distance  $D_k$  must be minimized for  $k$  between 0 and  $Q$ . The algorithm developed on this idea is given in Appendix 8 and is implemented as part of the simulation model to generate the appropriate sequence.

### **5.3 Model Inputs**

Almost all real systems contain one or more sources of randomness. Likewise, in the system being modeled in this research, the sources of randomness include processing times at the various stages on the feeder line, quality fallout, machine operating time before a breakdown and machine repair time. It is generally necessary to represent each source of system randomness by a probability distribution (rather than just its mean) in the simulation model. If it is possible to collect data on the source of randomness, these data can then be used in one of the following ways to specify a distribution:

1. The data values themselves can be used directly in a *trace-driven simulation*. For example, if the data represent service times, then one of the data values is used whenever a service time is needed.
2. The data values can be used to define an empirical distribution function and this distribution can be sampled whenever a data value is needed in the simulation.

3. Standard techniques of statistical inference are used to fit a theoretical distribution form (e.g., exponential, normal, poisson, etc.) to the data and to perform hypothesis tests to determine the goodness of fit.

In some simulation studies it may not be possible to collect data on the random variables of interest. For example, the system being modeled does not exist or there is no easy access to a real system that is representative of the model being simulated. Law and Kelton(1991) describe the following heuristics for choosing a distribution in the absence of data, which have been used by many simulation practitioners:

1. “Experts” in the kind of system being modeled are asked for their most optimistic estimate, most likely estimate and pessimistic estimates for the values that a random variable can take, say  $a$ ,  $b$  and  $c$  respectively. Given  $a$ ,  $b$  and  $c$  the random variable is considered to have a triangular distribution on the interval  $[a, b]$  with mode  $c$ .
2. Similar to the procedure described above, but instead of a triangular distribution, fit a beta distribution on the interval  $[a, b]$ .
3. Use existing research in the field under study as a basis for the distributions.

The literature survey conducted as part of this research and detailed in Chapter 4 revealed that (refer to section 4.1.5.3) most researchers have adopted a normal distribution for processing times in JIT systems. This research also assumes normal distribution for the processing times. For the time between machine breakdowns it assumes an exponential distribution and for repair times of the machines it assumes a normal distribution. The values for the various distribution parameters were identified through pilot runs and are detailed in the following sections.

### 5.3.1. Master Production Schedule

The master production schedule (MPS) in this research, gives the final assembly line information as to how many items of a given product type need to be produced during a shift. Typically, the MPS is the result of a forecast and is developed for a couple of months into the future. It is then translated into a shift production schedule and the assembly line is given this schedule at the beginning of every shift. Adjustments are made to the MPS depending on the market situation. However for the sake of simplicity, this research assumes a constant MPS throughout the horizon for which the model is run. Since the number of product types assembled on the final assembly line is one of the experimental factors in this study, different MPS's are used depending on the number of product types being assembled. Table 5-2 shows the various MPS's used in this research.

Table 5-2: Master Production Schedule Used

No. of Products Types assembled	Product Type - 1	Product Type - 2	Product Type - 3	Product Type - 4	Product Type - 5	Total
3	20	22	18	-	-	60
4	15	16	15	14	-	60
5	13	11	12	14	10	60

Since 60 products need to be assembled during each shift and if an 8 hour shift is assumed, the cycle time becomes 8 minutes. So every 8 minutes a product will be launched on to the assembly line. The product type is dependent on the output of the sequencing algorithm.

### 5.3.2 Processing Times

The processing times used at the various stages for the various product types on the feeder line are shown in Table 5-3 through Table 5-7. As discussed earlier, the processing times at all the stages are assumed to be normally distributed. Since processing time variability is an experimental factor in this research, it helps to define the *coefficient of variability* (CV) which is the standard deviation divided by the mean. The table below shows the standard deviations associated with the three levels of CV used in this research:

Table 5-3 : Processing times at various stages for product type 1

	Mean (min)	$\sigma$ for CV of 0.1	$\sigma$ for CV 0.15	$\sigma$ for CV of 0.2
WS1	8.2	0.82	1.23	1.64
Stage 1	7.5	0.75	1.125	1.5
Stage 2	7.7	0.77	1.155	1.54
Stage 3	7.6	0.76	1.14	1.52
Stage 4	7.1	0.71	1.065	1.42
Stage 5	6.8	0.68	1.02	1.36

Table 5-4 : Processing times at various stages for product type 2

	Mean (min)	$\sigma$ for CV of 0.1	$\sigma$ for CV 0.15	$\sigma$ for CV of 0.2
WS1	7	0.7	1.05	1.4
Stage 1	6.2	0.62	0.93	1.24
Stage 2	7.4	0.74	1.11	1.48
Stage 3	7.3	0.73	1.095	1.46
Stage 4	6.9	0.69	1.035	1.38
Stage 5	6.9	0.69	1.035	1.38

Table 5-5 : Processing times at various stages for product type 3

	Mean (min)	$\sigma$ for CV of 0.1	$\sigma$ for CV 0.15	$\sigma$ for CV of 0.2
WS1	8	0.8	1.2	1.6
Stage 1	7.9	0.79	1.185	1.58
Stage 2	8	0.8	1.2	1.6
Stage 3	8.25	0.825	1.2375	1.65
Stage 4	7	0.7	1.05	1.4
Stage 5	6.9	0.69	1.035	1.38

Table 5-6 : Processing times at various stages for product type 4

	Mean (min)	$\sigma$ for CV of 0.1	$\sigma$ for CV 0.15	$\sigma$ for CV of 0.2
WS1	7.2	0.72	1.08	1.44
Stage 1	7.3	0.73	1.095	1.46
Stage 2	7.5	0.75	1.125	1.5
Stage 3	7.1	0.71	1.065	1.42
Stage 4	6.9	0.69	1.035	1.38
Stage 5	6.7	0.67	1.005	1.34

Table 5-7 : Processing times at various stages for product type 5

	Mean (min)	$\sigma$ for CV of 0.1	$\sigma$ for CV 0.15	$\sigma$ for CV of 0.2
WS1	7.4	0.74	1.11	1.48
Stage 1	7.2	0.72	1.08	1.44
Stage 2	7.6	0.76	1.14	1.52
Stage 3	7.5	0.75	1.125	1.5
Stage 4	6.8	0.68	1.02	1.36
Stage 5	7	0.7	1.15	1.4

### 5.3.3. Quality defectives and Machine Breakdowns

The process at each of the stages is assumed to yield 5% defectives. The time for the quality inspection is incorporated into the processing times at the stages. If an item is identified to be defective, it has to be reworked by the operator at the stage. The rework times are assumed to follow a normal distribution and are given in the Table 5-8. The stages on the feeder line break down according to an exponential distribution and their repair time follows a normal distribution. Stage break down and repair distributions are given in Table 5-9.

Table 5-8 : Quality rework times at various stages

	Mean (min)	$\sigma$ (min)
Stage 1	3.1	0.8
Stage 2	3.2	0.84
Stage 3	3.4	0.78
Stage 4	3.3	0.9
Stage 5	3.2	0.64

Table 5-9 : Time between failures of the stages and their repair times

	Mean time between failures (min)	Mean time to repair (min)	$\sigma$ of repair time (min)
Stage 1	20, 000	45	5
Stage 2	20, 100	50	8
Stage 3	20, 150	40	5
Stage 4	23, 125	55	7.5
Stage 5	21, 230	62	8



## **5.4 Simulation Model**

Previous sections of this chapter contain descriptions of the important components of the system being modeled. Once the system has been clearly defined, the next step is to translate the system into a computer simulation model. The following sections deal with the task of constructing the computer representation of the model.

### **5.4.1 Simulation Language**

SLAM II, a Fortran based simulation package, was used for this study. It was selected for the following reasons:

1. It allows for easy scheduling of events and collection of statistics.
2. It provides the user with built-in distributions and easy ways to generate data for the processing times, time between machine breakdowns and machine repair times.
3. It automatically maintains all the files and makes access to the user very easy.
4. Both SLAM and Fortran are quite popular and easily accessible.

### **5.4.2 Simulation Model Characteristics and Assumptions**

The simulation model was structured to represent the JIT system described in section 5-1. The following assumptions were made and maintained throughout the simulation:

1. The shift length is 8 hours or 480 minutes.
2. The system operates without any interruptions other than machine breakdowns. Time between machine breakdowns and their repair times are listed in section 5.3.3. After the repair, the machine will continue operating on the job from the point at which it was interrupted.

3. The master production schedule remains fixed throughout the simulation run. The master production schedules used in the simulation are listed in section 5.3.1. In all cases 60 products are to be manufactured. Due to assumption 1, the cycle time is 480 minutes/60, i.e., 8 minutes.
4. The sequencing algorithm described in section 5.2 generates the sequence to be used at the beginning of the simulation run and the same sequence is used from shift to shift for the period of the simulation run.
5. The feeder line that is being simulated is assumed to be feeding into the workstation at the head of the final assembly line. This assumption will cause production triggers to occur at every cycle time. It is also a way of eliminating uncertainty associated with the demand process.
6. Lot sizes are assumed to be 1 unit throughout the system. This assumption has the implication that each stage requests only 1 item from its preceding stage. So, for example, when a demand is placed at Workstation-1 on the final assembly line, it will consume 1 item from the output storage location of Stage-1 on the feeder line. Stage-1 will consume 1 item from its input storage location causing Stage-2 on the feeder line to produce 1 item, and so on.
7. The kanban posts are monitored constantly. So when a production ordering kanban is posted at Stage-1's kanban post, if the operator is free, then this kanban will trigger production at Stage - 1. Similarly, if a withdrawal kanban is posted at Stage -1's withdrawal kanban post, it will immediately cause the withdrawal of an item from Stage-2's output storage area if an item matching this request is present in Stage-2's output storage area. This assumption may be a deviation from reality where the kanban posts may only be polled periodically.
8. Time to detach kanbans from the items and attach kanbans to the items are all done by the operator and is assumed to be part of the processing time at that stage.

9. The time required to transfer an item from one stage's output storage location to another stage's input storage location when a withdrawal kanban is satisfied, is assumed to be a constant of 1 minute. It is also assumed that there is no operator intervention to achieve this task. This assumption is in line with the thought that there might be a chute or some other form of conveyance connecting the two stages.
10. The raw materials at the tail of the feeder line are assumed to be infinite. So there is no delay in replenishing the input items at that stage. Typically, the tail of the feeder lines may be connected to other supply-chains. This assumption is made in order to avoid complicating the model.
11. Each stage produces 5% defective items which are then reworked. The operator will inspect the job after he has completed work on it, and if need be he/she will rework the job to make it conform to quality specifications. Its also assumed that the rework will make the product to be of acceptable quality. So, there are no discarded jobs and a single rework is sufficient to make the defective item conform to quality specifications. The time spent in rework at various stages are provided in section 5.3.3. The time the operator spends inspecting the job is factored into its processing time.

### **5.4.3 Elements of the simulation model**

The discrete event simulation model developed in this research consists essentially of the following elements:

### ***5.4.3.1 Initialization and Sequencing of Products***

Before the simulation model can be run, it is necessary that the proper initial conditions be set. These conditions include the shift length, the number of product types to be produced on the final assembly line, the total number of products to be assembled, generating an appropriate sequence of the products to be assembled on the final assembly line, etc. Since the simulation program has been written using SLAM network models and user-written Fortran inserts, the program begins execution in the subroutine INTLC. Subroutine INTLC in turn calls subroutine SEQUENCE which performs all the initializations mentioned above.

The number of product types to be assembled on the final assembly line, the number of items of each product type to be manufactured and the cycle time are all variables in this subroutine. These variables thus offers the capability of easily adapting the model to simulate different scenarios. Once these variables have been set to appropriate values, the goal chasing algorithm is used to come up with an appropriate sequence.

After returning from the subroutines SEQUENCE and INTLC, control statements in the network model are executed. The ENTRY control statements are used to place entities into the output storage areas and the input storage areas at the various stages. The number of entities placed in the output storage area of a stage using the ENTRY statement controls the number of production kanbans at that stage. Similarly the number of entities placed in the input storage area of a stage using the ENTRY statement determines the number of withdrawal kanbans at that stage. Attributes assigned to these entities determine whether they are production ordering kanbans or withdrawal kanbans. Attribute values assigned to the production ordering and withdrawal kanban entities also determine the product type that they represent. The details of the attributes and the information they contain are given in Appendix 9 along with the complete listing of the simulation program.

#### ***5.4.3.2 Demand Generation on the final assembly line***

An entity is created with the time between arrivals equal to the cycle time, as determined in subroutine SEQUNCE. To determine the product type, the EVENT node causes subroutine EVENT(JEVNT) with JEVNT equal to 1, to be called every time an entity arrives at the EVENT node. The EVENT subroutine in turn invokes subroutine NXT which assigns the product type to the entity. Subroutine NXT determines the product type by using the sequence determined by subroutine SEQUNCE.

The arriving demand entity will be put into a queue if the workstation on the final assembly line is busy. If the workstation is idle, processing will start on the entity. However, before processing can start, a check is made to determine if there is a raw material of the same type as the demand in the output storage area of the feeder line. If a raw material of an appropriate kind is not present, then it's considered as a *stockout* situation. Subroutine STOCK handles identification of stockout situations and the accumulation of the stockout statistics. The time that the entity spent from the moment it enters the system to the moment it finishes processing at the workstation on the final assembly line is defined as the time in system, and statistics regarding this value are collected by the COLCT node in the simulation network.

#### ***5.4.3.3 Machine breakdowns and repair***

An entity is generated for each stage and this entity is delayed for a period corresponding to a random number drawn from the distribution associated with the stage's time between breakdown. When this entity finishes its delay, it will preempt the stage and keep track of the duration of service remaining on the job that the stage was processing. The entity is then delayed for a period corresponding to a random number drawn from the repair distribution associated with the stage. When this delay is completed, the job that the stage was working on before being preempted is restarted with the remaining part of processing.

#### ***5.4.3.4 Production ordering kanbans***

Whenever an item is removed from the output storage area of a stage, a production ordering kanban is posted at the stage's production ordering kanban post. In the simulation model, there are no separate entities to represent the item and the production ordering kanban. The entities that are created at the time of initialization through the ENTRY statements represent both an item and the production ordering kanban. When this entity is in the output storage area (modeled as a QUEUE) associated with the stage, it may be visualized as representing the item to which a production ordering kanban has been attached. When an item is withdrawn from the output storage area, by the use of MATCH and BATCH nodes, it should be visualized as representing the "consumption" of the item and "detachment" of the production ordering kanban. The "detached" production ordering kanban is placed in the production ordering kanban post (modeled as a QUEUE). The use of a MATCH node between the production ordering kanban post and the input storage area for that stage (modeled as another QUEUE), ensures that a check is made to see if a raw material exists. If there is a raw material, then processing will start at that stage whenever the stage becomes free. The duration between the time that the production ordering kanban arrived at the production ordering kanban post and the time when production actually started at the stage is termed as the kanban waiting time. Statistics are collected for this statistic using the COLCT node.

#### ***5.4.3.5 Withdrawal kanbans***

Whenever an item is removed from the input storage area of a stage, a withdrawal kanban is removed from the item and posted at the stage's withdrawal ordering kanban post. In the simulation model, there are no separate entities to represent the item and the withdrawal kanban. The entities that are created at the time of initialization through the ENTRY statements represent both an item and the withdrawal kanban. When an entity is in the input storage area (modeled as a QUEUE) associated with a stage, it may be

visualized as representing an item to which a withdrawal kanban has been attached. Thus, when an item is withdrawn from the input storage area, by the use of MATCH and BATCH nodes, it should be visualized as representing the consumption of the item and detachment of the withdrawal kanban. The “detached” withdrawal kanban is placed in the withdrawal kanban post (modeled as a QUEUE). The use of a MATCH node between the withdrawal kanban post and the output storage area of the preceding stage (modeled as another QUEUE), ensures that a check is made to see if an item exists at the preceding stage. If there is an item in the output storage area of the preceding stage, then this item is transferred to the stage that requested the item through the withdrawal kanban. The duration between the time that the withdrawal kanban arrives at the withdrawal kanban post and the time when an item was actually withdrawn from the preceding stages’ output storage area is termed as the kanban waiting time and statistics are collected for this statistic using the COLCT node.

#### **5.4.3.6 *End of a shift***

Entities are created every 480 minutes (8 hour shifts are assumed) to signify the end of a shift. When an end of shift entity arrives at an EVENT node, it causes the EVENT node to call subroutine EVENT(JEVNT) with JEVNT equal to 10. The EVENT subroutine causes subroutine SHIFTOPT to be invoked. Each shift generates a sample for the statistics that are of interest in this research. So, this subroutine contains code to evaluate the stockout, the throughput, work in progress, time in system, kanban waiting times and utilization at the various stages on the feeder line. This subroutine also resets variables for the collection of statistics during the next shift.

#### **5.4.4 Model testing and verification**

Model testing and verification is a process whereby the simulation model is evaluated to determine whether it satisfactorily duplicates the behavior of the system. In order to test and verify it, the simulation model was executed for short periods of time under varying conditions, by the computer, with the TRACE feature of SLAM turned on. The TRACE feature provides the modeler with the ability to view the flow of entities through the system. The flow of entities was then analyzed to verify that they are following the appropriate rules. The simulation model also contains various subroutines. Modular testing was conducted to verify that each of these subroutines was producing outputs as expected.

#### **5.5 Design of Experiment**

The objective of this research is to analyze the relationship between various performance measures and design and operational factors associated with the kanban controlled feeder line. The performance measures were:

1. Time in system
2. Throughput
3. Stockout
4. Kanban waiting times
5. Work in progress inventory
6. Utilization of stages

The experimental factors and the levels used for these factors are shown on the next page in Table 5-10.



Table 5-10 : Experimental factors and their levels

	Level - 1	Level - 2	Level - 3
Number of Stages on the feeder line	3	4	5
Number of product types manufactured	3	4	5
Number of kanbans of each product type	1	2	3
Coefficient of variation of processing times	0.1	0.15	0.2

A  $3^4$  factorial experiment was designed and is shown in Appendix 1. Due to the resource intensive nature of the simulations, no replications were made.

## **Chapter 6**

### **Analysis of Results**

Once the simulation model was tested and verified, the factorial experiments were performed and data were collected. The simulation model provided the following data:

1. Average time in system for the products being assembled at the workstation on the final assembly line.
2. Average work in process on the feeder line.
3. The number of stockouts at the workstation on the final assembly line during a shift.
4. Average waiting times of the production ordering and withdrawal kanbans on the feeder line.
5. Throughput at the workstation on the final assembly line during a shift.
6. Average utilization of all the stages on the feeder line.

Details of the results of the  $3^4$  factorial experiment are provided in Appendix 1. Certain experimental trials resulted in infeasible combination of the experimental factors and hence no steady state values for the performance measures were obtained for those trials. The following sections of this chapter analyze each of the performance measures and tries to characterize how they react to changes in the design and operational factors.

#### **6.1 Time In System**

Time in system as defined in this research is the time between the arrival of a demand at the first workstation (please refer to Figure 5-1) on the final assembly line and completion of processing at that workstation. Time in system is an important metric because of the fact that the final assembly line has a specified cycle time and if the time in system exceeds the cycle time then it will result in the stoppage of the assembly line. Ideally, the time in

system must be close to the cycle time. But it may deviate from the cycle time due to various factors. This research studies the behavior of the time in system under varying design and operating conditions on the feeder line that feeds the first workstation.

The time in system obtained for the various experiments are shown in Appendix 1. An analysis of variance (ANOVA) was performed to determine the factors that influence time in system and the results of this ANOVA are shown in Appendix 2. Using a 95% statistical significance level, the ANOVA table indicates that time in system is influenced by the number of kanbans, number of product types being assembled on the line and the level of processing time variability at the stages on the feeder line. The ANOVA table also indicates that the interaction effects due to number of kanbans and number of stages, number of kanbans and number of products, number of kanbans and level of processing time variability, and the number of product types and level of processing time variability are significant.

Figure 6-1 contains a plot showing the variation of time in system with the number of kanbans. This figure indicates that the time in system is strongly influenced by the number of kanbans maintained at the various stages on the feeder line. The figure indicates that the time in system is above the cycle time of 8 minutes when there is 1 kanban of each product type maintained at the stages on the feeder line. As soon as the number of kanbans is

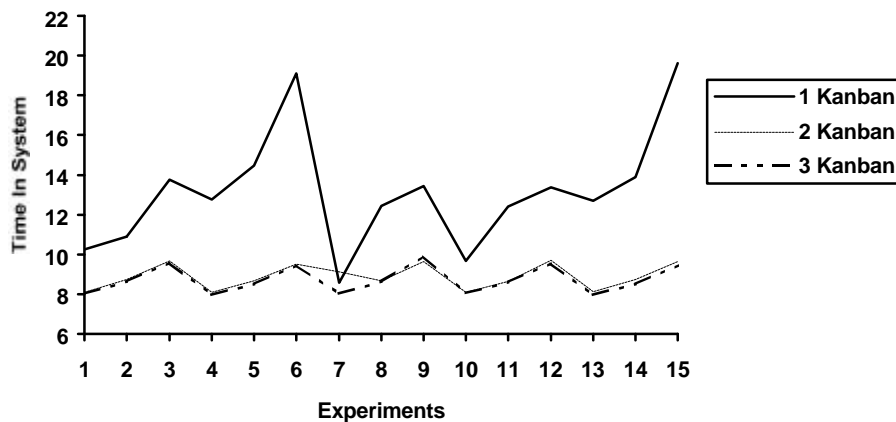


Figure 6-1 : Relationship between time in system and number of kanbans

increased to 2, the time in system approaches the cycle time. However, when the number of kanbans is increased to 3 from a level of 2, there is not much difference in the behavior of time in system. The fact that there is no significant change in the behavior of time in system from kanban level 2 to kanban level 3 suggests that there might be a “threshold” level for the number of kanbans beyond which any further increase in the number of kanbans does not impact the time in system.

