1972

A Simulation Study Of Multistage Inventory Policies

Howard Wesley Prout

A SIMULATION STUDY OF MULTISTAGE INVENTORY POLICIES

by

Howard Wesley Prout
School of Business Administration

Submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Faculty of Graduate Studies
The University of Western Ontario
London, Canada
August 1972

To my Mother and Father

Abstract

The development of inventory control theory began in the early 1900's when W. E. Camp published what is now known as the Wilson economic order quantity model. Since then, many models have been developed. Most of these models have been for single-stage operating systems.

Many industrial companies operate production facilities which produce a variety of products in several stages. Some components and raw materials are used in several products while others have unique applications. Also, some components may have direct market demands while others are used solely in the assembly of other products.

The Wilson EOQ model has been used to analyse the inventory requirements for many applications. Some applications violated the assumptions of the model. Such applications were usually made assuming that even though the underlying assumptions were violated, the solutions derived from the application of the model would not be appreciably in error.

The recent development of the requirements planning concept of inventory control has raised some questions regarding the advisability of using a statistical version of the EOQ model, such as an (s,S) policy, for all inventory items in a multistage operating system. The concept of requirements planning is to time-phase the assembly of components to facilitate the production of finished products

and meet the demand requirements for finished products. Requirements planning systems can incorporate any type of inventory policy, the selection is left to the user.

To test the relative economies of statistical and requirements planning inventory policies, an experiment implementing the two types of policies was made on a simulated factory. An (s,S) policy was used for the statistical application and a Lot-for-Lot policy was used for the requirements planning application. A third application incorporating an (s,S) policy for finished products and a Lot-for-Lot policy for components was also tested. The measure of effectiveness was the total operating costs. The inventory policies dominated the operating system so that workforce levels and production rates were governed by the inventory policies.

The requirements planning Lot-for-Lot applications resulted in lower total operating costs than either the combined or the statistical applications for operating cycles of one month and one week. When the labour adjustment costs were not included in the total costs, the combined statistical-requirements planning policies had the lowest inventory-related costs. The requirements planning Lot-for-Lot policy resulted in lower inventory-related costs than the statistical policy for the monthly operating cycle but the opposite held for the weekly operating cycle.

The results of this research show that the type of inventory policy which minimizes inventory-related costs does not necessarily minimize total operating costs. The length of the operating cycle is shown to influence the type of inventory policy which minimizes inventory-related costs. The cost of uncertainty from stochastic demand was found to vary substantially with the type of inventory policy selected. For inventory policies which can carry safety stock at several levels, the cost minimizing profiles were found to concentrate the safety stock at the finished products and raw materials levels.

Acknowlegements

The generation of a thesis requires resources from many people and in several forms, some of which are support, assistance, encouragement and understanding. The number of people who have provided resources of one kind or another in the preparation of this thesis is large.

Financial support has come from many sources. The Richard Ivey Doctoral Fund made generous contributions for the three years of the program. The Ontario Student Awards Program was a continuous source of financial support. The Associates' Fund provided financial assistance at various times throughout the program. Shell Canada Limited was also very generous in awarding a Shell Canada Research Support Grant which defrayed many of the expenses incurred in preparing this thesis. I gratefully acknowledge the assistance which these sources have provided. Without such financial support, I probably would not be fortunate enough to write these acknowledgements.

The School of Business Administration has provided an encouraging atmosphere and many facilities. The funding for the computations was provided by the Research Fund from the School. The faculty were a constant source of encouragement.

I am most grateful to my committee for their untiring assistance in guiding this project to its' fruition. Dr. C. Haeling von Lanzenauer and Dr. John R. M. Gordon were always available to discuss the progression of the project and offer useful comments. Dr. Robert R. Britney gave me the best counselling any advisor could give a candidate. He provided a constant source of encouragement by always looking for the positive.

Finally I want to express my sincere appreciation to my wife and family. Jean gave assistance whenever possible which often required her to make sacrifices which only she can value. My debt to her can never be repayed. There were two little people who were a constant source of encouragement although they didn't understand what was happening. Lynne and Bruce nobly accepted the strain imposed on them. The happy looks on their faces whenever I could spend some time with them made it all worthwhile.