This behavior of time in system with variation in the number of kanbans can be explained by noting that the number of kanbans maintained at a stage is an indication of the level of inventory maintained in the input and output storage areas of the stage. Thus, when 1 kanban of each product type is maintained at each stage, the risk of running out of inventory is increased as compared to when 2 kanbans are maintained. This risk of running out of inventory translates into more delays thereby increasing the time in system. To understand this risk, consider the following case: suppose that the sequence of items to be assembled on the final assembly line is A A B C. Also imagine that only 1 kanban of each product type is maintained. Thus when the first item of type A is assembled, it consumes the item that the kanban represents and triggers production on the feeder line to replenish the item consumed. However, before the item can be replenished by the feeder line, demand for another A occurs on the final assembly line and this demand has to wait until an item is ready, thereby increasing its time in system.

As the graph in Figure 6-1 depicts, the time in system seems to stabilize and approach the cycle time by increasing the number of kanbans from a level of 1 to a level of 2. This is an indication that the inventory of items at the storage locations seems sufficient to meet the demand requirements. Since the risk of running out of inventory has been overcome by increasing the level of kanbans from 1 to 2, there is no further advantage gained by increasing the number of kanbans above the level of 2. This conclusion can also be drawn from the graph in Figure 6-1 where the behavior of time in system for kanban levels 2 and 3 are almost the same.

Figure 6-2 is a graphical representation of the variation of time in system with the number of product types. The figure shows that, under all experimental scenarios, the time in system decreases with the increase in the number of product types being assembled. This behavior can be attributed to the fact that with increasing number of products being assembled, the frequency for assembly of the same product type decreases. As a result, the feeder lines will have a greater chance to finish producing a sub-component of a particular product type and have it in inventory by the time an order is placed on the final assembly

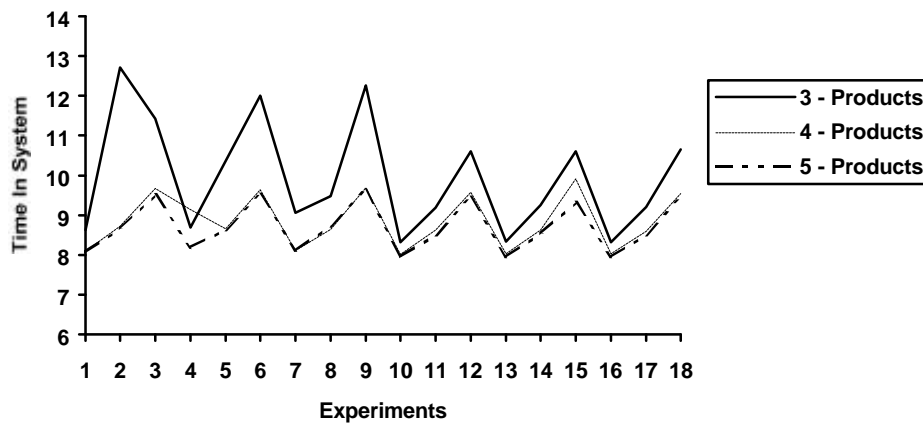


Figure 6-2 : Relationship between time in system and number of product types.

line for that particular product type. For example, consider that a sequence of product types to be assembled is A B C D A C. After the first product of type A is assembled, there are three intermediate products to be assembled till the next product of type A. This gap between placement of orders for product type A will enable the raw-material for product type A to be replenished at the feeder line thereby allowing the next order for product type A to be assembled without waiting for raw-materials. However, increasing the number of product types being assembled above a particular “threshold” has no impact on the time in system. This behavior is clearly evidenced by the similarity of the time in system for the systems with number of product types equal to 2 and 3.

Figure 6-3 is a plot of the variation of time in system with the number of stages on the feeder line. The figure seems to indicate that the behavior of time in system for all the levels of number of stages is approximately the same. Although one would expect that increasing the number of stages on the feeder line would increase the time in system due to the uncertainties at the stages, this conclusion cannot be drawn from this graph. Further research needs to be done to determine the relationship between number of stages and time in system.

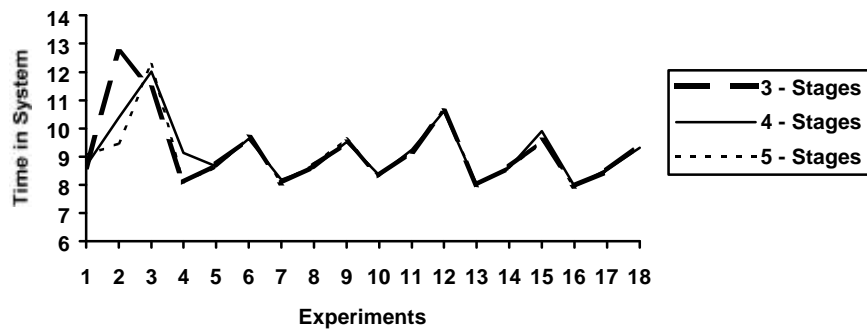


Figure 6-3 : Relationship between time in system and number of stages on the feeder line

Figure 6-4 is a graphical representation of the variation in time in system with levels of processing time variability. We can conclude that greater variability in processing time at the stages on the feeder line leads to a greater time in system.

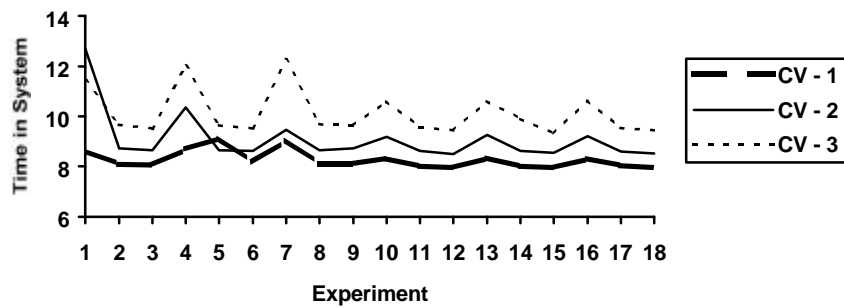


Figure 6-4 : Relationship between time in system and processing time variability

The interactions may be best studied by plotting a graph of average time in system versus the factors of interest. Figure 6-5 shows the graph for the interaction between number of kanbans and number of product types.

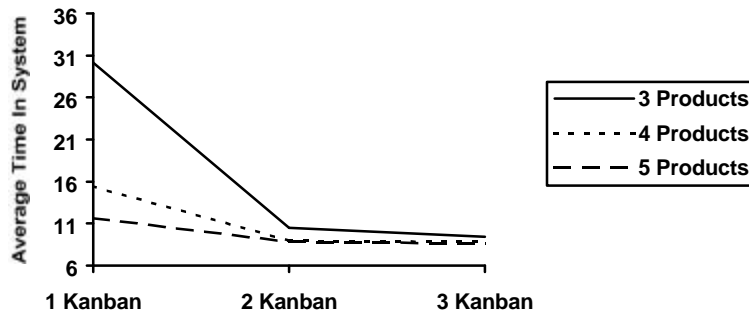


Figure 6-5 : Interaction between number of kanbans and number of product types

The graph depicts that when there is 1 kanban, the time in system decreases with increase in the number of product types. When there are 2 kanbans, the time in system still decreases with increase in the number of product types. However, the magnitude of decrease in time in system when there are 2 kanbans is different from the magnitude of decrease when there is 1 kanban in the system. As mentioned previously, when there is only 1 kanban in the system, the system is exposed to the risk of running out of inventory. However by increasing the number of product types being assembled, this risk is reduced and the time in system decreases due to the decrease in the frequency between placement of orders for the same product type. But when the system has 2 kanbans, the “threshold” number of kanbans has been reached and the risk of running out of inventory has already been overcome. Thus when the number of product types are increased, the demand frequency of orders for the same kind of product still decreases, but does not translate into a decrease in time in system.

Figure 6-6 is a plot of the interaction between the number of kanbans and the number of stages on the feeder line. Although the Type-I SS in the ANOVA table (Appendix 2)

indicates that there is a significant interaction between the number of kanbans and number of stages, the Type-III SS indicates that the interaction effect is not significant. When Type-I and Type-III SS have different conclusions regarding the significance of an effect, it is an indication that the effect is confounded with the effect of another variable. Thus no conclusion can be drawn regarding this interaction and further research needs to be made to understand this relationship.

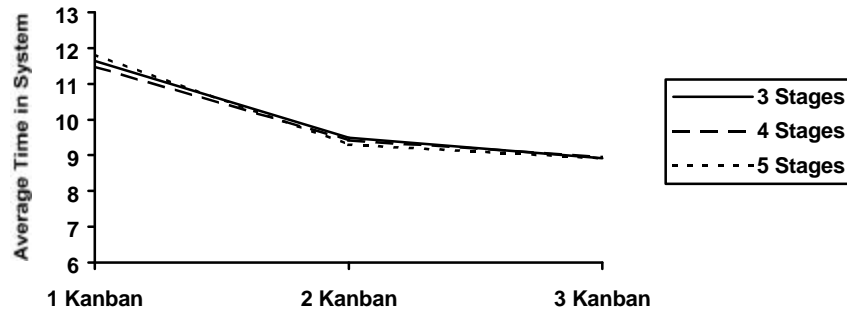


Figure 6-6 : Interaction between number of kanbans and number of stages

Figure 6-7 is a plot showing the interaction between number of kanbans and the level of variability. The graph indicates that at lower levels of the number of kanbans, the impact of level of variability in processing times is greater on the time in system than at higher levels of kanbans. This behavior can again be explained by using the argument that at lower levels of kanbans, there is some risk of running out of inventory. This risk is all the

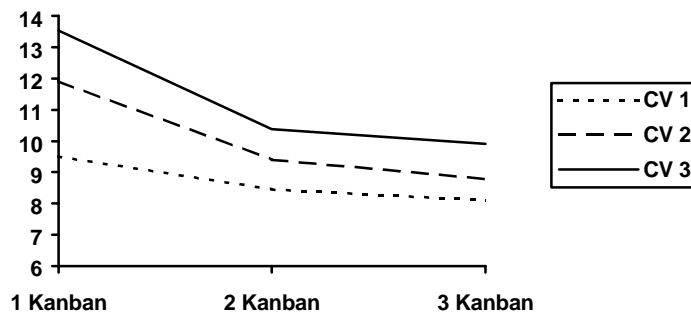


Figure 6-7 : Interaction between number of kanbans and level of variability



more increased if the variation in processing times is increased. Therefore, having low kanbans and high variability in processing times has the cumulative effect of increasing Time in system. On the other hand, if the number of kanbans is increased and the variation in processing time is also increased, then there is certainly an increase in the risk of running out of inventory due to increasing variability in processing times. But this increase in risk is offset in part by the fact that there are now more kanbans. Thus, the behavior of time in system is different at different levels of kanbans and processing time variability.

Appendix 2 also indicates that the interaction between number of product types and variability is also significant. This interaction is graphically represented in Figure 6-8.

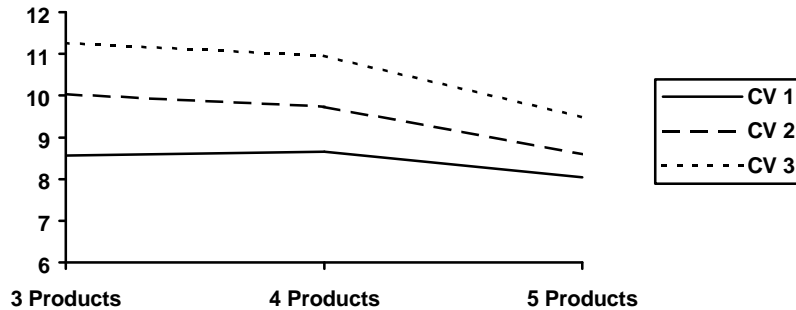


Figure 6-8 : Interaction between number of product types and level of processing time variability

This interaction is similar to the interaction between the number of kanbans and the level of variability. Increasing the number of product types has the effect of decreasing the frequency with which orders for the same item are placed. This decrease in frequency allows the feeder line to replenish items that have been consumed. Therefore with increase in the number of product types being assembled, the time in system decreases. However, coefficient of variation has an effect on the decrease in the time in system that is observed. For example, when there are 3 product types being assembled, the feeder line is more sensitive to changes in processing time variability as compared to the case where there are 4 product types being assembled. This is due to the fact that when there are 4 product types being assembled as opposed to 3 product types, the time in system is already

reduced. Thus any further increase in variability will increase the time in system, but not by the same magnitude as compared to when there are 3 product types.

## **6.2 Work In Process**

The principal consideration of the Toyota Production System and other JIT manufacturing systems is to reduce costs by reducing waste to a minimum. As mentioned in Chapter 2, four kinds of waste can be found in manufacturing operations: excessive production resources, overproduction, excessive inventory and unnecessary capital investment. As the term indicates, Work-In-Process (WIP) includes all the material resources which are in an intermediate condition, pending to be converted to the final product. WIP is an ingredient of all the above mentioned wastes and thus serves as an excellent measure of the waste in the manufacturing operations. Due to its direct bearing on the operation of the manufacturing system, an investigation into the relationship between WIP and the experimental factors would provide valuable insights to the manager, in streamlining the manufacturing operations. As also mentioned in Chapter 5, each stage has two inventory locations: an output storage area and an input storage area. WIP in this research is the time-average of the quantities in the various inventory locations along the feeder line. Obtaining a measure of this statistic is easily accomplished by the TIMST statement provided by SLAM.

The values for WIP that were obtained during the various experiments performed as part of this research are shown in Appendix 1. Appendix 3 contains the results of the ANOVA performed on WIP. This ANOVA table indicates that at a 95% significance level, the main effects of the number of kanbans, number of stages on the feeder line, number of product types being assembled and the level of variability in processing times are statistically significant. The table also indicates that certain two way interactions are also significant. The two-way interactions that are significant include the following: the interaction between number of kanbans and number of stages on the feeder line, interaction between

number of kanbans and number of products being assembled, and the interaction between number of stages and the number of product types.

Figure 6-9 depicts the variation of work-in-process with the number of kanbans. As can be seen from this figure, the more the number of kanbans that are maintained at each of the

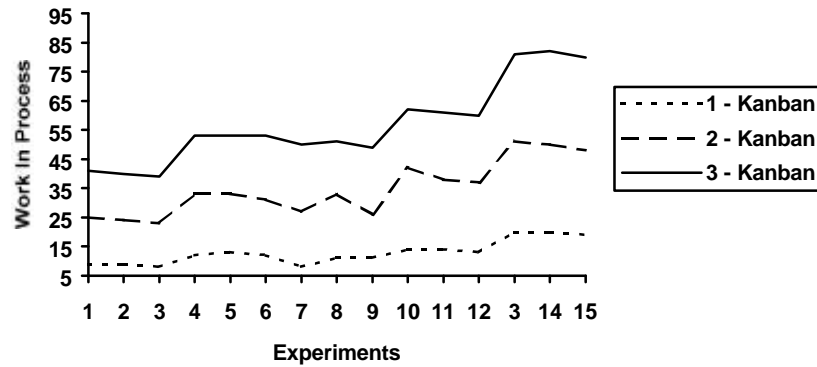


Figure 6-9 : Relationship between work in process and number of kanbans

stages, the more work-in-progress will exist. Each kanban represents an item of inventory in the inventory storage location associated with that kanban. Thus a withdrawal ordering kanban represents an item of inventory in the input storage area of the stage, while a production ordering kanban represents an item of inventory in the output storage area. By increasing the number of kanbans, the number of inventory items in the storage locations increases and therefore the work-in-process increases correspondingly.

Figure 6-10 depicts the variation of work-in-process with the number of stages. As can be seen from this graph, with increase in the number of stages, there is an increase in the work in progress. This behavior is in tune with the intuitive notion that with an increase in the number of stages on the feeder line, the number of inventory locations also increases thereby causing more work-in-process.

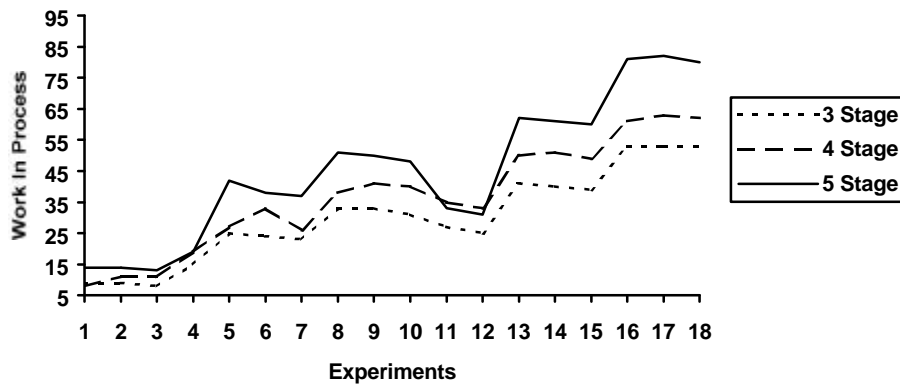


Figure 6-10 : Relationship between work in process and number of stages

An interesting thing to note is that although each unit increase in the number of stages is accompanied by the addition of 2 additional inventory locations, the work-in-process doesn't increase by a factor of 2. This is an indication of the interaction effects that may exist. The interaction effects are described in detail later in this section.

The number of product types that are assembled also has a significant effect on the work-in-process. Figure 6-11 gives a graphical representation of the variation of work-in-process with the number of product types.

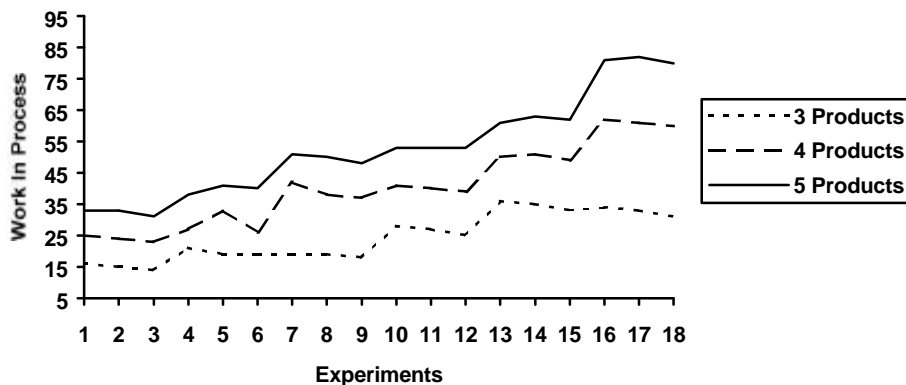


Figure 6-11 : Relationship between work in process and number of product types

The impact that the number of product types has on the work-in-process is similar to the impact that the number of kanbans has on work-in-process. With increase in the number of product types being assembled, the work-in-process also increases. This behavior may be explained by noting that with an increase in the number of product types, the magnitude of the number of kanbans increases (there may still be only 1 kanban per product type, but if there are 3 product types, then there are a total of 3 kanbans, 1 for each product type). The increase in the number of kanbans, as we have already seen, causes more inventory items to be kept at the various inventory storage locations thereby leading to a larger work-in-process.

The other main effect on work-in-process studied in this research is the level of processing time variability. Long processing times at a stage for a particular product type may cause items of other product types to reside in the inventory storage locations of preceding stages while the item completes processing thereby causing an overall increase in work-in-process. On the other hand, short processing times will “pull” items from preceding stages more quickly causing an overall decrease in work-in-process. However, due to processing time variability, in the long run, the increase in work-in-process due to longer processing times may be balanced by lower work-in-process during instances of short processing times. Thus the average work-in-process may be largely unaffected by the level of processing time variability. The ANOVA table in Appendix 3 indicates that the main effect of the level of processing time variability is statistically significant at a 95% significance level. But this effect has the lowest F-value among all the other main effects and thus may not be a major contributor to work-in-process.

Figure 6-12 shows the variation of work-in-process with the level of processing time variability. This figure indicates that the work-in-process remains more or less the same for all the levels of variability in processing times.

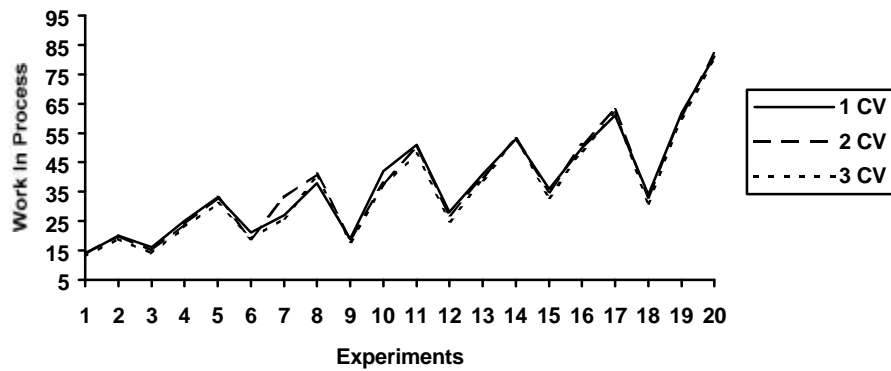


Figure 6-12 : Relationship between work in process and processing time variability

In the light of this behavior of work-in-process under varying levels of variability in processing times and also due to the low F-Value (refer to Appendix 3) for this statistic, one is tempted to infer that work-in-process is not affected by variability in processing times despite of the ANOVA table in Appendix 3 indicating otherwise. As pointed out in Chapter 7, further research needs to be conducted to establish the true nature of this behavior.

The ANOVA table in Appendix 3 indicates that there are two-way interactions that are significant. One of these interactions is the interaction between the number of kanbans and the number of stages on the feeder line. To analyze the nature of the interaction effects, it is helpful to plot the average work-in-process at different levels of the number of kanbans and the number of stages on the feeder line. Figure 6-13 is a graph depicting the average work-in-process at different levels of the factors involved in the interactions. The lack of parallelism between the lines in the graph is an indication that there is interaction between the number of kanbans and number of stages. This interaction makes the work-in-process behave differently under different combinations of number of kanbans and number of stages. For example, the figure indicates that work-in-process increases with increase in the number of stages under both the 1-kanban situation as well as the 2-kanban situation.

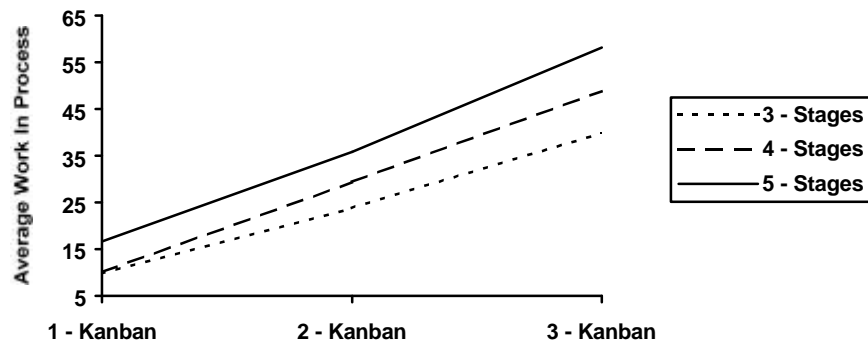


Figure 6-13 : Interaction between number of kanbans and number of stages

However, the magnitude of increases in the 1-kanban situation and the 2-kanban situation are different. To understand this interaction, it is important to remember that when there is only 1 kanban of each product type at each stage, the risk of running out of inventory when a demand is placed for a product type is greatly increased. The fact that the risk of running out of inventory is high under this situation is an indication that the inventory locations on the feeder line are sparsely populated. The sparse population of the inventory locations leads to a low work-in-process. Even when more stages are added (thereby resulting in more inventory locations) to the feeder line, under this kanban condition, the sparse population of the inventory locations leads to a low work-in-process. However, when the number of kanbans is increased to 2, the risk of running out of inventory when a demand is placed is decreased. The decrease in risk is because now there are more number of kanbans which has the effect of having more number of items in the inventory storage locations. Because there is an increase in the number of items in the inventory storage locations, there is more work-in-process. Adding more stages under this situation only leads to a further increase in work-in-process.

Another significant interaction is between the number of kanbans and the number of product types. Again it helps to plot the graph in Figure 6-14 depicting the average work-in-process at different levels of these factors.

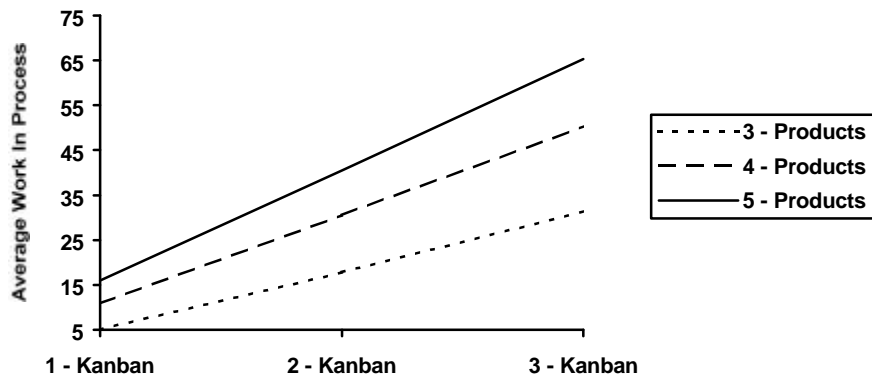


Figure 6-14 : Interaction between number of kanbans and number of product types

When the feeder line has 1 kanban of each product type, it has more risk of running out of inventory than when the feeder line has 2 kanbans of each product type. Correspondingly, a 1 kanban feeder line has lower work-in-process than a 2 kanban feeder line. Now consider the case where there is 1 kanban at the each of the stages on the feeder line. Under this situation, if the number of product types are increased, then as has already been discussed previously, the frequency of orders for a product type decreases. This decrease in frequency of orders will lead to items spending more time in their inventory locations, thus leading to a greater work-in-process. If on the other hand, the feeder line has 2 kanbans to begin with, “intrinsicly” it already has a greater inventory than a feeder line with 1 kanban. If on such a line (a line with 2 kanbans at each of the stages), the number of product types is increased, the work-in-process will increase in the same way it did on a feeder line having 1 kanban. But the “total” work-in-process will be larger for the feeder line with 2 kanbans due to the fact that it started out with more “intrinsic” inventory than feeder line with 1 kanban. Thus, the level of the number of kanbans and the level of the number of product types being assembled has an impact on work-in-process.

The ANOVA table in Appendix 3 indicates that there is another interaction effect that is significant: interaction between the number of stages on the feeder line and the number of product types being assembled. To analyze this interaction, Figure 6-15 depicts a plot of



the average work-in-process under different levels of number of stages and number of product types.

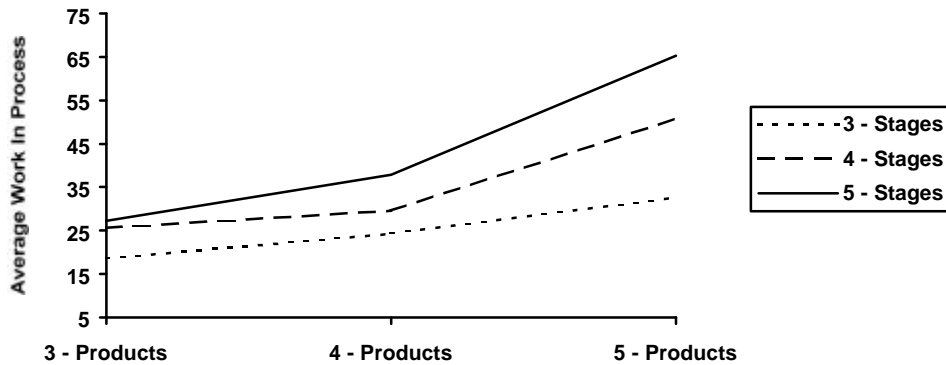


Figure 6-15 : Interaction between number of stages and number of product types

Increasing the number of products being assembled decreases the frequency with which orders are placed for a particular product type. Due to the decrease in frequency, items spend more time in the inventory storage locations leading to an increase in work-in-process. When the number of stages on the feeder line is increased, the number of inventory storage locations is also increased and thus work-in-process increases. However, the increase in work-in-process is dependent on both the number of stages and the number of product types being assembled. For example, when the system has 3 products and 4 stages, increasing the number of stages on the feeder line to 5 has no significant increase in the work-in-process. But when the system has 4 product types and 4 stages, increasing the number of stages to 5 has a significant increase in work-in-process. This behavior may be due to the fact that when the system has 3 products, the frequency of orders is so high that even upon adding a new stage, the inventory locations are still very sparsely populated. However under the system which has 4 products, the frequency of orders may not be that high and adding a new stage will only cause more work-in-process in the system.

### **6.3 Stockout**

Another performance measure that is studied in this research is the stockout. Stockout in this research is defined as the instance when a demand is placed on the final assembly line and there are no input materials to immediately start processing the demand. A stockout situation at a workstation may cause delays in processing the demand which may in turn lead to stoppage of the conveyer system. Stockout may also cause a loss in throughput thus making the production system unable to meet the master production schedule specifications.

Generally in JIT systems, stockout situations are handled by issuing an *emergency kanban*. The JIT philosophy mandates that all kinds of wastes be eliminated and a high quality be maintained. So if there are stockouts, then it is an indication that the system is not operating under ideal conditions. So, an emergency kanban is issued and the production is halted to analyze and resolve the cause for the stockout. To keep the analysis simple, this research does not issue emergency kanbans nor does it analyze the cause for the stockout. This research just focuses on the key design and operating factors that affect the number of stockouts per shift.

The average number of stockouts per shift, observed during the various experiments performed as part of this research, are shown in Appendix 1. An ANOVA was done on this data to identify the significant factors and the results of the ANOVA are shown in Appendix 4. As the ANOVA table in Appendix 4 indicates, the number of kanbans and the number of products are the only significant main effects at a 95% significance level. The ANOVA table also indicates that the interaction between the number of kanbans and the number of product types is also significant at the 95% significance level.

To understand the impact of the number of kanbans on the number of stockouts observed during a shift, Figure 6-16 was plotted and shows the number of stockouts per shift versus the number of kanbans. The figure shows that when the number of kanbans is increased

from a level of 1 to a level of 2, the number of stockouts reduces drastically. As mentioned previously, the number of kanbans at a stage is equal to the number of inventory items that can be present in the associated inventory storage location (production ordering kanbans indicate the number of inventory items in the output storage location at the stage, while withdrawal kanbans indicate the number of inventory items in the input storage location at the stage). When the feeder line has only 1 kanban of each product type at each of the stages, the above graph indicates that it is not a sufficient number and there is a high probability that when an order is placed for a particular product type, it will cause a stockout situation. But as soon as the number of kanbans of each product type is increased to 2, there are almost no stockouts. However, since the inventory maintained at the stages under a 2 kanban feeder line is sufficient to avoid stockouts, there is no significant reduction in stockouts by increasing the number of kanbans to 3.

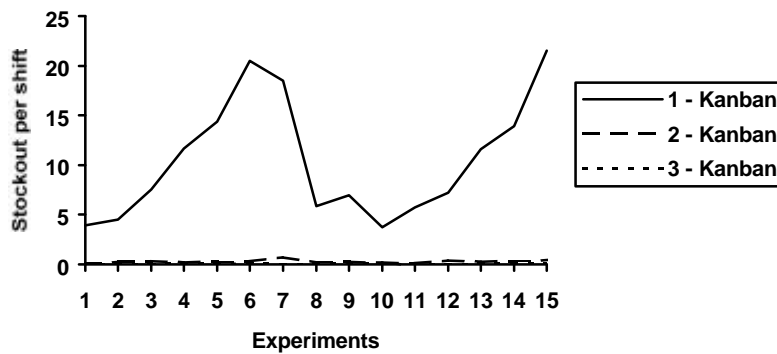


Figure 6-16 : Relationship between stockout and number of kanbans

Thus one may infer from this graph that a feeder line operating under low levels of kanbans is prone to stockouts and the stockouts decrease as the number of kanbans are increased. But once the number of kanbans reaches a “threshold” level where stockouts are minimized, further increase in the number of kanbans has no effect on stockouts.

The ANOVA table in Appendix 4 indicates that the number of product types also has a significant effect on the number of stockouts. Figure 6-17 depicts the variation of stockouts with the number of product types being assembled on the feeder line as

observed during the various experiments conducted in this research. It can be observed from Figure 6-17, that under all experimental conditions, with increase in the number of product types, there is a decrease in the number of stockouts observed per shift. When the number of product types being assembled on the final assembly line is increased, the frequency of demand for a product of a particular kind decreases. For example, if there are three products being assembled on the final assembly line, then a sequence of orders placed on the line may be A B C A, resulting in 2 intervening product types between successive placement of orders for the product type A. If there are 4 product types, then a sequence of orders placed on the line may be A B C D A, resulting in 3 intervening product types between successive placement of orders for product type A.

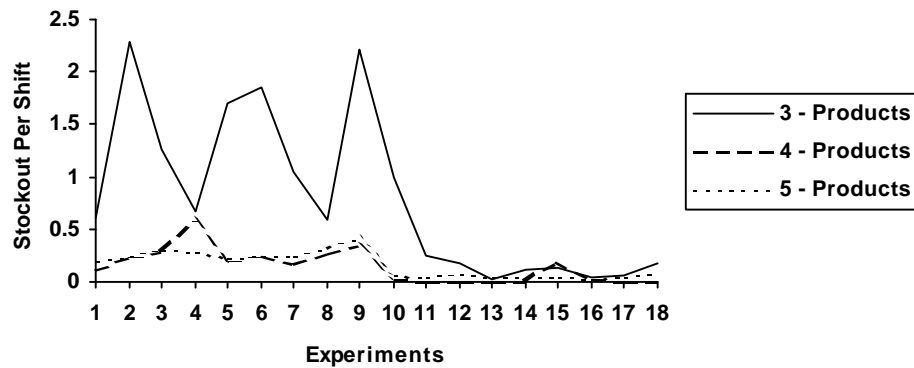


Figure 6-17 : Relationship between stockout and number of product types

This decrease in the frequency of demand for a particular product type or increase in the time between placement of orders for a particular product type, enables the feeder line to replenish items that have been consumed by previous orders by the time another order for the same product type is placed. This explains the behavior which the figure depicts, namely with increase in the number of product types being assembled, the number of stockouts per shift is reduced. However, similar to the case with the number of kanbans, there is a “threshold” number of product types which minimizes the number of stockouts observed. Further increases in the number of product types beyond this “threshold” value does not change the stockout pattern.

The ANOVA table in Appendix 4 indicates that the number of stages and the level of processing time variability are not statistically significant at a 95% significance level. However one must note that both these factors add a degree of “variability” to the feeder line which may affect the availability of items; thereby leading to stockouts. For example, the level of processing time variability could cause large processing times at the stages on the feeder line which in turn may delay the availability of items for assembly on the final assembly line. In a similar vein, an increase in the number of stages on the feeder line could also increase the delay in the availability of items. This research does not provide sufficient proof to conclusively establish the impact of these two factors on the number of stockouts observed during a shift.

The ANOVA table in Appendix 4 indicates that the interaction between the number of kanbans and the number of product types is the only interaction that is statistically significant at a 95% significance level. To study the interaction, Figure 6-18 depicts the average stockout observed under different combinations of number of kanbans and number of product types being assembled. This figure indicates that when there is 1 kanban maintained at the stages on the feeder line, with increase in the number of product types being assembled, the stockout decreases. However, when the number of kanbans are increased to 2 at each of the stages, the increase in product types does not lead to a decrease in the stockout of the same magnitude as it did under the 1 kanban situation. The reason for the unequal behavior at the two levels of kanbans with increase in the number of product types is that, when there are 2 kanbans, there is sufficient inventory at the inventory storage locations to avoid stockouts. Under this situation, when the number of product types are increased, it results in a decrease in the frequency for demands for a particular type. But because the kanbans have already avoided the stockouts, the reduction in frequency does not have any further effect on the stockouts. However if the feeder line with 1 kanban is considered, the stockouts are high. Thus, when the number of product

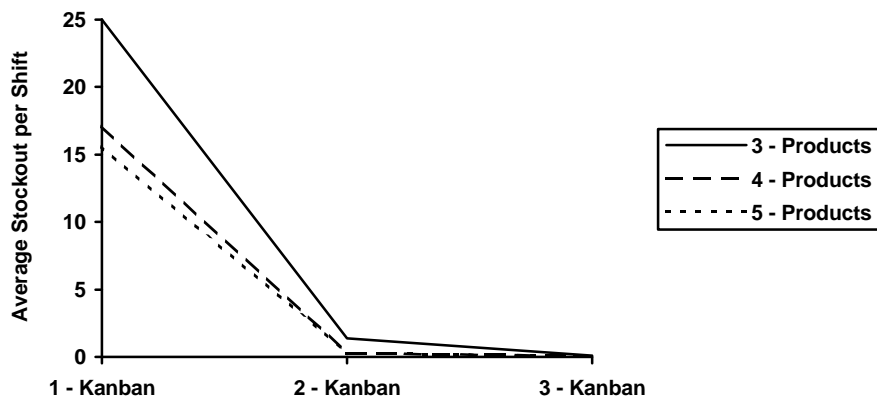


Figure 6-18 : Interaction between number of kanbans and number of product types

types assembled on the feeder line is increased, it leads to a decrease in frequency of demand for a particular product type. This decrease in frequency aids the feeder line in providing it with time to replenish the items that have been consumed, thereby reducing the number of stockouts observed.

#### **6.4 Kanban Waiting Times**

Kanban waiting times are the duration that the kanbans (production ordering and withdrawal kanbans) spend waiting at their kanban posts, (i.e. production ordering kanban post and withdrawal kanban post respectively). The kanban waiting times are an important performance measure because they indicate how quickly the kanbans are serviced. Production ordering kanbans are posted to the production ordering kanban post when an item is withdrawn from the output storage location of that stage. If the production ordering kanban spends too much time waiting for service at a stage, then there is a chance that when an order is placed at that stage by the final assembly line or by another subsequent stage, a stockout situation may arise. Withdrawal kanbans, on the other hand, are posted to the withdrawal kanban post at a stage when an input item is withdrawn from the input storage location to produce an output item. The withdrawal kanban then tries to withdraw an item from the preceding stage's output storage location. If the withdrawal

kanban spends too much time in withdrawing an item from the preceding stage's output storage location, then stoppage of work may result at the stage that posted the withdrawal kanban which in turn can lead to stockout situations at subsequent stages on the feeder line.

The time that the withdrawal kanbans and production ordering kanbans spend waiting for service is collectively called *kanban waiting time*. No distinction is made between the two waiting times in this research for the sake of simplicity of analysis and also because the purpose of this research is to characterize the general nature of kanban waiting times under varying operational and design parameters of the feeder line. The values obtained for kanban waiting times under different experimental conditions are in Appendix 1. An ANOVA was performed on these data and the results of the ANOVA are given in Appendix 5. The ANOVA table indicates that the level of processing time variability, the number of stages on the feeder line, and the number of product types are all statistically significant factors at a 95% significance level. The ANOVA also indicates that the interactions between the number of stages and the number of product types, the number of kanbans and the number of stages, the number of products and the level of variability in processing times, and the number of stages and the level of variability in processing times are all statistically significant at the 95% significance level. The following paragraphs analyze each of the significant factors and attempts to explain why the relationships exist.

Figure 6-19 depicts the variation of kanban waiting times with the level of processing time variability. This figure indicates that with increase in the variability of processing times, the kanban waiting time increases. This follows directly from the fact that with increasing variability in processing times, chances are high that longer processing times will be encountered at the stages. Longer processing time at the stages will in turn cause the production ordering kanbans to wait longer to trigger production at the stage and withdrawal kanbans to wait longer for an item to be present in the output storage location.

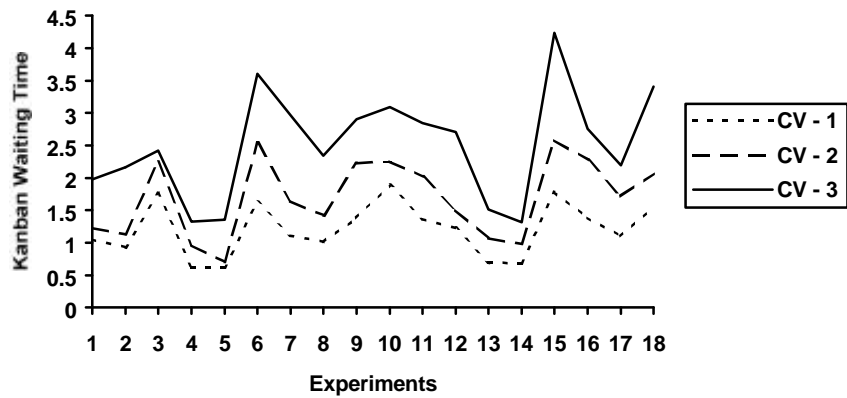


Figure 6-19 : Relationship between kanban waiting time and processing time variability

Another significant factor that affects the kanban waiting time is the number of product types being assembled. Figure 6-20 depicts a plot of the variation of kanban waiting time with the number of product types being assembled. The figure indicates that the kanban waiting time decreases with increase in the number of product types being assembled on the feeder line. The decrease in the kanban waiting time may be attributed to the decrease

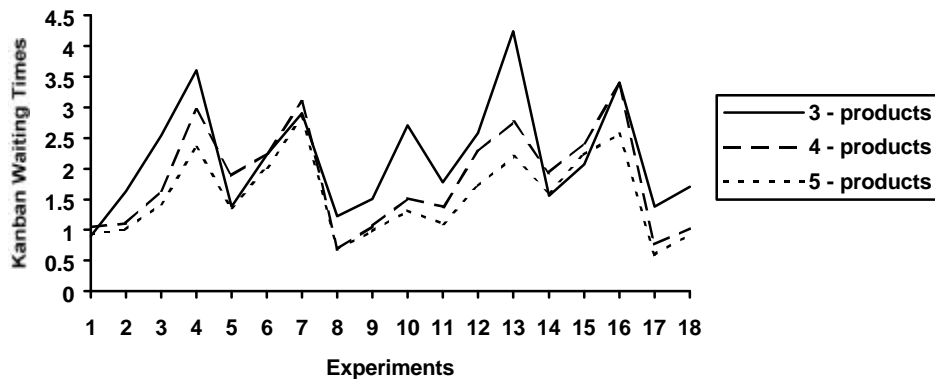


Figure 6-20 : Relationship between kanban waiting times and number of product types

increasing number of product types being assembled on the line. The decrease in the frequency of demand for a particular product type will enable the feeder line to replenish



items that have been withdrawn by previous orders, before the next order for the same product type is placed. This has the effect that when a withdrawal kanban for a particular product type is placed in the withdrawal kanban post of a particular stage, it is able to find an item in the output storage area of the preceding stage, thereby reducing the waiting time of the withdrawal kanban. Since the withdrawal kanbans are able to retrieve items from preceding stages more quickly, the production ordering kanbans will find raw-materials more quickly, thereby reducing their waiting time. Thus increasing the number of product types being assembled has the overall effect of reducing the kanban waiting times.