Howard W. Prout

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter One

Introduction

1.1 The Role of Inventory Control in Operations Management

The importance of inventory control as a management function is illustrated by the percentage of total assets invested in inventories. Table 1.1 compares the estimated investments in inventory and in total assets of all Canadian industries. The ratios of inventory to total assets as presented in Table 1.1 indicate that there is a stable relationship between these investment classifications. When the focus of attention is narrowed to include only manufacturing industries, it can be seen in Table 1.2 that inventory is of even greater importance. The same stable relationship between inventory and total assets is evident, but at a higher ratio. A regression analysis using inventory investment as the dependent variable and total assets as the independent variable indicated an index of determination in excess of 0.91 both for the manufacturing industries and for all industries. This indicates a very stable relationship between the two variables.

The fact that inventory investment represents approximately 22 percent of the total assets for all

Table 1.1

Estimated Inventory and Total Assets

For All Industries By Quarters, 1968-1971

Year	Quarter	Inventory ($000,000's)	Total Assets ($000,000's)	Inventory / Total Assets (%)
1968	1	13,938	75,872	18.37
	2	13,978	77,496	18.03
	3	13,840	78,624	17.60
	4	14,482	80,309	18.03
1969	1	14,854	81,238	18.28
	2	15,060	83,346	18.06
	3	15,235	84,687	17.98
	4	15,731	86,341	18.21
1970	1	15,895	85,567	18.57
	2	15,934	89,358	17.83
	3	15,820	90,194	17.53
	4	15,773	90,876	17.35
1971	1	16,077	92,127	17.45
	2	16,184	93,904	17.23
	3	16,403	95,526	17.17
	4	16,583	96,930	17.10
			Average	17.80

SOURCE: Statistics Canada, Industrial Corporations Financial Statistics, Catalogue 61-003 Quarterly, Fourth Quarter 1971, Information Canada, Ottawa, May 1972, pp. 20-21.

Table 1.2

Estimated Inventory and Total Assets

For All Manufacturing Industries By Quarters, 1968-1971

Year	Quarter	Inventory ($000,000's)	Total Assets ($000,000's)	Inventory / Total Assets (%)
1968	1	8,188	36,337	22.53
	2	8,067	37,101	21.74
	3	8,126	37,666	21.57
	4	8,276	38,110	21.71
1969	1	8,426	38,191	22.06
	2	8,587	39,379	21.80
	3	8,378	40,074	20.90
	4	8,994	40,649	22.12
1970	1	9,146	40,831	22.39
	2	9,157	41,739	21.93
	3	9,173	42,169	21.75
	4	9,183	42,366	21.67
1971	1	9,302	42,854	21.70
	2	9,267	43,312	21.39
	3	9,321	44,041	21.16
	4	9,443	44,368	21.28
			Average	21.73

SOURCE: Statistics Canada, Industrial Corporations Financial Statements, Catalogue 61-003 Quarterly, Fourth Quarter 1971, Information Canada, Ottawa, May 1972, pp. 60-61.

manufacturing industries in Canada supports the importance of inventory control. If it is assumed that the average cost of carrying inventory is approximately 20 percent of the inventory value per year, then the cost associated with the nearly $10 billion invested in inventories by Canadian manufacturing industries is approximately $2 billion per year. If inventory control procedures can be improved by even one percent, this would increase national profits $20 billion per year. For these reasons, inventory control will remain an important function in industrial organizations.

Inventories can be divided into two basic categories which will be referred to as system dependent and system independent inventories. System dependent inventories are those required by the production or operating system to support the system in its operations. These include transit inventories, and work-in-process inventories. System independent inventories are discretionary and are held to assist the improvement of the organization's performance. Finished product inventories and safety stocks are typical system independent inventories.

The system dependent and system independent classifications of inventories are not as distinct as would be preferred but a reasonable guideline might be the amount of control or management which can be exercised with respect to the inventory. Generally, if the inventory can be managed independent of the functions in the system, it can be classified as system independent inventory. Otherwise it would be classified as system dependent inventory.