The ANOVA table in Appendix 5 indicates that the number of kanbans has a significant effect on the waiting time if Type-I SS is used as a criterion. However, Type-III SS suggests that the number of kanbans does not have a significant effect. The lack of concurrence in Type-I SS and Type-III SS is an indication that the effect of the number of kanbans is confounded with some other effect and thus no inferences can be drawn regarding the significance of the number of kanbans. Nonetheless, from a conceptual point of view, it is interesting to understand the relationship between kanban waiting times and the number of kanbans.

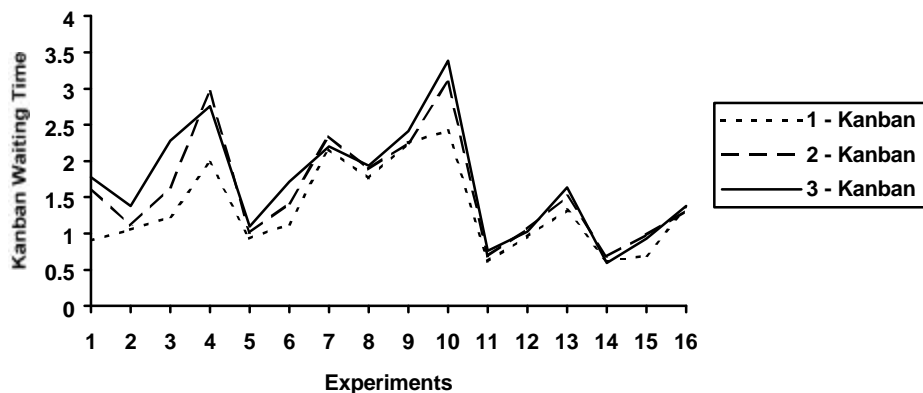


Figure 6-21 : Relationship between kanban waiting time and number of kanbans

The above figure depicts the variation of kanban waiting time with the number of kanbans maintained. When the number of kanbans is increased, there are two conflicting things that

happen. On one hand, there is an increase in the queue length associated with the kanban posts, while on the other hand the availability of raw-material is increased. With an increase in the availability of raw-material, one would expect that the withdrawal kanban waiting time decreases, thereby causing a decrease in the production ordering kanban waiting time. However, Figure 6-21 indicates that there is a general trend towards greater kanban waiting time with the increase in number of kanbans. This trend may be attributed to the increase in the queue length associated with the kanban posts when the number of kanbans are increased. In this research, the kanbans are serviced on a first-in-first-out (FIFO) basis. Due to this policy, it may so happen that the kanban to be serviced next does not have a raw-material in the input storage area. There may be other kanbans in the post which have raw-materials in the input storage area. But service on those kanbans cannot start until the service on the first kanban in the kanban post has started. Thus, inspite of having raw-materials, certain kanbans are forced to wait until service on kanbans ahead of them in the queue has begun.

The number of stages on the feeder line is also a significant effect at a 95% significance level. Figure 6-22 depicts the variation of the kanban waiting times with the number of stages on the feeder line.

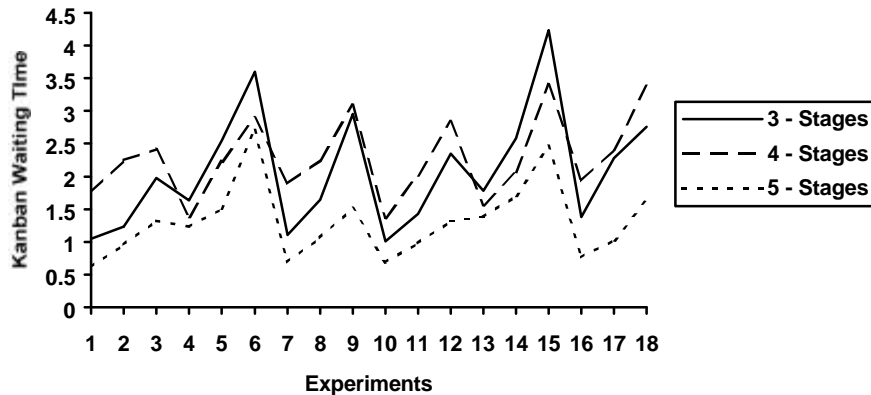


Figure 6-22 : Relationship between kanban waiting time and number of stages

The above figure indicates that with increase in the number of stages on the line, the kanban waiting time decreases. An increase in the number of stages leads to an increase in the number of inventory locations on the feeder line. The larger number of inventory locations makes the raw-materials available more quickly. Due to the quick availability of raw-materials, the withdrawal kanbans are serviced quicker which in turn results in the production ordering kanbans being serviced quicker. Thus an increase in the number of stages has the effect of decreasing the kanban waiting time due to increased inventory locations.

The ANOVA table in Appendix 5 indicates that the following 2-way interactions are significant at a 95% significance level: interaction between number of kanbans and number of stages, interaction between number of stages and number of product types being assembled, interaction between number of stages and level of processing time variability, and interaction between number of product types and the level of processing time variability. To analyze the interactions, it is helpful to plot the variation of the average kanban waiting times versus the concerned factors.

Figure 6-23 shows the interaction between the number of kanbans and the number of stages on the feeder line. This figure indicates that with an increase in the number of kanbans, the kanban waiting time increases. However, keeping the number of kanbans constant, an increase in the number of stages leads to a decrease in the kanban waiting time. The non-parallelism of the lines in the figure, suggests that the decrease in kanban waiting time due to increase in the number of stages, depends on the number of kanbans. At low levels of kanbans, the kanban waiting times is small due to the smaller queue lengths in the kanban posts. Increasing the number of stages at low levels of kanbans has the effect of increasing the inventory locations on the feeder line, thereby making the raw-materials available more quickly. These two factors, raw-material availability and smaller queue sizes, lower kanban waiting time under these situations. If the number of kanbans on a feeder line with large number of stages is increased, the larger queue lengths at the

kanban posts dominates over the raw-material availability and causes an increase in the kanban waiting time.

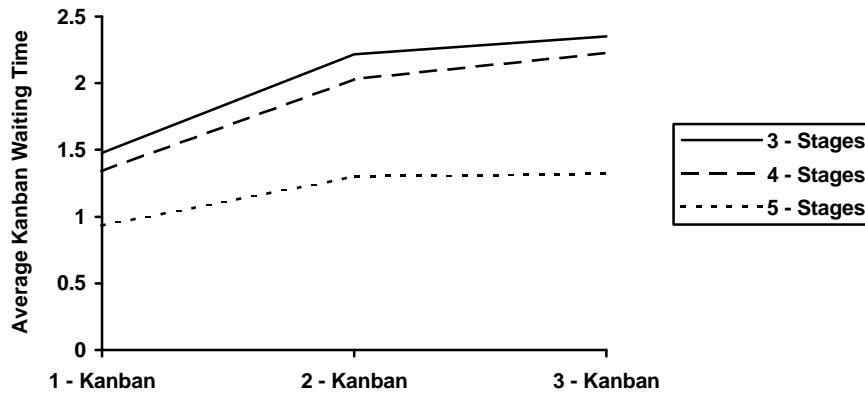


Figure 6-23 : Interaction between number of kanbans and number of stages

Another interaction that is significant is the interaction between the number of stages on the feeder line and the number of product types being assembled. Figure 6-24 depicts the variation of kanban waiting time with the number of stages and the number of product types. As the figure indicates, with the increase in number of stages, keeping the number of product types a constant, the kanban waiting time decreases. This observation is consistent with the main effect of the number of stages which is to decrease the kanban waiting time through increased availability of raw-materials. The figure also indicates that, with the increase in number of product types, keeping the number of stages a constant, the kanban waiting time decreases. This behavior is also consistent with the main effect of the number of product types which is, to decrease the kanban waiting time through decrease in the frequency of orders for a particular product type. However, the non-parallelism of the graph indicates that there is an interaction between the number of product types and the number of stages on the line. It appears from this graph that at higher levels of the number of product types, an increase in the number of stages leads to a larger decrease in the kanban waiting time as compared to feeder lines having lower number of product types. This behavior may be due to the combined effect of increased availability of raw-materials (due to an increase in the number of stages) and due to decreased frequency of

orders for the same product type because of the high number of product types being assembled.

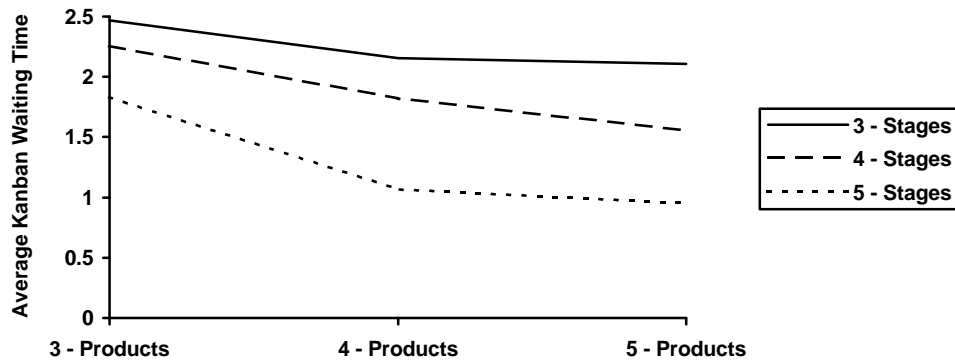


Figure 6-24 : Interaction between number of product types and number of stages

The ANOVA table in Appendix 5 indicates that there is also a significant interaction between the number of stages on the feeder line and the processing time variability. Figure 6-25 is a plot showing the average kanban waiting time at different levels of the number of stages on the feeder line and the level of processing time variability. This figure shows that, with higher processing time variability while keeping the number of stages a constant, leads to larger kanban waiting time. The figure also shows that with increase in the

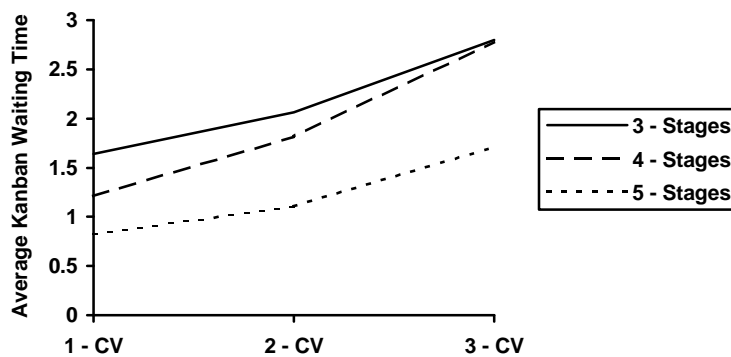


Figure 6-25 : Interaction between processing time variability and number of stages

number of stages while keeping the level of processing time variability a constant, leads to a decrease in the kanban waiting times. However, the magnitude by which the kanban waiting time decreases or increases is dependent on the levels of these two factors.

Figure 6-26 depicts the interaction between the number of product types and the level of processing time variability. This figure indicates that with the increase in the level of

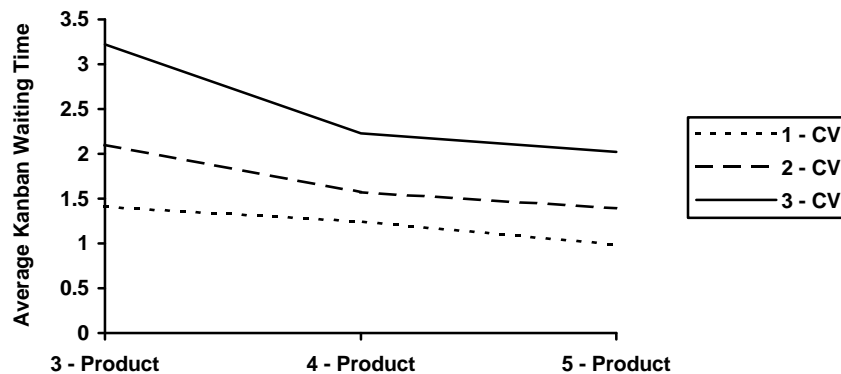


Figure 6-26 : Interaction between number of product types and level of processing time variability

variability in processing time, keeping the number of product types a constant, the kanban waiting time increases. With increasing variability in processing time, the chances of having a large processing time increases. Large processing time in turn make the withdrawal kanbans wait longer at the posts. Longer withdrawal kanban waiting times leads to longer waiting time for production ordering kanbans. Figure 6-26 also indicates that, as the number of product types are increased, keeping the level of variability a constant, the kanban waiting time decreases. Increasing the number of product types has the effect of decreasing the frequency between orders for the same product type. The decreased frequency enables the feeder line to replenish items withdrawn. Thus when a withdrawal kanban is posted at the withdrawal kanban post, the kanban is serviced immediately because of availability of materials. One must however note that the decrease in kanban waiting time by increasing the number of product types being assembled is

dependent on the level of variability of processing times. If the level of variability is high, then inspite of having a large number of product types, the kanban waiting time will still be relatively high due to the high chance of large processing times.

#### **6.4 Utilization and Throughput**

Utilization measures the extent to which the production resources are utilized. In the context of this research, the production resources were the various stages on the feeder line. Throughput is also a very important performance measure of the manufacturing system, because it gives an indication of how well the production schedule is being met. In many cases, production schedules are contractual agreements and there is a high financial risk associated with not meeting the production schedule. This research used a single quantity, equal to the average of the utilizations of the individual stages, as the representative utilization of the feeder line. Throughput in the context of this research is defined as the number of products finished at the first workstation on the final assembly line during a shift of eight hours.

This research attempted to study the behavior of utilization of the feeder line and throughput, under varying operational and design conditions. However, as Appendix 1 indicates, due to the inherent nature of the system being modeled (the master production schedule used, processing times used, etc.) the utilizations obtained were above 90% and throughput obtained was close to 60 for all the experiments, thus making it hard to analyze the impact of the operational and design parameters on utilization and throughput.

The ANOVA study for utilization (shown in Appendix 6) yielded a sum of squares total (SST) of about 0.0081 suggesting that utilization is more or less a constant in the series of experiments performed as part of this research. Although the ANOVA table indicates that the number of stages on the feeder line and the number of product types being assembled, are statistically significant at a 95% significance level, due to the low SST it is dangerous to draw any conclusions about the significance of these factors in explaining utilization.

The ANOVA study for throughput (shown in Appendix 7) yielded a mean of 60 and an SST of 11.74 (there were 15 experiments where throughput was 59 and all the remaining experiments had throughput equal to 60). In addition, the ANOVA yielded a R-square value of 0.61. Thus this research did not yield significant data regarding utilization and throughput of the feeder line, thereby precluding a study of their behavior under differing operational and design conditions. As noted in Chapter 7, further research needs to be done to explore the relationship of utilization and throughput with the factors analyzed in this research.



## Chapter 7

### Conclusions and Recommendations

#### 7.1 Summary And Conclusions

Just In Time (JIT) production systems are finding wide acceptance among companies as a way of remaining competitive. At the core of any JIT production system lies the kanban controlled inventory system and it was the aim of this research to study the characteristics of these inventory systems. The success of kanban controlled JIT systems depends on both the design and operational factors associated with the system. The design factors that were studied in this research include the number of product types that are produced on the line, the number of stages on the production line and the number of kanbans that are maintained between stages. While a line may have many operational factors like processing time variability, quality fallout, etc., the only operational factor that was studied in this research was the processing time variability.

As there is no single metric that captures in entirety the different aspects of measuring the performance of a kanban controlled production line, this research used the following set of metrics to measure the performance:

1. Time in system, defined as the time between the placement of an order and its completion at a station on the production line.
2. Work in process, defined as all the material resources which are in an intermediate condition, pending to be converted to the final product.
3. Stockout, defined as the number of times an order is placed with no inventory item present to satisfy the request.
4. Kanban waiting times, defined as the time that the production-ordering and withdrawal kanbans spend waiting for start of service.

5. Utilization of the line, defined as the average of the utilization of all the stages on the production line.
6. Throughput, defined as the number of items assembled on the assembly line during a shift.

The research studied the relationships between the performance measures listed above and the design and operational factors described earlier in this section. A simulation model was developed to investigate these relationships. The model that was simulated is shown in figure 5-1 and essentially consisted of a final assembly line with various workstations. Each of the workstations was in turn fed by a kanban controlled feeder line. The simulation model modeled the first workstation on the final assembly line and the associated kanban controlled feeder line.

The simulation model was programmed using SLAM II and FORTRAN. Necessary data for the simulation were obtained through pilot runs and using the literature survey as a guide for the distributions. Although the computer program was written to simulate an assembly line and the feeder line associated with one of the workstations, it can very easily be adapted to other situations. The program is very flexible towards changes in key parameters associated with the line, for example, processing times, number of stages on the line, etc.

The computer program provides the following output:

1. Average time in system of the products being assembled at the workstation on the final assembly line.
2. Average work in process on the feeder line.
3. The number of stockouts at the workstation on the final assembly line during a shift.

4. Average waiting times of the production ordering and withdrawal kanbans on the feeder line.
5. Throughput at the workstation on the final assembly line during a shift.
6. Average utilization of all the stages on the feeder line.

A  $3^4$  factorial experiment was designed and the simulation model was run once for each of the cells of the design. Due to the resource intensive nature of the simulations, no replications were made. Detailed statistical analysis were then performed to analyze the relationships between the performance measures and the experimental factors.

Time in system was observed to be affected by the number of kanbans maintained at the stages, the number of product types being produced on the feeder line, and the level of variability at the stages on the feeder line. The analysis also indicated that the interaction effects due to number of kanbans and number of stages, number of kanbans and number of products, number of kanbans and level of variability, and the number of product types and level of processing time variability are significant. With the increase in number of kanbans, the time in system approached the cycle time. There was also an indication that once the number of kanbans has been increased such that the time in system is close to the cycle time, further increases in the number of kanbans beyond this “threshold” value do not affect the time in system. Time in system was also observed to decrease with increases in the number of product types being assembled and similar to the case with the number of kanbans, there was a “threshold” value for the number of product types beyond which time in system remained largely unaffected with further increases in the number of product types being assembled. As expected, the time in system increased with increasing the level of variability in processing times.

The analysis of work-in-process (WIP) indicated that, the main effects of the number of kanbans, number of stages on the feeder line, number of product types being assembled, and the level of variability in processing times are statistically significant. The following

two-way interactions were also found to be significant: the interaction between number of kanbans and number of stages on the feeder line, interaction between number of kanbans and number of products being assembled, and the interaction between number of stages and number of product types. It was observed that work-in-process increased with the increase in the number of kanbans. An increase in the number of stages on the feeder line led to an increase in the work-in-process. It was also observed that work-in-process increased with an increase in the number of product types being produced on the feeder line.

When the number of stockouts observed during a shift was analyzed it was concluded that the number of kanbans and the number of products were the only significant main effects. The interaction between the number of kanbans and the number of product types was also significant. It was observed that with an increase in the number of kanbans, the number of stockouts per shift decreases. The behavior of stockouts with the number of product types was similar to the behavior with the number of kanbans and the stockouts decreased with an increase in the number of product types being assembled.

The analysis of kanban waiting times indicated that the level of processing time variability, the number of stages on the feeder line, the number of product types, and the number of kanbans were all statistically significant factors. The analysis also indicated that the interactions between the number of stages and the number of product types, the number of kanbans and the number of stages, the number of products and the level of variability in processing times, and the number of stages and the level of variability in processing times were also statistically significant. It was observed that with an increase in the processing time variability, the kanban waiting times also increased. Similarly, the kanban waiting times were found to increase with an increase in the number of kanbans and number of stages on the feeder line. However, kanban waiting time was observed to decrease with an increase in the number of product types being assembled.

Due to the inherent nature of the system being modeled, the master production schedule used, processing times used, etc., the values for utilization obtained were above 90% and the values for throughput obtained were close to 60 for all the experiments, thus making it hard to analyze the impact of the operational and design parameters on utilization and throughput. So, no conclusions were drawn regarding how the design and operational factors affect throughput and utilization.

## **7.2 Recommendations For Future Research**

The following recommendations are made for future research to further refine and extend the capabilities of the methodology presented in this research:

1. This research assumed an equal number of production ordering and withdrawal kanbans at all the stages. It will be interesting to analyze the impact of differing the number of production ordering and withdrawal kanbans at the stages. Another related issue is to maintain different number of production ordering and withdrawal kanbans at the same stage.
2. This research assumed a first come first served (FCFS) policy at the production ordering and withdrawal kanban posts. This had the impact that, for some kanbans in the post, service could not be started on them inspite of raw-materials existing in the storage locations, until service started on the first kanban in the queue. It'll be interesting to analyze how the behavior of the feeder line will be impacted if the FCFS policy is changed. Future research should aim at allowing preempting waiting kanbans and also evaluate other scheduling policies like SPT, etc.
3. Lot sizes of one unit were assumed at all the stages in this research. While this simplified the research, there are a lot of situations in which unit lot sizes are uncommon. Further research should be done to analyze how different lot sizes at different stages will affect the performance measures.

4. This research was not able to establish the behavior of the utilization and throughput of the feeder line. New experiments need to be designed to analyze their behavior.
5. The behavior of certain performance measures are conflicting in nature. For example, the time in system approaches the cycle time with an increase in the number of kanbans. But the work in process increases with an increase in the number of kanabans. For optimal performance on the feeder line, time in system must approach the cycle time and work in process must be minimized. But due to conflicting nature of their behavior, some kind of a goal based metric needs to be designed, which will integrate all the performance measures analyzed in this experiment and provide a single yardstick to measure the performance.
6. The model that was simulated in this research deviated from reality when it ignored scheduled maintenance activities on the feeder line, worker productivity, sick time, etc. These and other real-life elements need to be incorporated into the model. As mentioned previously, the model developed in this research is flexible and adaptable to future changes.
7. Certain combinations of experimental factors resulted in infeasible conditions for the system being modeled and as a result steady state metrics for the performance measures were not reached. Due to the lack of complete data, certain effects were confounded with others. Further investigation needs to be done in order to determine the nature of these confounded relationships.
8. A fractional factorial design or central composite design of experiments can be used instead of the  $3^4$  design used in this research. This would enable replications of trials to be made economically, thereby making the experimentation statistically more robust.
9. Setup times for producing different product types were not explicitly considered in this research. Further research needs to be done to determine how the performance measures would be impacted with non-zero setup times.

## References

- Abernathy, Clark and Kanthrow (1981), "The new industrial strategy", Harvard Business Review, Vol. 39, Pg. 68-81
- Abdulnour, Dudek and Smith (1995), "Maintenance Policies in JIT production systems", International Journal of Production Research, Vol. 33, No. 2, Pg. 565-583
- Aggarwal, S.C., 1985, "MRP, JIT, OPT, FMS ? Making sense of production operations system", Harvard Business Review, Vol. 63, Pg. 8-16
- Ahmed, Tunc and Montagno (1991), "A comparative study of US manufacturing firms at various stages in JIT implementation", International Journal of Production Research, Vol. 29, No. 4, Pg. 787-802
- Albino, Carella and Okogbaa (1992), " Maintenance policies in JIT manufacturing lines", International Journal of Production Research, Vol. 30, No. 2, Pg. 369-382
- Ansari and Modarress (1988), "JIT purchasing as a quality and productivity center", International Journal of Production Research, Vol. 26, No. 1, Pg. 19-26
- Bard and Golany (1991), "Determining the number of Kanbans in a multiproduct multistage production system", International Journal of Production Research, Vol. 29, No. 5, Pg. 881-895
- Berkeley, B.J. (1991), "Tandem queues and kanban controlled lines", International Journal of Production Research, Vol. 29, No. 10, Pg. 2057-2081
- Berkeley (1994), "Testing minimum performance levels for kanban controlled lines", International Journal of Production Research, Vol. 32, No. 1, Pg. 93-109
- Berkeley (1993a), "Setting minimum performance levels for two-card kanban controlled lines", International Journal of Production Research, Vol. 31, No. 5, Pg. 1003-1021
- Berkeley (1993b), "Effect of buffer capacity and sequencing rules on single card kanban system performance", International Journal of Production Research, Vol. 31, No. 12, Pg. 2875-2893
- Berkeley and Kiran (1991), "A simulation study of sequencing rules in a kanban controlled flow shop", Decision Sciences, Vol. 22, No. 3, Pg. 559-582
- Bielecki and Kumar (1988), "Optimality of zero-inventory policies for unreliable manufacturing systems", Operations Research, Vol. 36, No. 4, Pg. 532-541
- Bitran, G. and Chang, L., (1987), "A mathematical programming approach to a deterministic Kanban system", Management Science, Vol. 33, No. 4, Pg. 427-441

- Burnham, J.M. (1985), "Improving manufacturing performance: a need for integration", Production and Inventory Management, Vol. 26, Pg. 51-58
- Chang and Yih (1994), "Generic kanban systems for dynamic environments", International Journal of Production Research, Vol. 32, No. 4, Pg. 889-902
- Changchit and Terrel (1988), "Issues in Just-In-Time production system", International Journal of Production Research, Vol. 26, No. 2, Pg. 360-364
- Chao, Das and Nagendra (1995), International Journal of Production Research, Vol. 33, No. 2, Pg. 377-389
- Chapman, S.N. (1989), "Just-In-Time supplier inventory: an empirical implementation model", International Journal of Production Research, Vol. 27, No. 12, Pg. 1993-2007
- Chao and Shih (1992), "Simulation studies in Just In Time production", International Journal of Production Research, Vol. 30, No. 11, Pg. 2573-2586
- Chaudhury and Whinston (1990), "Towards an adaptive kanban system", International Journal of Production Research, Vol. 28, No. 3, Pg. 437-458
- Cooper, C. (1984), "The Japanese connection: imitate or emulate", Production & Inventory Management, Vol. 25, Pg. 114-125
- Crawford, K.M., Blackstone, J.H., and Cox, J.F. (1988), "A study of JIT implementation and operating problems", International Journal of Production Research, Vol. 26, No. I, Pg 1561-1568
- Crawford and Cox (1990), "Designing performance measurements for JIT operations", International Journal of Production Research, Vol. 28, No. 11, Pg. 2025-2036
- Davis and Stubitz, 1987, "Configuring a kanban system using a discrete optimization of multiple stochastic responses", International Journal of Production Research, Vol. 25, No. 5, Pg. 721-740
- Deleersnyder, Hodgson, Muller, O'Grady, 1989, "Kanban controlled pull systems: an analytic approach", Management Science, Vol. 35, No. 9, Pg. 1079-1091
- Ebrahimpour and Schonberger (1984), "The Japanese Just-In-Time/Total Quality Control production system: potential for developing countries", International Journal of Production Research, Vol. 22, No. 3, Pg. 421-430
- Finch and Cox (1986), "An examination of JIT management for the small manufacturer with an illustration", International Journal of Production Research, Vol. 24, No. 2, Pg. 329-342



Flapper, Miltengerg and Wigngaard (1991), "Embedding JIT into MRP", International Journal of Production Research, Vol. 29, No. 2, Pg. 329-341

Funk (1989), "A comparison of inventory cost reduction strategies in a JIT manufacturing system", International Journal of Production Research, Vol. 27, Pg. 1065-1080

Gelders and Wassenhove, 1984, "Capacity planning in MRP, JIT and OPT: a critique", Engineering cost and production economics, Vol. 9, Pg. 201

Gilberth, J.P. (1990), "The state of JIT implementation and development in the USA", International Journal of Production Research, Vol. 28, No. 6, Pg. 1099-1109

Golhar and Stamm (1991), "The Just-In-Time philosophy: a literature review", International Journal of Production Research, Vol. 29, No. 4, Pg. 657-676

Graham, I (1992), "Comparing trigger and kanban control of flow-line manufacture", International Journal of Production Research, Vol. 30, No. 10, Pg. 2351-2362

Gravel and Price (1988), "Using the kanban in a job shop environment", International Journal of Production Research, Vol. 26, No. 6, Pg. 1105-1118

Groeflein, Luss, Rosenwein and Wahls, 1989, "Final Assembly Sequencing for Just-In-Time Manufacturing", International Journal of Production Research, Vol. 27, No. 2, Pg. 199-213

Grunwald, H., Striekwold, P.E.T., Weeda, P.J., 1989, "A framework for quantitative comparison of production control concepts", International Journal of Production Research, Vol. 27, No. 2, Pg. 281-292

Gupta and Gupta (1989), "A system dynamics model for a multi-stage multi-line dual-card JIT-Kanban system", International Journal of Production Research, Vol. 27, No. 2, Pg. 309-352

Haynsworth, H.C., 1984, "A Theoretical Justification for the use of Just-In-Time scheduling", Production and Inventory Management, Vol. 25, No. 1, Pg. 1-3

Hiller and Boling (1967), "Finite queues in series with exponential or erlang service times - a numerical approach", Operations Research, Vol. 15, No. 2, Pg. 286-303

Hodgson, T.J. and Wang, D. (1991a), "Optimal hybrid push/pull control strategies for a parallel multistage system: part-1", International Journal of Production Research, Vol. 29, No. 6, Pg. 1279-1287

Hodgson, T.J. and Wang, D. (1991b), "Optimal hybrid push/pull control strategies for a parallel

multistage system: part-2", International Journal of Production Research, Vol. 29, No. 7, Pg. 1453-1460

Huang, D.Y., Rees, L.P, and Taylor B.W. (1983), "A simulation analysis of the Japanese just-in-time technique (with kanbans) for a multi-line, multi-stage production system", Decision Sciences, Vol. 14, Pg. 326-343

Jeffrey L. Funk (1989), "A comparison of inventory cost reduction strategies in a JIT manufacturing system", International Journal of Production Research, Vol. 27, No. 7, Pg. 1065-1080

John B. South (1986), "A minimum production lot-size formula for stockless production", Production and Inventory Management, 1986, Vol. 27, No. 2, Pg. 143-150

Karmakar, V. (1989), "Getting control of Just-In-Time", Harvard Business Review, Vol. 67, Pg. 122-131

Kim (1985), "Just-In-Time manufacturing system: a periodic pull system", International Journal of Production Research, Vol. 23, Pg. 553-562

Kimura and Terada (1981), "Design and analysis of pull system, a method of multi stage production control", International Journal of Production Research, Vol. 19, Pg. 241

Krajewski, King, Ritzman & Wong, 1987, "Kanban, MRP and shaping the manufacturing environment", Management Science, Vol. 33, No. 1, Pg. 39-57

Krishnamurty, Swenseth and Wilson (1992), "Describing processing time when simulating JIT environments", International Journal of Production Research, Vol. 30, No. 1, Pg. 1-11

Landers and Mohammed, 1989, "Repetitive mixed-model lot processing", Production and Inventory Management, Vol. 30, No. 3, Pg. 9-13

Law and Carson (1991), Simulation modeling and analysis, Second Edition

Lee, L.C., 1987, "Parametric appraisal of the JIT system", International Journal of Production Research, Vol. 25, No. 10, Pg. 1415-1429

Li and Co, H.C. (1991), "A dynamic programming model for the Kanban assignment problem in a multistage multi-period production system", International Journal of Production Research, Vol. 29, No. 1, Pg. 1-16

Miltenberg, J. (1989), "Level schedules for mixed-model assembly lines in just-in-time production system", Management Science, Vol. 35, No.1, Pg. 192-207

Miltenberg and Wigngaard (1991), "Designing and phasing in Just-in-time production system", International Journal of Production Research, Vol. 29, No. 1, Pg. 115-131

Miyazaki, Ohta and Nishiyama (1988), "The optimal operation planning of kanban to minimize the total operation cost", International Journal of Production Research, Vol. 26, No. 10, Pg. 1605-1611

Monden (1993), "Toyota Production System, Second Edition", Industrial Engineering and Management Press, Pg. 235-260

Parnaby, J. (1988), "A systems approach to the implementation of JIT methodologies in Lucas Industries", International Journal of Production Research, Vol. 26, No. 3, Pg. 483-492

Philipoom, P.R., Rees, L.P., Taylor, B.W., Huang, P.Y., 1990, "A mathematical programming approach for determining work-center lot-sizes in a Just-In-Time system with signal kanbans", International Journal of Production Research, Vol. 28, No. 1, Pg. 1-15

Philipoom, P.R., Taylor, B.W., Huang, P.Y., and Rees, L.P. (1990), "A mathematical programming approach for determining work-center lot-sizes in a just-in-time system with signal kanbans", International Journal of Production Research, Vol. 28, No. 1, Pg. 1-15

Philipoom, Rees, Taylor and Huang (1987), "An investigation of the factors influencing the number of kanbans required in the implementation of the JIT technique with kanbans", International Journal of Production Research, Vol. 25, No. 3, Pg. 457-472

Plenert, Gerhard and Bert, 1986, "MRP, JIT & OPT: what's best?", Production and Inventory Management, Vol. 27, No. 2, Pg. 22-29

Porteus, E.L., 1985, "Investing in reduced setups in the EOQ model", Management Science, Vol. 31, No. 8, Pg. 998-1010

Price, W.L., Gravel, M., Nsakanda, A.L., and Cantin, F. (1995), "Modeling the performance of a kanban assembly shop", International Journal of Production Research, Vol. 33, No. 4, Pg. 1171-1177

Price, W.L., Gravel, M., Nsakanda, A.L., and Cantin, F. (1992), "A mathematical programming model of a kanban job-shop", International Journal of Production Research, Vol. 26, No. 6, Pg. 1105-1118

Pritsker, A. Alan. B (1995), Introduction to simulation and SLAM II

Rees, Philipoom, Taylor and Huang (1987), "Dynamically adjusting the number of kanbans in a

JIT production system using estimated values of leadtime”, IIE Transactions, Vol. 19, No. 2, Pg. 199-207

Rees, Huang and Taylor (1989), “A comparative analysis of an MRP lot-for-lot system and a kanban system for a multi-stage production operation”, International Journal of Production Research, Vol. 27, No. 8, Pg. 1427-1443

Rice and Yoshikawa, 1982, “Comparison of Kanban and MRP concepts for the control of repetitive manufacturing systems”, Production & Inventory Mangement, Vol. 23, No. 1, Pg. 1-13

Richmond and Blackstone (1988), “Just-In-Time in the plastics processing industry”, International Journal of Production Research, Vol. 26, No. 1, Pg. 27-34

Ritzman, King and Krajewski (1984), "Manufacturing performance - pulling the right levels", Harvard Business Review, Vol. 84, Pg. 143-152

Sarker and Fitzimmons (1989), "The performance of push and pull systems: a simulation and comparative study", International Journal of Production Research, Vol. 27, Pg. 1715-1731

Sarkar and Roy D. Harris (1988), “The effect of imbalance in a just-in-time production systems: a simulation study”, International Journal of Production Research, Vol. 26, No. 1, Pg. 1-18

Schroer et al. (1984), "Microcomputer analysis of 2-card kanban system in Just-In-Time small batch production", Industrial Engineering, Vol. 16, Pg. 54-65

Schroer, Black and Zhang (1985), "Just-In-Time, with kanban, manufacturing system simulation on a microcomputer", Simulation, Vol. 45, No. 2, Pg. 62-70

Siha, S. (1994), “The pull production system: modeling and characteristics”, International Journal of Production Research, Vol. 32, No. 4, Pg. 933-949

So, K.C., and Pinault, S.C. (1988), “Allocating buffer storage in a pull system”, International Journal of Production Research, Vol. 26, No. 12, Pg. 1959-1980

South, J.B. (1986), "A minimum production lot size formula for stockless production", Production and Inventory Management, Vol. 27, Pg. 143-150

Spence, A.M., and Porteus, E.L. (1987), “Setup reduction and increased effective capacity”, Management Science, Vol. 33, No. 10, Pg. 1291-1301

Sujimori et al (1979), "Toyota production system and kanban system. Materialization of just-in-

time and respect for human systems", International Journal of Production Research, Vol. 15, Pg. 553-564

Sumichrast, Russel and Taylor (1992), "A comparative analysis of sequencing procedures for mixed-model assembly lines in a JIT production system", International Journal of Production Research, Vol. 30, No. 1, Pg. 199-214

Tae-Moon Kim, 1985, "Just-In-Time Manufacturing System: a periodic pull system", International Journal of Production Research, Vol. 23, No. 3, Pg. 553-562

Villeda, R, Dudek and Smith (1988), "Increasing the production rate of a just-in-time production system with variable operation times", International Journal of Production Research, Vol. 26, No. 11, Pg. 1749-1768

Wacker, J.G. (1986), "Can holding costs be overstated for just-in-time manufacturing system", Production & Inventory Management, Vol. 27, Pg. 11-14

Wang and Wang (1990), "Determining the number of Kanbans: a step towards non-stock-production", International Journal of Production Research, Vol. 28, No. 11, Pg. 2101-2115

Wildemann, H. (1988), "Just-In-Time production in West Germany", International Journal of Production Research, Vol. 26, Pg. 521-538

Wheelwright, S.C., (1981), "Japan - where operations really are strategic", Harvard Business Review, Vol. 59, Pg. 67-74

## Appendix - 1

The 3<sup>4</sup> factorial experiments and the data collected for each of the experiments are shown in the following tables. The number of kanbans used, number of product types, number of stages on the feeder line and the level of processing time variability is also shown.

Each simulation run performed for each experiment gives the following set of data as output:

1. Average time in system of the products being assembled at the workstation on the final assembly line. This statistic is abbreviated as TYS in the following tables.
2. Average work in process on the feeder line. This statistic is abbreviated as WIP in the following tables.
3. The number of stockouts at the workstation on the final assembly line during a shift. This statistic is abbreviated as STOCK in the following tables.
4. Average waiting times of the production ordering and withdrawal kanbans on the feeder line. This statistic is abbreviated as FKTIME in the following tables.
5. Throughput at the workstation on the final assembly line during a shift. This statistic is abbreviated as TPUT in the following tables.
6. Average utilization of all the stages on the feeder line. This statistic is abbreviated as UTL in the following tables.

The tables below summarize the mean (denoted as MEAN) and standard deviation (STD) for each statistic for the entire duration of the simulation run (1200 shifts). In order to reduce the serial-correlation that exists in simulation runs, consecutive observations were “batched” together. Multiple iterations were made through the data series to determine the appropriate batch size (criterion used was that the serial-correlation be less than or equal to 0.4). For each performance measure, initially a batch size of 1, i.e. each observation

being a batch by itself, was tried to determine if the data series yielded the appropriate serial-correlation. If it did not, the batch size was increased to 2 observations yielding 600 batches. The 600 data elements were then checked for serial correlation and if even this degree of “batching” was insufficient, higher batch sizes of 3, 4, 5, 6, 8, 10, 12, 15, 16, 20, 24 and 30 were tried in sequence until the desired serial correlation coefficient was achieved.

The tables below contain the number of batches (denoted as N) that were required to attain the required serial-correlation coefficient. The tables also contain the serial-correlation coefficient for the data series (denoted as CORR) and standard deviation of the batches (denoted as Batch STD).

**Experiment No: 1**

One Kanban, Three Stages, Three Product Types, First Level of Variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	30.183	5	59.99	24.84	0.9	0.9567
STD	23.09	1.27	1.19	7.88	0.94	0.013
CORR	0.42	0.37	0.09	0.41	0.33	0.033
N	50	400	1200	150	400	1200
Batch STD	8.863	0.658	0.310	4.034	0.45	0.014

**Experiment No: 2**

One Kanban, Three stages, Three product types, second level of variation

No steady state values were obtained

**Experiment No: 3**

One Kanban, Three stages, three product types, third level of variation

No steady state values were obtained

**Experiment No: 4**

Two Kanban, Three stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.63	16	60.00	0.61	1.63	0.96
STD	1.78	3.17	0.77	3.33	2.27	0.01
CORR	0.39	0.39	-0.48	0.256	0.39	0.01
N	1200	300	1200	1200	300	1200
Batch STD	0.817	1.44	0.03	0.401	1.026	0.014

### **Experiment No: 5**

Two Kanban, Three stages, three product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	12.711	15	60	2.29	2.54	0.96
STD	3.83	4.02	0.97	1.43	1.2	0.0127
CORR	0.29	0.41	-0.32	0.41	0.39	0.09
N	60	120	1200	60	100	1200
Batch STD	3.453	1.633	0.185	0.539	1.017	0.014

### **Experiment No: 6**

Two Kanban, Three stages, three product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	11.43	14	60.00	1.26	3.6	0.96
STD	3.82	3.10	0.99	3.68	2.67	0.01
CORR	0.41	0.336	-0.42	0.36	0.36	0.07
N	400	150	1200	400	150	1200
Batch STD	1.8	1.116	0.069	0.79	0.94	0.014

### **Experiment No: 7**

Three Kanban, Three stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.32	28	60	0.03	1.781	0.96
STD	0.44	3.57	0.69	0.32	2.48	0.01
CORR	0.065	0.35	-0.52	0.0625	0.3701	0.033
N	1200	240	1200	1200	240	1200
Batch STD	0.199	1.469	0.046	0.013	0.98	0.014



**Experiment No: 8**

Three Kanban, Three stages, three product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.19	27	60	0.118	2.58	0.96
STD	1.42	3.8	0.76	1.05	2.83	0.01
CORR	0.25	0.35	-0.49	0.18	0.35	0.09
N	1200	240	1200	400	240	1200
Batch STD	0.617	1.711	0.032	0.096	1.24	0.014

**Experiment No: 9**

Three Kanban, Three stages, three product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	10.61	25	60	0.17	4.24	0.96
STD	2.56	4.48	0.92	1.55	3.54	0.013
CORR	0.234	0.39	-0.49	0.21	0.41	0.116
N	1200	200	1200	1200	200	1200
Batch STD	0.889	2.13	0.032	0.079	1.69	0.014

**Experiment No: 10**

One Kanban, Three stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	10.25	9	60	3.91	1.05	0.95
STD	6.29	2.39	1.079	1.72	0.53	0.0136
CORR	0.3127	0.41	-0.33	0.315	0.31	0.03
N	240	300	1200	400	240	1200
Batch STD	2.573	0.93	0.171	0.247	0.53	0.014

**Experiment No: 11**

One Kanban, Three stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	10.88	9	59	4.51	1.23	0.95
STD	5.91	2.25	1.054	2.24	1.08	0.014
CORR	0.33	0.33	-0.31	0.33	0.34	0.058

N	300	240	1200	400	300	1200
Batch STD	2.36	0.71	0.072	3.17	0.502	0.014

### **Experiment No: 12**

One Kanban, Three stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	13.77	8	59	7.56	1.98	0.95
STD	7.92	2.35	1.55	5.36	1.77	0.015
CORR	0.37	0.39	0.24	0.38	0.36	0.085
N	200	400	1200	200	400	1200
Batch STD	4.038	1.104	0.126	2.271	0.812	0.014

### **Experiment No: 13**

Two Kanban, Three stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.09	25	59	0.1175	1.1088	0.951
STD	0.82	2.94	1.24	1.34	1.86	0.01
CORR	0.282	0.33	0.41	0.0715	0.32	0.092
N	1200	400	1200	1200	400	1200
Batch STD	0.347	1.203	0.046	0.044	0.737	0.014

### **Experiment No: 14**

Two Kanban, Three stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.72	24	59	0.22	1.64	0.95
STD	1.29	3.09	1.015	1.88	2.18	0.0125
CORR	0.16	0.314	0.35	0.36	0.34	0.09
N	1200	300	240	1200	300	1200
Batch STD	0.478	1.002	0.074	0.047	0.688	0.014

### **Experiment No: 15**

Two Kanban, Three stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.67	23	59	0.28	2.96	0.95

STD	1.84	4.0083	1.3002	2.0822	2.82	0.01
CORR	0.182	0.402	0.38	0.24	0.33	0.11
N	1200	400	1200	1200	300	1200
Batch STD	0.75	1.508	0.051	0.016	0.952	0.014

**Experiment No: 16**

Three Kanban, Three stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.01	41	60	0.014	1.38	0.95
STD	0.31	3.48	0.67	0.38	2.27	0.01
CORR	-0.0021	0.28	-0.49	0.35	0.26	0.03
N	1200	300	1200	1200	300	1200
Batch STD	0.168	1.521	0.0	0.118	0.85	0.014

**Experiment No: 17**

Three Kanban, Three stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.627	40	60	0	2.28	0.95
STD	0.802	3.42	0.73	0	2.19	0.013
CORR	0.133	0.41	-0.51	0	0.35	0.06
N	1200	400	1200		300	1200
Batch STD	0.299	1.71	0.023	0	0.99	0.014

**Experiment No: 18**

Three Kanban, Three stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.57	39	60	0	2.76	0.95
STD	1.74	4.032	0.794	0	2.93	0.013
CORR	0.2023	0.39	-0.47	0	0.321	0.095
N	1200	240	1200		200	1200
Batch STD	0.64	1.73	0.046	0	1.211	0.014

**Experiment No: 19**

One Kanban, Three stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	12.75	12	59	11.67	0.93	0.951
STD	5.23	2.17	1.44	6.52	1.31	0.015

CORR	0.35	0.30	0.305	0.38	0.34	-0.04
N	600	400	1200	600	400	1200
Batch STD	2.09	0.88	0.132	2.64	0.512	0.014

**Experiment No: 20**

One Kanban, Three stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	14.46	13	59	14.34	1.1323	0.95
STD	5.8653	1.96	1.47	6.32	1.38	0.01
CORR	0.3254	0.36	0.26	0.38	0.39	0.01
N	300	300	1200	400	400	1200
Batch STD	2.076	0.803	0.092	2.768	0.633	0.014

**Experiment No: 21**

One Kanban, Three stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	19.102	12	59	20.52	2.163	0.96
STD	8.38	2.1225	1.3	7.07	1.56	0.015
CORR	0.32	0.34	0.19	0.41	0.38	0.051
N	100	150	1200	240	150	1200
Batch STD	3.781	1.070	0.209	3.366	0.798	0.014

**Experiment No: 22**

Two Kanban, Three stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.066	33	60	0.19	1.01	0.95
STD	0.97	3.0362	0.67	1.36	2.031	0.011
CORR	0.284	0.387	-0.48	0.096	0.4091	-0.0156
N	1200	400	1200	1200	400	1200
Batch STD	0.395	1.302	0.04	0.435	0.834	0.014

**Experiment No: 23**

Two Kanban, Three stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.66	33	60	0.25	1.423	0.95
STD	1.41	3.06	0.68	1.64	2.09	0.012
CORR	0.3158	0.39	-0.49	0.173	0.35	0.064

N	1200	300	1200	1200	240	1200
Batch STD	0.602	1.45	0.051	0.609	0.978	0.014

**Experiment No: 24**

Two Kanban, Three stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.511	31	60	0.3025	2.346	0.949
STD	1.93	3.51	0.72	1.96	2.54	0.01
CORR	0.27	0.35	-0.51	0.25	0.37	0.11
N	1200	300	1200	1200	300	1200
Batch STD	0.743	1.567	0	0.741	1.14	0.014

**Experiment No: 25**

Three Kanban, Three stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	7.96	53	60	0.049	1.1	0.95
STD	0.503	3.20	0.69	0.55	2.06	0.01
CORR	0.23	0.38	-0.51	0.16	0.39	0.05
N	1200	300	1200	1200	300	1200
Batch STD	0.222	1.798	0.023	0.169	0.779	0.014

**Experiment No: 26**

Three Kanban, Three stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.5021	53	59	0.043	1.715	0.95
STD	0.75	2.78	1.19	0.49	1.8	0.012
CORR	0.069	0.33	0.35	0.076	0.33	0.075
N	1200	400	240	1200	400	1200
Batch STD	0.28	1.81	0.052	0.16	0.82	0.014

**Experiment No: 27**

Three Kanban, Three stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.4451	51	59	0.069	2.198	0.95
STD	1.523	3.57	1.23	0.79	2.622	0.0134
CORR	0.124	0.37	0.38	0.29	0.403	0.074
N	1200	240	300	1200	240	1200

Batch STD	0.56	1.94	0.114	0.31	1.135	0.014
-----------	------	------	-------	------	-------	-------

**Experiment No: 28**

One Kanban, four stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	54.07	7	59	28.17	0.79	0.95
STD	14.88	1.44	1.68	6.49	0.87	0.016
CORR	0.35	0.35	0.26	0.27	0.37	-0.014
N	50	400	1200	50	600	1200
Batch STD	12.136	0.755	0.343	3.898	0.426	0.014

**Experiment No: 29**

One Kanban, four stages, three product types, second level of processing time variation

No steady state reached

**Experiment No: 30**

One Kanban, four stages, three product types, third level of processing time variation

No steady state reached

**Experiment No: 31**

Two Kanban, four stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.69	21	60	0.669	1.38	0.94
STD	1.99	3.41	0.77	3.06	1.76	0.012
CORR	0.34	0.37	-0.44	0.30	0.41	0.018
N	600	300	1200	600	300	1200
Batch STD	0.842	1.9	0.051	1.251	1.043	0.014

**Experiment No: 32**

Two Kanban, four stages, three product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	10.367	19	60	1.69	2.226	0.94
STD	3.43	3.69	0.864	4.15	2.14	0.014
CORR	0.32	0.39	-0.42	0.39	0.39	0.073
N	240	200	1200	300	200	1200
Batch STD	1.596	1.827	0.089	1.609	1.035	0.014

**Experiment No: 33**

Two Kanban, four stages, three product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	12.007	19	59	1.86	2.9	0.94
STD	4.99	3.693	4.65	1.45	2.28	0.015
CORR	0.35	0.35	0.28	0.29	0.36	0.09
N	300	200	300	1200	200	1200
Batch STD	2.179	1.724	0.114	3.509	1.043	0.014

**Experiment No: 34**

Three Kanban, four stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.34	36	60	0.012	1.56	0.94
STD	0.45	3.93	0.69	0.15	2.085	0.0114
CORR	0.0424	0.39	-0.49	-0.006	0.40	0.077
N	1200	200	1200	1200	200	1200
Batch STD	0.213	1.69	0.023	0.055	0.848	0.014

**Experiment No: 35**

Three Kanban, four stages, three product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.25	35	60	0.1075	2.067	0.94
STD	1.28	3.95	0.77	1.06	2.06	0.013
CORR	0.16	0.32	-0.49	0.38	0.34	0.017
N	1200	240	1200	1200	240	1200
Batch STD	0.47	1.53	0.046	0.44	0.76	0.014

**Experiment No: 36**

Three Kanban, four stages, three product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	10.61	33	60	0.133	3.41	0.94
STD	2.35	4.77	0.88	0.99	2.78	0.014
CORR	0.22	0.38	-0.49	0.24	0.4	0.02
N	1200	150	1200	1200	150	1200

Batch STD	0.93	2.05	0.05	0.37	1.189	0.014
-----------	------	------	------	------	-------	-------

**Experiment No: 37**

One Kanban, four stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.57	8	59.98	18.501	1.76	0.94
STD	23.98	2.17	1.67	9.33	2.32	0.021
CORR	0.39	0.34	0.06	0.41	0.39	0.13
N	40	50	1200	50	300	1200
Batch STD	1.204	0.85	0.54	3.74	1.19	0.015

**Experiment No: 38**

One Kanban, four stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	12.44	11	60	5.85	2.25	0.93
STD	7.124	2.5	1.05	2.91	1.32	0.02
CORR	0.31	0.38	-0.29	0.34	0.36	0.03
N	100	240	1200	200	240	1200
Batch STD	3.037	1.18	0.159	3.56	0.612	0.014

**Experiment No: 39**

One Kanban, four stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	13.44	11	60	6.95	2.42	0.94
STD	7.39	2.29	1.14	8.42	1.20	0.023
CORR	0.32	0.39	-0.29	0.37	0.39	0.049
N	150	400	1200	300	400	1200
Batch STD	2.69	0.96	0.123	3.886	0.48	0.014

**Experiment No: 40**

Two Kanban, four stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.13	27	60	0.68	1.89	0.94
STD	6.023	5.36	0.73	2.22	3.49	0.014
CORR	0.339	0.397	-0.42	0.34	0.39	0.049



N	150	100	1200	120	80	1200
Batch STD	2.42	2.657	0.083	0.873	1.893	0.014

**Experiment No: 41**

Two Kanban, four stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.66	33	60	0.2	2.25	0.93
STD	0.99	3.45	0.73	1.42	1.57	0.015
CORR	0.061	0.403	-0.52	0.104	0.35	-0.026
N	1200	400	1200	1200	300	1200
Batch STD	0.359	1.579	0.051	0.465	0.652	0.014

**Experiment No: 42**

Two Kanban, four stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.64	26	60	0.249	3.09	0.94
STD	1.84	3.72	0.78	1.74	2.017	0.015
CORR	0.153	0.332	-0.48	0.24	0.34	-0.0024
N	1200	240	1200	1200	240	1200
Batch STD	0.729	1.228	0.032	0.732	0.65	0.014

**Experiment No: 43**

Three Kanban, four stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.0156	50	60	0	0.932	0.939
STD	0.296	3.41	0.65	0	1.58	0.012
CORR	0.0014	0.396	-0.456	0	0.39	0.0347
N	1200	400	1200		400	1200
Batch STD	0.163	1.99	0.046	0	0.668	0.014

**Experiment No: 44**

Three Kanban, four stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.62	51	60	0	1.415	0.93
STD	0.807	3.55	0.75	0	1.72	0.014

CORR	0.064	0.33	-0.52	0	0.34	0.04
N	1200	240	1200		240	1200
Batch STD	0.286	2.157	0.046	0	0.817	0.014

**Experiment No: 45**

Three Kanban, four stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.914	49	60	0.19	2.38	0.94
STD	3.64	4.13	0.82	1.52	2.51	0.015
CORR	0.33	0.358	-0.46	0.36	0.36	0.052
N	120	60	1200	120	50	1200
Batch STD	1.546	1.77	0.079	0.584	0.93	0.014

**Experiment No: 46**

One Kanban, four stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	31.225	13	60	21.37	2.36	0.94
STD	25.84	3.12	1.3	8.47	2.08	0.017
CORR	0.35	0.39	-0.054	0.36	0.41	0.1218
N	40	100	1200	80	120	1200
Batch STD	9.569	1.290	0.353	3.288	0.837	0.015

**Experiment No: 47**

One Kanban, four stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	43.99	16	59	24.02	3.88	0.94
STD	18.68	2.175	1.379	9.231	2.198	0.0146
CORR	0.35	0.35	0.37	0.40	0.364	0.018
N	100	300	1200	240	80	1200
Batch STD	7.618	0.767	0.092	1.15	0.128	0.014

**Experiment No: 48**

One Kanban, four stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	58.59	15	60	20.2	4.517	0.94
STD	12.11	2.604	1.833	10.875	1.1074	0.018

CORR	0.414	0.353	0.41	0.38	0.37	0.015
N	200	200	120	120	150	1200
Batch STD	3.95	0.924	0.033	2.74	0.457	0.014

**Experiment No: 49**

Two Kanban, four stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.179	38	60	0.295	1.365	0.941
STD	1.44	5.77	0.696	1.8131	3.65	0.014
CORR	0.265	0.396	-0.5146	0.38	0.39	0.09
N	400	200	1200	1200	200	1200
Batch STD	0.559	2.393	0.04	0.893	1.429	0.014

**Experiment No: 50**

Two Kanban, four stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.624	41	60	0.2267	2.0189	0.938
STD	1.243	2.93	0.71	1.55	1.43	0.0135
CORR	0.3097	0.382	-0.48	0.187	0.378	0.041
N	1200	300	1200	1200	300	1200
Batch STD	0.573	1.703	0.032	0.646	0.704	0.014

**Experiment No: 51**

Two Kanban, four stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.52	40	60	0.25	2.84	0.938
STD	1.825	3.5009	0.745	1.65	1.81	0.015
CORR	0.256	0.396	-0.45	0.23	0.339	0.0406
N	1200	300	1200	1200	240	1200
Batch STD	0.778	1.726	0.065	0.702	0.714	0.014

**Experiment No: 52**

Three Kanban, four stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	7.946	61	60	0.037	1.61	0.94
STD	0.365	5.42	0.66	0.5	3.18	0.014

CORR	0.054	0.36	-0.51	0.055	0.365	-0.0013
N	1200	75	1200	1200	75	1200
Batch STD	0.189	8.352	0.023	0.164	1.158	0.014

### **Experiment No: 53**

Three Kanban, four stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.54	63	60	0.045	2.23	0.938
STD	0.84	3.61	0.663	0.518	1.71	0.014
CORR	0.074	0.39	-0.50	0.15	0.39	0.07
N	1200	400	1200	1200	400	1200
Batch STD	0.284	2.39	0.023	0.175	0.719	0.014

### **Experiment No: 54**

Three Kanban, four stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.32	62	60	0.04	2.57	0.937
STD	1.38	3.38	0.72	0.41	1.68	0.015
CORR	0.17	0.34	-0.52	-0.0097	0.35	0.04
N	1200	300	1000	1200	300	1200
Batch STD	0.528	2.27	0.056	0.136	0.668	0.014

### **Experiment No: 55**

One Kanban, five stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	43.559	6	59	26.1	0.651	0.93
STD	33.89	1.3647	1.63	6.62	0.71	0.17
CORR	0.417	0.407	0.33	0.39	0.36	-0.012
N	40	300	1200	50	600	1200
Batch STD	7.081	0.796	0.378	1.55	0.367	0.014

### **Experiment No: 56**

One Kanban, five stages, three product types, second level of processing time variation

No steady state

**Experiment No: 57**

One Kanban, five stages, three product types, third level of processing time variation

No steady state

**Experiment No: 58**

Two Kanban, five stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.045	19	59	1.042	1.229	0.93
STD	3.68	3.62	1.99	3.68	1.47	0.014
CORR	0.27	0.35	0.30	0.41	0.36	0.062
N	300	240	1200	400	240	1200
Batch STD	1.417	1.533	0.072	1.59	0.63	0.014

**Experiment No: 59**

One Kanban, five stages, three product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.47	19	60	0.584	1.505	0.933
STD	2.01	3.39	0.79	3.05	1.53	0.014
CORR	0.35	0.37	-0.49	0.33	0.39	-0.034
N	1200	240	1200	1200	240	1200
Batch STD	0.909	1.126	0.023	1.395	0.515	0.014

**Experiment No: 60**

One Kanban, five stages, three product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	12.267	18	60	2.212	2.71	0.934
STD	5.073	3.964	1.06	5.045	1.92	0.017
CORR	0.41	0.41	-0.39	0.37	0.264	0.0624
N	240	240	1200	400	200	1200
Batch STD	2.145	1.553	0.107	2.393	0.710	0.014

**Experiment No: 61**

Three Kanban, five stages, three product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
--	-----	-----	------	-------	--------	-----

MEAN	8.3078	34	60	0.035	1.38	0.93
STD	0.42	4.04	0.702	0.2981	1.53	0.013
CORR	0.0196	0.3975	-0.4788	0.645	0.3617	0.0086
N	1200	200	1200	1200	150	1200
Batch STD	0.188	1.884	0.023	0.114	0.614	0.014

**Experiment No: 62**

One Kanban, five stages, three product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.2086	33	60	0.053	1.703	0.934
STD	1.244	3.64	0.75	0.45	1.48	0.015
CORR	0.182	0.279	-0.479	0.055	0.298	0.0118
N	1200	150	1200	1200	150	1200
Batch STD	0.485	1.477	0.046	0.149	0.571	0.014

**Experiment No: 63**

One Kanban, five stages, three product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	10.65	31	60	0.177	2.464	0.934
STD	2.584	4.2209	0.89	1.69	1.997	0.0154
CORR	0.2481	0.39	-0.475	0.2596	0.402	0.047
N	1200	200	1200	1200	200	1200
Batch STD	0.905	1.77	0.065	0.63	0.854	0.014

**Experiment No: 64**

One Kanban, five stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.681	14	59	3.722	0.618	0.929
STD	2.75	9.19	1.34	2.56	1.18	0.016
CORR	0.341	0.39	0.34	0.39	0.34	0.115
N	1200	75	1200	60	240	1200
Batch STD	2.551	0.989	0.086	1.359	0.267	0.014

**Experiment No: 65**

One Kanban, five stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	12.41	14	60	5.711	0.96	0.93
STD	8.57	2.48	0.647	8.14	0.96	0.017

CORR	0.37	0.39	-0.26	0.42	0.39	0.009
N	150	240	1200	240	240	1200
Batch STD	3.932	1.152	0.126	3.944	0.433	0.014

**Experiment No: 66**

One Kanban, five stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	13.368	13	60	7.2	1.324	0.93
STD	7.7025	2.3362	1.079	8.68	0.985	0.017
CORR	0.4129	0.3676	-0.319	0.4061	0.3638	0.0019
N	240	300	1200	400	300	1200
Batch STD	4.88	1.219	0.126	4.679	0.466	0.014

**Experiment No: 67**

Two Kanban, five stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.1024	42	60	0.1642	0.6885	0.9292
STD	1.2093	5.2828	0.7009	1.9451	1.1431	0.0135
CORR	0.3608	0.3826	-0.4669	0.330	0.4066	-0.008
N	1200	60	1200	1200	400	1200
Batch STD	0.547	0.972	0.001	0.853	0.465	0.014

**Experiment No: 68**

Two Kanban, five stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.64	38	60	0.1183	1.065	0.93
STD	0.88	3.44	0.7237	0.874	1.34	0.015
CORR	0.071	0.3935	-0.5016	0.0647	0.4135	-0.0161
N	1200	300	1200	1200	300	1200
Batch STD	0.326	1.877	0.046	0.288	0.685	0.014

**Experiment No: 69**

Two Kanban, five stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.689	37	60.0	0.3442	1.515	0.92
STD	1.98	3.45	0.81	1.82	1.445	0.0164

CORR	0.2123	0.41	-0.486	0.263	0.41	0.039
N	1200	240	1200	600	240	1200
Batch STD	0.826	1.59	0.001	0.76	0.586	0.014

**Experiment No: 70**

Three Kanban, five stages, four product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.035	62	60	0.022	0.7624	0.92
STD	0.43	3.44	0.67	0.69	1.234	0.0139
CORR	0.066	0.397	-0.488	0.0819	0.3883	0.006
N	1200	300	1200	1200	300	1200
Batch STD	0.202	2.353	0.023	0.237	0.529	0.014

**Experiment No: 71**

Three Kanban, five stages, four product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.591	61	60	0	1.024	0.92
STD	0.74	3.303	0.7254	0	1.24	0.015
CORR	0.035	0.313	-0.5135	0	0.305	-0.063
N	1200	300	1200	0	300	1200
Batch STD	0.288	2.269	0.046	0	0.497	0.014

**Experiment No: 72**

Three Kanban, five stages, four product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.54	60	60	0	1.634	0.929
STD	1.59	3.85	0.789	0	1.53	0.0165
CORR	0.1917	0.3145	-0.4833	0	0.3108	0.0087
N	1200	240	1200		240	1200
Batch STD	0.584	2.442	0.001	0	0.662	0.014

**Experiment No: 73**

One Kanban, five stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	12.704	20	60	11.576	0.6181	0.93
STD	4.78	2.515	0.896	5.9417	0.8917	0.0164
CORR	0.4183	0.367	-0.3458	0.3755	0.3551	-0.001



N	400	400	1200	400	400	1200
Batch STD	1.99	1.11	0.1	2.392	0.368	0.014

**Experiment No: 74**

One Kanban, five stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	13.885	20	59	13.912	0.699	0.93
STD	4.39	2.22	1.375	6.14	0.798	0.0166
CORR	0.31	0.3835	0.3519	0.3676	0.3579	-0.0601
N	400	600	1200	600	600	1200
Batch STD	1.607	0.956	0.086	2.514	0.321	0.014

**Experiment No: 75**

One Kanban, five stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	19.633	19	59	21.53	1.3531	0.93
STD	8.35	2.56	1.63	6.81	1.025	0.018
CORR	0.324	0.41	0.18	0.38	0.36	0.039
N	150	240	1200	240	200	1200
Batch STD	4.643	1.669	0.205	4.274	0.677	0.014

**Experiment No: 76**

Two Kanban, five stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.11	51	60	0.257	0.6838	0.93
STD	1.174	2.959	0.6822	1.78	1.041	0.015
CORR	0.256	0.31	-0.504	0.089	0.3073	-0.0414
N	1200	400	1200	1200	400	1200
Batch STD	0.461	1.932	0.046	0.582	0.448	0.013

**Experiment No: 77**

Two Kanban, five stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.722	50	60	0.3283	0.981	0.9289
STD	1.674	3.4107	0.699	2.0256	1.2984	0.0161
CORR	0.4059	0.3613	-0.4898	0.2338	0.3811	-0.0419
N	1200	300	1200	1200	300	1200

Batch STD	0.855	2.228	0.04	0.906	0.654	0.014
-----------	-------	-------	------	-------	-------	-------

**Experiment No: 78**

Two Kanban, five stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.6319	48	60	0.4284	1.3173	0.928
STD	1.546	3.542	0.6991	2.028	1.263	0.016
CORR	0.3858	0.4197	-0.4898	0.2357	0.4037	-0.0414
N	300	200	1200	1200	240	1200
Batch STD	0.437	1.825	0.021	0.847	0.467	0.014

**Experiment No: 79**

Three Kanban, five stages, five product types, first level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	7.9523	81	60	0.021	0.595	0.9287
STD	0.3147	2.833	0.6579	0.3117	0.9896	0.0131
CORR	0.0123	0.3626	-0.4894	-0.0045	0.3573	-0.0368
N	1200	300	1200	1200	300	1200
Batch STD	0.168	2.634	0.152	0.109	0.441	0.013

**Experiment No: 80**

Three Kanban, five stages, five product types, second level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	8.5224	82	60	0.045	0.93	0.93
STD	0.74	3.474	0.6579	0.4288	1.2618	0.0156
CORR	0.1168	0.3126	-0.512	0.0112	0.3152	-0.0535
N	1200	400	1200	1200	400	1200
Batch STD	0.288	2.645	0.04	0.152	0.499	0.014

**Experiment No: 81**

Three Kanban, five stages, five product types, third level of processing time variation

	TYS	WIP	TPUT	STOCK	FKTIME	UTL
MEAN	9.4457	80	60	0.1042	1.38	0.93
STD	1.71	3.877	0.73	0.88	1.467	0.0165
CORR	0.164	0.3599	-0.4977	0.1131	0.3682	0.045
N	1200	400	1200	1200	400	1200
Batch STD	0.648	2.751	0.061	0.299	0.594	0.014

## Appendix - 2

This appendix contains the results of the analysis of variance (ANOVA) performed on the time in system. PROC GLM of the SAS-System was used to perform the ANOVA. In the following report:

1. The variable name KANBANS refers to the number of kanbans maintained at each of the stages on the feeder line.
2. The variable STAGES refers to the number of stages on the feeder line.
3. the variable PROD refers to the number of product types produced on the feeder line.
4. The variable VARI refers to the level of processing time variability at the stages on the feeder line.
5. The variable TYS refers to the time in system.

General Linear Models Procedure  
Class Level Information

Class	Levels	Values
KANBANS	3	1 2 3
STAGES	3	3 4 5
PROD	3	3 4 5
VARI	3	1 2 3

Number of observations in data set = 69

General Linear Models Procedure

Dependent Variable: TYS

Source	DF	Sum of Squares	F Value	Pr > F
Model	31	364.27189540	26.50	0.0001
Error	37	16.40632303		
Corrected Total	68	380.67821843		



## Appendix - 3

This appendix contains the results of the analysis of variance (ANOVA) performed on the work-in-process (WIP). PROC GLM of the SAS-System was used to perform the ANOVA. In the following report:

1. The variable name KANBANS refers to the number of kanbans maintained at each of the stages on the feeder line.
2. The variable STAGES refers to the number of stages on the feeder line.
3. the variable PROD refers to the number of product types produced on the feeder line.
4. The variable VARI refers to the level of processing time variability at the stages on the feeder line.
5. The variable WIP refers to the work-in-process.

General Linear Models Procedure  
Class Level Information

Class	Levels	Values
KANBANS	3	1 2 3
STAGES	3	3 4 5
PROD	3	3 4 5
VARI	3	1 2 3

Number of observations in data set = 69

General Linear Models Procedure

Dependent Variable: WIP

Source	DF	Sum of Squares	F Value	Pr > F
Model	31	23833.9840499	312.00	0.0001
Error	37	91.1753704		
Corrected Total	68	23925.1594203		

	R-Square	C.V.	WIP Mean	
	0.996189	4.680838	33.5362319	
Source	DF	Type I SS	F Value	Pr > F
KANBANS	2	13215.2779388	2681.45	0.0001
STAGES	2	2037.3444444	413.39	0.0001
PROD	2	7190.2522222	1458.94	0.0001
VARI	2	32.7246377	6.64	0.0034
KANBANS*STAGES	4	290.9792593	29.52	0.0001
KANBANS*PROD	3	402.2592593	54.41	0.0001
KANBANS*VARI	4	6.2235105	0.63	0.6432
STAGES*PROD	4	636.6388889	64.59	0.0001
STAGES*VARI	4	16.3435185	1.66	0.1805
PROD*VARI	4	5.9403704	0.60	0.6631
Source	DF	Type III SS	F Value	Pr > F
KANBANS	2	14094.2407407	2859.80	0.0001
STAGES	2	1190.7625000	241.61	0.0001
PROD	2	6872.9166667	1394.55	0.0001
VARI	2	21.1606326	4.29	0.0210
KANBANS*STAGES	4	508.3703704	51.58	0.0001
KANBANS*PROD	3	401.8148148	54.35	0.0001
KANBANS*VARI	4	6.1214049	0.62	0.6504
STAGES*PROD	4	636.6388889	64.59	0.0001
STAGES*VARI	4	17.1315079	1.74	0.1624
PROD*VARI	4	5.9403704	0.60	0.6631

## Appendix - 4

This appendix contains the results of the analysis of variance (ANOVA) performed on the stockout (STOCK). PROC GLM of the SAS-System was used to perform the ANOVA.

In the following report:

1. The variable name KANBANS refers to the number of kanbans maintained at each of the stages on the feeder line.
2. The variable STAGES refers to the number of stages on the feeder line.
3. the variable PROD refers to the number of product types produced on the feeder line.
4. The variable VARI refers to the level of processing time variability at the stages on the feeder line.
5. The variable STOCK refers to stockout.

General Linear Models Procedure  
Class Level Information

Class	Levels	Values
KANBANS	3	1 2 3
STAGES	3	3 4 5
PROD	3	3 4 5
VARI	3	1 2 3

Number of observations in data set = 69

General Linear Models Procedure

Dependent Variable: STOCK

Source	DF	Sum of Squares	F Value	Pr > F
Model	31	1541.52878422	13.11	0.0001
Error	37	140.38425926		
Corrected Total	68	1681.91304348		





## Appendix - 5

This appendix contains the results of the analysis of variance (ANOVA) performed on the kanban waiting time (FKTIME). PROC GLM of the SAS-System was used to perform the ANOVA. In the following report:

1. The variable name KANBANS refers to the number of kanbans maintained at each of the stages on the feeder line.
2. The variable STAGES refers to the number of stages on the feeder line.
3. the variable PROD refers to the number of product types produced on the feeder line.
4. The variable VARI refers to the level of processing time variability at the stages on the feeder line.
5. The variable FKTIME refers to the kanban waiting time.

General Linear Models Procedure  
Class Level Information

Class	Levels	Values
KANBANS	3	1 2 3
STAGES	3	3 4 5
PROD	3	3 4 5
VARI	3	1 2 3

Number of observations in data set = 69

General Linear Models Procedure

Dependent Variable: FKTIME

Source	DF	Sum of Squares	F Value	Pr > F
Model	31	40.79957489	27.60	0.0001
Error	37	1.76427088		
Corrected Total	68	42.56384577		



## Appendix - 6

This appendix contains the results of the analysis of variance (ANOVA) performed on the utilization (UTL). PROC GLM of the SAS-System was used to perform the ANOVA. In the following report:

1. The variable name KANBANS refers to the number of kanbans maintained at each of the stages on the feeder line.
2. The variable STAGES refers to the number of stages on the feeder line.
3. the variable PROD refers to the number of product types produced on the feeder line.
4. The variable VARI refers to the level of processing time variability at the stages on the feeder line.
5. The variable UTL refers to utilization.

General Linear Models Procedure  
Class Level Information

Class	Levels	Values
KANBANS	3	1 2 3
STAGES	3	3 4 5
PROD	3	3 4 5
VARI	3	1 2 3

Number of observations in data set = 69

General Linear Models Procedure

Dependent Variable: UTL

Source	DF	Sum of Squares	F Value	Pr > F
Model	31	0.00781340	32.70	0.0001
Error	37	0.00028515		
Corrected Total	68	0.00809855		

R-Square	C.V.	UTL Mean
0.964790	0.295284	0.94014493

Source	DF	Type I SS	F Value	Pr > F
KANBANS	2	0.00000225	0.15	0.8644
STAGES	2	0.00708549	459.70	0.0001
PROD	2	0.00035065	22.75	0.0001
VARI	2	0.00003768	2.44	0.1006
KANBANS*STAGES	4	0.00003052	0.99	0.4251
KANBANS*PROD	3	0.00000648	0.28	0.8392
KANBANS*VARI	4	0.00003195	1.04	0.4015
STAGES*PROD	4	0.00018102	5.87	0.0009
STAGES*VARI	4	0.00006201	2.01	0.1130
PROD*VARI	4	0.00002535	0.82	0.5194

Source	DF	Type III SS	F Value	Pr > F
KANBANS	2	0.00001435	0.93	0.4031
STAGES	2	0.00667028	432.76	0.0001
PROD	2	0.00033509	21.74	0.0001
VARI	2	0.00004108	2.67	0.0829
KANBANS*STAGES	4	0.00001343	0.44	0.7821
KANBANS*PROD	3	0.00000509	0.22	0.8817
KANBANS*VARI	4	0.00002868	0.93	0.4570
STAGES*PROD	4	0.00018102	5.87	0.0009
STAGES*VARI	4	0.00005645	1.83	0.1435
PROD*VARI	4	0.00002535	0.82	0.5194

## Appendix - 7

This appendix contains the results of the analysis of variance (ANOVA) performed on the throughput (TPUT). PROC GLM of the SAS-System was used to perform the ANOVA. In the following report:

1. The variable name KANBANS refers to the number of kanbans maintained at each of the stages on the feeder line.
2. The variable STAGES refers to the number of stages on the feeder line.
3. the variable PROD refers to the number of product types produced on the feeder line.
4. The variable VARI refers to the level of processing time variability at the stages on the feeder line.
5. The variable TPUT refers to throughput.

General Linear Models Procedure  
Class Level Information

Class	Levels	Values
KANBANS	3	1 2 3
STAGES	3	3 4 5
PROD	3	3 4 5
VARI	3	1 2 3

Number of observations in data set = 69

General Linear Models Procedure

Dependent Variable: TPUT

Source	DF	Sum of Squares	F Value	Pr > F
Model	31	7.20246377	1.89	0.0318
Error	37	4.53666667		
Corrected Total	68	11.73913043		



## Appendix – 8

This appendix contains the sequencing algorithm used in this research to generate the sequence of product types to be assembled on the final assembly line. The following notation is used (refer to figure 5-2).

$Q$  = Total production quantity of all products  $A_i$  ( $i = 1, \dots, \mathbf{a}$ )

$$= \sum_{i=1}^{\mathbf{a}} Q_i, (Q_i = \text{production quantity of each product } A_i)$$

$N_j$  = Total necessary quantity of part  $a_j$  to be consumed for producing all products  $A_i$  ( $i = 1, \dots, \mathbf{a}; j = 1, \dots, \mathbf{b}$ )

$X_{jk}$  = Total necessary quantity of part  $a_j$  to be utilized for producing the products of the determined sequence from first to  $K$ th.

$\frac{N_j}{Q}$  = Average necessary quantity of part  $a_j$  per unit of a product

$\frac{K \cdot N_j}{Q}$  = Average necessary quantity of part  $a_j$  for producing  $K$  units of products.

$$D_k = \|G_k - P_k\| = \sqrt{\sum_{j=1}^{\mathbf{b}} \left( \frac{K \cdot N_j}{Q} - X_{jk} \right)^2}$$

$b_{ij}$  = Necessary quantity of part  $a_j$  ( $j = 1, \dots, \mathbf{b}$ ) for producing one unit of product  $A_i$  ( $i = 1, \dots, \mathbf{a}$ ).

$S_k$  = The set of products that remain to be produced after the production of  $k$  units

Having defined the above notation, the following steps need to be done :

Step 1: Set  $K = 1, X_{j,k-1} = 0, (j = 1, \dots, \mathbf{b}), S_{k-1} = \{1, 2, \dots, \mathbf{a}\}$ .

Step 2: Set as  $K$  th order in the sequence product  $A_{i^*}$  which minimizes the distance  $D_k$ . The minimum distance will be found by the following formula :

$$D_{ki^*} = \min\{D_{ki}\}, \forall i \in S_{k-1},$$

$$\text{where } D_{ki} = \sqrt{\sum_{j=1}^{\mathbf{b}} \left( \frac{K \cdot N_j}{Q} - X_{j,k-1} - b_{ij} \right)^2}$$

Step 3: If all units of a product  $A_{i^*}$  were ordered and included in the sequence, then set  $S_k = S_{k-1} - \{i^*\}$ .

If some units of a product  $A_{i^*}$  are still remaining as being not ordered, then set  $S_k = S_{k-1}$ .

Step 4. If  $S_k = \Phi$  (empty set), the algorithm will end.

If  $S_k \neq \Phi$ , then compute  $X_{jk} = X_{j,k-1} + b_{ij}$  ( $j = 1, \dots, \mathbf{b}$ ) and go back to Step 2 after setting  $K = K + 1$ .



## Appendix - 9

This appendix contains the listing of the simulation program. Following is a list of variables and attributes that are referenced in the program:

- ATTRIB(1) : Time of entry into the system
- ATTRIB(2) : Product type
- ATTRIB(3) : Indicates whether it is a production ordering or withdrawal  
kanban
- ATTRIB(4) : Used to store the time of entry into a queue
- ATTRIB(5) : Used to evaluate the time spent in a queue
- II : Counter to count the number of products assembled  
during a shift

```
C-----  
C                               Subroutine INTCL  
C-----
```

```
      SUBROUTINE INTLC  
      COMMON /SCOM1/ATTRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,  
1 NCLNR, NCRDR, NPRNT, NNFUN, NNSET, NTAPE, SS(100), SSL(100), TNEXT,  
2 TNOW, XX(100)  
      COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT  
C  
C          Create the sequence of products to be produced  
C  
  
      CALL SEQUENCE  
      RETURN  
      END
```

```
C-----  
C                               Subroutine Event: handles the user defined events  
C-----
```

```
      SUBROUTINE EVENT(I)  
      COMMON /SCOM1/ATTRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,  
1 NCLNR, NCRDR, NPRNT, NNFUN, NNSET, NTAPE, SS(100), SSL(100), TNEXT,  
2 TNOW, XX(100)  
      COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT  
      COMMON/MYCOM2/ATSYS(400), IND, APOK1(400), IPOK1, AWKQ1(400), IWKQ1,  
3 APOK2(400), IPOK2, AWKQ2(400), IWKQ2, APOK3(400), IPOK3  
      COMMON/MYCOM3/LS(3), TOT(3)
```

```

GO TO (1,2,3,4,10) I
1  CALL  NXT
   RETURN
2  CALL  TSYS
   RETURN
3  CALL  SEQUENCE
   RETURN
4  CALL  STOCK
   RETURN
10 CALL  SHIFTOPT
   RETURN
   END

```

```

C-----
C          Subroutine STOCK: Evaluates if there is a stock out situation and increments the
C          variable that contains the number of stock out incidents during the shift
C-----

```

```

COMMON /SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,
1  NCLNR, NCRDR, NPRNT, NNFUN, NNSSET, NTAPE, SS(100), SSL(100), TNEXT,
2  TNOW, XX(100)
COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT
COMMON/MYCOM2/ATSYS(400), IND, APOK1(400), IPOK1, AWKQ1(400), IWKQ1,
3  APOK2(400), IPOK2, AWKQ2(400), IWKQ2, APOK3(400), IPOK3
COMMON/MYCOM3/LS(3), TOT(3)
DIMENSION A(12)

ITYPE=ATRIB(2)
IF (NNQ(2).GT.0) THEN
    DO 10 I=1, NNQ(2)
        CALL COPY(I, 2, A)
        ITEMP = A(2)
        IF (ITYPE.EQ.ITEMP) GO TO 20
10    CONTINUE
        LS(ITYPE) = LS(ITYPE) + 1
        TOT(ITYPE) = TOT(ITYPE) + 1
    ELSE
        LS(ITYPE) = LS(ITYPE) + 1
        TOT(ITYPE) = TOT(ITYPE) + 1
    END IF
20    RETURN
    END

```

```

C-----
C          SUBROUTINE OTPUT: prints stock out messages
C-----

```

```

SUBROUTINE  OPUT
COMMON /SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,
1 NCLNR, NCRDR, NPRNT, NNFUN, NNSSET, NTAPE, SS(100), SSL(100), TNEXT,
2 TNOW, XX(100)
COMMON/MYCOM3/LS(3), TOT(3)

WRITE(*,*) 'Stock out of type 1 is: ', TOT(1)
WRITE(*,*) 'Stock out of type 2 is: ', TOT(2)
WRITE(*,*) 'Stock out of type 3 is: ', TOT(3)
RETURN
END

```

```

C-----
C          SUBROUTINE  TSYS
C-----

```

```

SUBROUTINE  TSYS
COMMON /SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,
1 NCLNR, NCRDR, NPRNT, NNFUN, NNSSET, NTAPE, SS(100), SSL(100), TNEXT,
2 TNOW, XX(100)
COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT
COMMON/MYCOM2/ATSYS(400), IND, APOK1(400), IPOK1, AWKQ1(400), IWKQ1,
3 APOK2(400), IPOK2, AWKQ2(400), IWKQ2, APOK3(400), IPOK3
COMMON/MYCOM3/LS(3), TOT(3)

IND = IND + 1
ATSYS(IND) = TNOW - ATRIB(1)
IF (ATRIB(2).EQ. 1) THEN
    WRITE(*,*) 'NUMBER OF TP1 COMPLETED:', XX(34)
ELSE IF (ATRIB(2).EQ. 2) THEN
    WRITE(*,*) 'NUMBER OF TP2 COMPLETED:', XX(35)
ELSE
    WRITE(*,*) 'NUMBER OF TP3 COMPLETED:', XX(36)
END IF
WRITE (*,*) 'ITEM NO:', XX(37), 'COMPLETED'
RETURN
END

```

```

C-----
C          SUBROUTINE  SHIFTOPT
C-----

```

```

SUBROUTINE  SHIFTOPT
COMMON /SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,
1 NCLNR, NCRDR, NPRNT, NNFUN, NNSSET, NTAPE, SS(100), SSL(100), TNEXT,
2 TNOW, XX(100)
COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT
COMMON/MYCOM2/ATSYS(400), IND, APOK1(400), IPOK1, AWKQ1(400), IWKQ1,
3 APOK2(400), IPOK2, AWKQ2(400), IWKQ2, APOK3(400), IPOK3
COMMON/MYCOM3/LS(3), TOT(3)

```

```

TATSYS = 0
TTSYS = 0
TAPOK1 = 0
TPOK1 = 0
TAWKQ1 = 0
TAPOK2 = 0
TPOK2 = 0
TAWKQ2 = 0
TWKQ2 = 0
TAPOK3 = 0
TPOK3 = 0
STOCKOUT = 0
TTSYS = CCAVG(2)
TPOK1 = CCAVG(3)
TWKQ1 = CCAVG(4)
TPOK2 = CCAVG(5)
TWKQ2 = CCAVG(6)
TPOK3 = CCAVG(7)
ISTO = LS(1) + LS(2) + LS(3)
ITE = XX(34)
UTLFA = RRAVG(13)
UTL1 = RRAVG(14)
UTL2 = RRAVG(15)
UTL3 = RRAVG(16)
WIP = TTAVG(1)
HCTP1 = XX(9) + XX(12) + XX(16) + XX(19) + XX(22) + XX(25)
HCTP2 = XX(10) + XX(13) + XX(17) + XX(20) + XX(23) + XX(26)
HCTP3 = XX(11) + XX(14) + XX(18) + XX(21) + XX(24) + XX(27)
IXX = XX(34)
IXY = XX(35)
IXZ = XX(36)
WRITE(1, 11) IXX, IXY, IXZ, LS(1), LS(2), LS(3), TTSYS
WRITE(1, 16) TPOK1, TWKQ1, TPOK2, TWKQ2, TPOK3, UTLFA, UTL1
WRITE(1, 16) UTL2, UTL3, WIP, HCTP1, HCTP2, HCTP3
11  FORMAT (1X, 6I3, 1X, F8.2)
16  FORMAT (1X, 13F7.3)
    IPOK1 = 0
    IWKQ1 = 0
    IPOK2 = 0
    IWKQ2 = 0
    IPOK3 = 0
    IND = 0
    LS(1) = 0
    LS(2) = 0
    LS(3) = 0
    DO 19 I=9, 37
        XX(I) = 0
19  CONTINUE
    CALL CLEAR
    RETURN
    END

```

C-----  
C                   SUBROUTINE SEQUENCE  
C-----

```

SUBROUTINE SEQUENCE
COMMON /SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,
1 NCLNR, NCRDR, NPRNT, NNFUN, NNSET, NTAPE, SS(100), SSL(100), TNEXT,
2 TNOW, XX(100)
COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT
DIMENSION NQTY(3)
DIMENSION NSCOMP(3,4)
DIMENSION IPROD(100)
DIMENSION XNUMBCOMP(4)
DIMENSION D(3)
DIMENSION IX(4)
INTEGER TOTNUM
DATA NSCOMP/1,1,0,0,1,1,1,0,1,1,1,0/
REAL NTEMP
II = 0
CTIME = 0
TOTNUM = 0
INDEX = 0
NSHIFT = 0
NQTY(1) = 20
NQTY(2) = 22
NQTY(3) = 18
NPROD = 3
NSC = 4
DO 10 I = 1, NPROD
    TOTNUM = TOTNUM + NQTY(I)
    IPROD(I) = 0
    D(I) = 0
10 CONTINUE
NSHIFT = TOTNUM
XX(15) = NSHIFT
CTIME = 480.0/NSHIFT
SHIFT = SHIFT + 1

ISUM = 0
DO 30 I = 1, NSC
    DO 20 J = 1, NPROD
        ISUM = ISUM + NQTY(J) * NSCOMP(J, I)
20 CONTINUE
    XNUMBCOMP(I) = ISUM
    ISUM = 0
30 CONTINUE

DO 40 I = 1, NSC
    XNUMBCOMP(I) = XNUMBCOMP(I)/ TOTNUM
40 CONTINUE

DO 60 I = 0, NSC
```

```

        IX(I) = 0
60    CONTINUE

    TEMP = 0
    DO 200 K= 1, TOTNUM
        DO 100 I= 1, NPROD
            IF (IPROD(I) .EQ. NQTY(I)) THEN
                D(I) = 10000.0* I
                GO TO 100
            END IF
            DO 90 J = 1, NSC
                TEMP1 = K*XNUMBCOMP(J) - NSCOMP(I,J)-IX(J)
                TEMP = TEMP+ TEMP1 * TEMP1
90        CONTINUE
            D(I) = SQRT(TEMP)
            TEMP = 0
100       CONTINUE
            NTEMP = 200000.0
            DO 110 I = 1, NPROD
                IF (D(I). LT. NTEMP) THEN
                    NTEMP = D(I)
                    INDEX = I
                END IF
110        CONTINUE
            IPROD(INDEX) = IPROD(INDEX) + 1
            ISEQUENCE(K) = INDEX
            DO 120 I = 1, NSC
                IX(I) = IX(I) + NSCOMP(INDEX, I)
120        CONTINUE
200       CONTINUE
    RETURN
    END

```

```

C-----
C           SUBROUTINE  NXT
C-----

```

```

    SUBROUTINE  NXT
    COMMON /SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,
1  NCLNR, NCRDR, NPRNT, NNFUN, NNSSET, NTAPE, SS(100), SSL(100), TNEXT,
2  TNOW, XX(100)
    COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT

    ATRIB(2) = ISEQUENCE(II)
    ATRIB(1) = TNOW
    ATRIB(3) = 1
    IND = IND + 1
    WRITE (*,*), 'ITEM NUM', IND, 'TYPE', ATRIB(2), 'LAUNCHED AT', TNOW
    RETURN
    END

```

```

C-----

```

C                   SUBROUTINE USERF(I)

C-----

```
      SUBROUTINE USERF(I)
      COMMON /SCOM1/ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA, MSTOP,
1     NCLNR, NCRDR, NPRNT, NNFUN, NNSET, NTAPE, SS(100), SSL(100), TNEXT,
2     TNOW, XX(100)
      COMMON/MYCOM1, ISEQUENCE(100), CTIME, NSHIFT

      GO TO (1, 2) I

1     USERF = NNQ(2) + NNQ(4) + NNQ(6) + NNQ(8) + NNQ(10) + NNQ(12)
      RETURN
2     USERF = CTIME
      RETURN
      END
```

;THE SLAM NETWORK FOLLOWS

```
GEN, ARVIND, JIT BC, 04/04/96, 1, Y, Y, Y/Y, Y, Y/1;
LIM, 31, 12, 5600;
TIMST, USERF(1), WORK IN PROCESS;
NETWORK;
```

```
    RESOURCE/1, DMF1(1), 13;
    RESOURCE/2, DMF2(1), 14;
    RESOURCE/3, DUM11(1), 15;
    RESOURCE/4, DUM12(1), 16;
    RESOURCE/5, DUM21(1), 17
    RESOURC/6, DUM22(1), 18
    RESOURCE/7, DUM31(1), 19
    RESOURCE/8, DUM32(1), 20
    RESOURCE/9, DUM41(1), 21;
    RESOURCE/10, DUM42(1), 22;
    RESOURCE/11, DUM51(1), 23;
    RESOURCE/12, DUM52(1), 24;
    RESOURCE/13, WSFA(1), 25;
    RESOURCE/14, WS1(1), 29, 26;
    RESOURCE/15, WS2(1), 30, 27;
    RESOURCE/16, WS3(1), 31, 28;
```

```
    ;shift terminating entity
    CREATE, 480, 480;
    EVENT, 10;
    ASSIGN, XX(34) = 0, XX(33) =0;
    TERMINATE, 1200;
```

```
    CREATE,,,1;
GN   GOON, 1;
      ACT, EXPON(20000);
P1   PREEMPT(29), WS1,,12;
      ACT,RNORM(45, 5);
      FREE, WS1;
```

```

ACT,,,GN;

CREATE,,,,1;
GN1 GOON, 1;
      ACT, EXPON(20100);
P2  PREEMPT(30), WS2,,12;
      ACT,RNORM(50, 8);
      FREE, WS2;
      ACT,,,GN1;

CREATE,,,,1;
GN2 GOON, 1;
      ACT, EXPON(20150);
P3  PREEMPT(31), WS3,,12;
      ACT,RNORM(40, 5);
      FREE, WS3;
      ACT,,,GN2;

CREATE,8;
AS  ASSIGN, II=II+1,1;
      ACT,,II.GT.60.NXT;
      ACT,,,EV1;
NXT ASSIGN, II=1;
EV1  EVENT, 1;
      ACT;
      ASSIGN, ATRIB(4) = TNOW;
      EVENT, 4;
DMQ  QUEUE(1),,,MAT;
OMQ1 QUEUE(2),,,MAT;
MAT  MATCH, 2, DMQ/DF1, OMQ1/DF2;

DF1  AWAIT(13),DMF1;
AS01 ASSIGN,ATRIB(5) = TNOW - ATRIB(4);
      COLCT, ATRIB(5), DMQ WAITING;
      ACT,,,BAT;

DF2  AWAIT(14), DMF2;
AS02 ASSIGN, ATRIB(5) = TNOW - ATR(4), ATR(6) = ATR(10) * ATR(5), 1;
      ACT,, ATR(2).EQ.1, TP1F;
      ACT,,ATR(2).EQ. 2, TP2F;
      ACT,,ATR(2).EQ.3, TP3F;

TP1F ASSIGN, XX(9) = XX(9) + ATRIB(6);
      ACT,,,BAT;
TP2F ASSIGN, XX(10) = XX(10) + ATRIB(6);
      ACT,,,BAT;
TP3F ASSIGN, XX(11) = XX(11) + ATRIB(6);

BAT  BATCH, ½,3,,LAST/9,ALL(7);
A2   AWAIT(25), WSFA;
      FREE, DMF1;
      FREE, DMF2;

```



```

                ACT,ATR(9);
FWS  FREE, WSFA;
NSTF UNBATCH, 7;
      GOON, 1;
            ACT,, ATR(3).EQ.11, PKT1;
            ACT;
      COLCT, INT(1), TIME IN SYSTEM;
      GOON, 1;
            ACT,,ATRIB(2).EQ.1, TRK1;
            ACT,,ATRIB(2).EQ.2, TRK2;
            ACT,,ATRIB(2).EQ.3,TRK3;
TRK1  ASSIGN, XX(34) = XX(34) + 1;
      ACT,,,EV2;
TRK2  ASSIGN, XX(35) = XX(35) + 1;
      ACT,,,EV2;
TRK3  ASSIGN, XX(36) = XX(36) + 1;

EV2   ASSIGN, XX(37) = XX(37) + 1;
      EVENT, 2;
      TERMINAT;

;
;WORK STATION 1 PROCESSING
;
PKT1  ASSIGN, ATRIB(9) = 0, ATRIB(4) = TNOW;
POK1  QUEUE(3),,,MAT1;
IMQ1  QUEUE(4),,,MAT1;
MAT1  MATCH, 2, POK1/D11, IMQ1/D12;
;
D11   AWAIT(15), DUM11;
AS11  ASSIGN, ATRIB(5) = TNOW - ATRIB(4);
      COLCT, ATR(5), POK1 WAITING;
      ACT,,,BAT1;

;
D12   AWAIT(16), DUM12;
AS12  ASSIGN, ATRIB(5) = TNOW - ATRIB(4), ATRIB(6) = ATRIB(10) * ATRIB(5), 1;
      ACT,,ATRIB(2).EQ.1,TP11;
      ACT,,ATRIB(2).EQ.2, TP21;
      ACT,,ATRIB(2).EQ.3, TP31;

;
TP11  ASSIGN, XX(12) = XX(12) + ATRIB(6);
      ACT,,,BAT1;
TP21  ASSIGN, XX(12) = XX(13) + ATRIB(6);
      ACT,,,BAT1;
TP31  ASSIGN, XX(14) = XX(14) + ATRIB(6);
      ACT,,,BAT1;

;
BAT1  BATCH, ½,2,,LAST/8,9,ALL(7);
      AWAIT(26), WS1;
      FREE, DUM11;
      FREE, DUM12;
            ACT, ATRIB(9);
      GOON, 1;

```

```

ACT,,0.05, QC1;
ACT,,,OK1;
QC1  GOON, 1;
      ACT, RNORM(3.1, 0.8);
OK1  FREE, WS1;
NST1 UNBATCH, 7;
      GOON, 1;
      ACT,, ATRIB(3).EQ.21, WKT1;
      ACT;
      ASSIGN, ATRIB(4) = TNOW;
      ACT,, ATRIB(2).EQ.1, PSF1;
      ACT,,ATRIB(2).EQ.2, PSF2;
      ACT,,ATRIB(2).EQ.3, PSF3;

PSF1  ASSIGN, ATRIB(9) = RNORM(8.2, 0.82), ATR(8) = 1.0;
      ACT,,, OMQ1;
PSF2  ASSIGN, ATRIB(9) = RNORM(7., 0.7), ATR(8) = 1.0;
      ACT,,, OMQ1;
PSF3  ASSIGN, ATRIB(9) = RNORM(8.0, 0.8), ATR(8) = 1.0;
      ACT,,, OMQ1;

WKT1  ASSIGN, ATRIB(4) = TNOW;
WKQ1  QUEUE(5),,,,MAT2;
OMQ2  QUEUE(6),,,,MAT2;

MAT2  MATCH, 2, WKQ1/D21, OMQ2/D22;

D21   AWAIT(17), DUM21;
AS21  ASSIGN, ATRIB(5) = TNOW - ATRIB(4);
      COLCT, ATR(5), WKQ1 WAITING;
      ACT,,,BAT2;
D22   AWAIT(18), DUM22;
AS22  ASSIGN, ATR(5) = TNOW - ATR(4), ATR(6) = ATR(10) * ATR(6), 1;
      ACT,,ATR(2).EQ.1, TP12;
      ACT,,ATR(2).EQ.2, TP22;
      ACT,, ATR(2).EQ.3, TP32;
TP12  ASSIGN, XX(16) = XX(16) + ATRIB(6);
      ACT,,,BAT2;
TP22  ASSIGN, XX(17) = XX(17) + ATRIB(6);
      ACT,,,BAT2;
TP32  ASSIGN, XX(18) = XX(18) + ATRIB(6);

BAT2  BATCH, 1/2,2,,,ALL(7);
      FREE, DUM21;
      FREE, DUM22;
      ACT, 1;
      UNBATCH, 7;
      GOON, 1;
      ACT,,ATR(3).EQ.12, PKT2;
      ACT;
      ASSIGN, ATR(4) = TNOW, 1;
      ACT,,ATR(2).EQ.1, PS11;

```

ACT,,ATR(2).EQ.2, PS12;  
 ACT,,ATR(2).EQ.3, PS13;

PS11 ASSIGN, ATR(8) = 1.0, ATR(9) = RNORM(7.5, 0.75);  
 ACT,,IMQ1;

PS12 ASSIGN, ATR(8) = 1.0, ATR(9) = RNORM(6.2, 0.62);  
 ACT,,IMQ1;

PS13 ASSIGN, ATR(8) = 1.0, ATR(10) = RNORM( 7.9, 0.79);  
 ACT,,IMQ1;

PKT2 ASSIGN, ATR(9) = 0, ATR(8) = 0, ATRIB(4) = TNOW;  
 POK2 QUEUE(7),,,MAT3;  
 IMQ2 QUEUE(8),,,MAT3;

MAT3 MATCH, 2, POK2/D1, IMQ2/D32;  
 D31 AWAIT(19), DUM31;  
 AS31 ASSIGN, ATRIB(5) = TNOW - ATRIB(4);  
 COLCT, ATR(5), POK2 WAITING;  
 ACT,,BAT3;

D32 AWAIT(20), DUM32;  
 AS32 ASSIGN, ATRIB(5) = TNOW - ATR(4), ATR(6) = ATR(10) \* ATR(5), 1;  
 ACT,, ATR(2).EQ.1, TP13;  
 ACT,, ATR(2).EQ.2 TP23;  
 ACT,, ATR(2)EQ.3, TP33;

TP13 ASSIGN, XX(19) = XX(19) + ATRIB(6);  
 ACT,,BAT3;

TP23 ASSIGN, XX(20) = XX(20) + ATRIB(6);  
 ACT,,BAT3;

TP33 ASSIGN, XX(21) = XX(21) + ATRIB(6);

BAT3 BATCH, 1/2, 2,,LAST/8, 9, ALL(7);  
 AWAIT(27), WS2;  
 FREE, DUM31;  
 FREE, DUM32;  
 ACT, ATRIB(9);  
 GOON, 1;  
 ACT,, 0.05, QC2;  
 ACT,, OK2;

QC2 GOON, 1;  
 ACT, RNORM(3.2, 0.84);

OK2 FREE, WS2;

NST2 UNBATCH, 7;  
 GOON, 1;  
 ACT,, ATRIB(3).EQ.22, WKT2;  
 ACT;  
 ASSIGN, ATRIB(4) = TNOW;  
 ACT,, OMQ2;

WKT2 ASSIGN, ATRIB(4) = TNOW;  
 WKQ2 QUEUE(9),,,MAT4;  
 OMQ3 QUEUE(10),,,MAT4;

MAT4 MATCH, 2, WKQ2/D41, OMQ3/D42;  
D41 AWAIT(21), DUM41;  
AS41 ASSSIGN, ATRIB(5) = TNOW - ATRIB(4);  
COLCT, ATR(5), WKQ2 WAITING;  
ACT,,,BAT4;  
D42 AWAIT(22), DUM42;  
AS42 ASSIGN, ATR(5) = TNOW - ATR(4), ATR(6) = ATR(10) \* ATR(5);  
ACT,,ATR(2).EQ.1, TP14;  
ACT,, ATR(2).EQ.2, TP24;  
ACT,, ATR(2).EQ.3, TP34;  
TP14 ASSIGN, XX(22) = XX(22) + ATRIB(6);  
ACT,,,BAT4;  
TP24 ASSIGN, XX(23) = XX(23) + ATRIB(6);  
ACT,,,BAT4;  
TP34 ASSIGN, XX(24) = XX(24) + ATRIB(6);  
ACT,,,BAT4;  
  
BAT4 BATCH, ½,2,,,ALL(7);  
FREE, DUM41;  
FREE, DUM42;  
ACT, 1;  
UNBATCH, 7;  
GOON, 1;  
ACT,, ATR(3).EQ.13, PKT3;  
ACT;  
ASSIGN, ATRIB(4) = TNOW, 1;  
ACT,,ATR(2).EQ.1, PS21;  
ACT,, ATR(2).EQ.2, PS22;  
ACT,, ATR(2).EQ.3, PS23;  
  
PS21 ASSIGN, ATR(8) = 1.0, ATR(9) = RNORM(7.7, 0.77);  
ACT,,,IMQ2;  
PS22 ASSIGN, ATR(8) = 1.0, ATR(9) = RNOMR(7.4, 0.74);  
ACT,,,IMQ2;  
PS23 ASSIGN, ATR(8) = 1.0, ATR(9) = RNOMR(8.0, 0.8);  
ACT,,,IMQ2;  
PKT3 ASSIGN, ATR(8) = 0, ATR(9) = 0, ATR(4) = TNOW;  
POK3 QUEUE(11),,,,MAT5;  
IMQ3 QUEUE(12),,,,MAT5;  
MAT5 MATCH, 2, POK3/D51, IMQ3/D52  
D51 AWAIT(23), DUM51;  
AS51 ASSIGN, ATRIB(5) = TNOW - ATRIB(4);  
COLCT, ATR(5), POK3 WAITING;  
ACT,,,BAT5;  
D52 AWAIT(24), DUM52;  
AS52 ASSIGN, ATR(5) = TNOW - ATR(4) , ATR(6) = ATR(10) \* ATR(5), 1;  
ACT,,ATR(2).EQ.1, TP15;  
ACT,,ATR(2).EQ.2, TP25;  
ACT,,ATR(2).EQ.3, TP35;  
  
TP15 ASSIGN, XX(25) = XX(25) + ATRIB(6);

```

ACT,,,BAT5;
TP25  ASSIGN, XX(26) = XX(26) + ATRIB(6);
      ACT,,,BAT5;
TP35  ASSIGN, XX(27) = XX(27) + ATRIB(6);
      ACT,,,BAT5;
BAT5  BATCH, 1/2,2,,LAST/8,9,ALL(7);
      AWAIT(28), WS3;
      FREE, DUM51;
      FREE, DUM52;
      ACT, ATRIB(9);
      GOON, 1;
      ACT,, 0.05, QC3;
      ACT,,,OK3;
QC3   GOON, 1;
      ACT, RNORM(3.4, 0.78);
OK3   FREE, WS3;
NST3  UNBATCH, 7;
      GOON, 1;
      ACT,,ATR(3).EQ.23, AS7;
      ACT;
      ASSIGN, ATRIB(4)= TNOW;
      ACT,,OMQ3;
AS7   ASSIGN, ATR(4) = TNOW, 1;
      ACT,,ATR(2).EQ.1,PS31;
      ACT,,ATR(2).EQ.2,PS32;
      ACT,,ATR(2).EQ.3,PS33;
PS31  ASSIGN, ATR(8) = 1, ATR(9) = RNORM(7.6, 0.76);
      ACT,,,IMQ3;
PS32  ASSIGN, ATR(8) = 1, ATR(9) = RNORM(7.3, 0.73);
      ACT,,,IMQ3;
PS33  ASSIGN, ATR(8) = 1, ATR(9) = RNORM(8.25, 0.825);
      ACT,,,IMQ3;
      ENDNETWORK;
ENTRY/2, 0, 1,11, 0,0,0,0,1,5, 0.001, 1;
ENTRY/2, 0,2, 11, 0,0,0,0,1,5, 0.001, 1;
ENTRY/2, 0,3, 11, 0,0,0,0,1,5,0.001, 1;
ENTRY/4, 0,1, 21, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/4, 0, 2, 21, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/4, 0, 3, 21, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/6, 0, 1, 12, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/6, 0, 2, 12, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/6, 0, 3,, 12, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/8, 0, 1, 22, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/8, 0, 2, 22, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/8, 0, 3, 22, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/10, 0, 1, 13, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/10, 0, 2, 13, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/10, 0, 3, 13, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/12, 0, 1, 23, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/12, 0, 2, 23, 0, 0, 0, 0, 1, 5, 0.001, 1;
ENTRY/12, 0, 3, 23, 0, 0, 0, 0, 1, 5, 0.001, 1;
FIN;

```

**ABSTRACT**  
**SIMULATION STUDY OF A**  
**KANBAN CONTROLLED PRODUCTION SYSTEM**

**By Arvind R Krishnappa**

Just-In-Time (JIT) has become one of the most popular production control philosophies in the last three decades. JIT did not find acceptance in the American companies until after the oil crisis. To be more flexible and to adapt quickly to changes in the market, many American companies took to the recourse of JIT and have been extremely successful. JIT systems typically employ kanbans as a means of inventory control. A JIT system operating under kanban control is commonly termed as a “pull system” due to the way in which succeeding stages trigger production at preceding stages. Owing to this close dependence of stages on a production line, the performance of a kanban controlled JIT system is sensitive to various kinds of stochasticity.

It was the aim of this thesis to characterize such inventory systems under different conditions. In particular, the research focused on kanban controlled *feeder lines*. Various design and operational parameters like number of kanbans, number of stages, number of product types assembled and processing time variability were studied. Metrics such as time-in-system, throughput, kanban waiting time, utilization, stockout and work-in-process were used to measure the performance of the system. A simulation model was constructed to model the system and to carry out the various experiments conducted as part of this research. It was observed that time in system was significantly affected by the number of kanbans, number of product types and the level of processing time variability at the stages. The analysis of work-in-process indicated that it was affected by the number of kanbans, number of stages, number of product types and the level of processing time variability at the stages. The factors affecting stockout were the number of kanbans and the number of products. Kanban waiting times were impacted by the number of kanbans, number of stages, number of product types and the level of processing time variability.

## VITA

The author, Arvind R. Krishnappa, was born in Bangalore, India on October 8, 1972. He received his secondary education in that city, where he graduated from National Public School in 1990.

In August, 1990 he entered Regional Engineering College, Warangal, India. He received his Bachelor Of Technology in Mechanical Engineering in April, 1994. In the fall of 1994, he enrolled at West Virginia University for a MS in Industrial and Management Systems Engineering. He is currently employed as a Senior Operations Research Analyst at Cendant Corporation in New York City.