For the purpose of this research, it will be assumed that the organization's goal is to minimize total operating costs. Given this objective, the purpose of inventory control is to manage all inventories within the system to aid in minimizing the costs of the system.

The cost considerations in inventory control analysis are of two basic types. The first of these is an ordering or set-up cost. This type of cost is considered to be independent of the quantity of the inventory item requested. If the item requested is obtained from an external supplier, the cost is related to the work required to place the order. Ordering costs can take on a wide range of values dependent upon the availability of suppliers, the frequency of ordering and the administrative work required to release a purchase order.

Set-up costs are usually associated with the internal manufacture of inventory items. A typical example is an item which required machining. Before the item can be machined, the equipment must be readied for the machining operation. Once the equipment has been readied, it is frequently assumed that there is no limit to the number of units which can be produced without re-setting the equipment. Set-up costs are often given in labour hours which must then be converted into dollars at the appropriate hourly rate.

The other type of cost associated with inventory control is related to the holding of inventory. Inventory holding or carrying costs are frequently expressed as a percentage of the value of the inventory held. Harty, Plossl and Wight (1963) suggested that the most common range of inventory carrying costs used in industry was between 15 and 25 percent per year of the value of the inventory. In addition to the cost of capital, inventory carrying cost factors usually include allowances for obsolescence, spoilage, pilferage, light, heat, rent, taxes and insurance. While it may be argued that the costs associated with holding inventory do not vary in a strictly linear relationship with the value of inventory stocked, industry has generally accepted the linear relationship to be a reasonable approximation of the actual costs experienced.

Before leaving the topic of the role of inventory control in operations management, the function of inventory control must be considered in its proper perspective. As previously stated, the purpose of inventory control is to manage inventories to aid the organization in being more efficient, which for this study is assumed to be the minimization of total operating costs. This does not imply that the purpose of inventory control is to minimize inventory related costs. The minimization of inventory related costs would eliminate all system independent or discretionary inventories which under most conditions would increase total costs substantially.

Inventory control procedures influence both the marketing and the production functions. Marketing related costs such as costs associated with not being able to supply the market demand within an acceptable time period can be included in inventory control analysis. Factors related to production planning are seldom included in the analysis of inventory control. These factors can include changes in production rates, equipment utilization and adjustments in the workforce required to produce the desired amount of product. These production related costs can be very large, and as will be shown below, can be a determining factor in the selection of inventory control procedures.

1.2 A Discussion of Some Inventory Models

Analytical models for controlling inventories can be categorized many ways. For the purpose of this study, the first distinction will be based on the number of operating or production stages considered in the model.

A basic operating stage is defined to include a production facility and an inventory level, as shown in Figure 1.1. The operating stage consumes inputs from various supply sources, does some operations on the inputs and then stores the resulting product in inventory to supply a demand sink. The supply source may be external vendors in which case the production function is a purchasing operation and the resulting inventory is raw materials. The supply source could also be other operating stages and the demand sink could be either other operating stages or external markets.

Figure 1.1

Basic Operating Stage

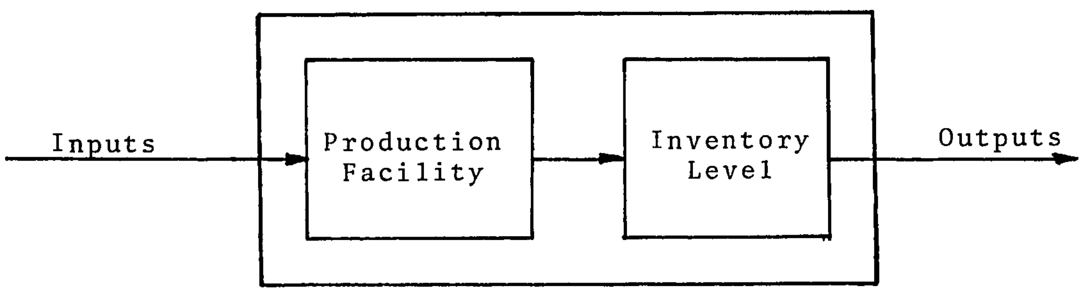
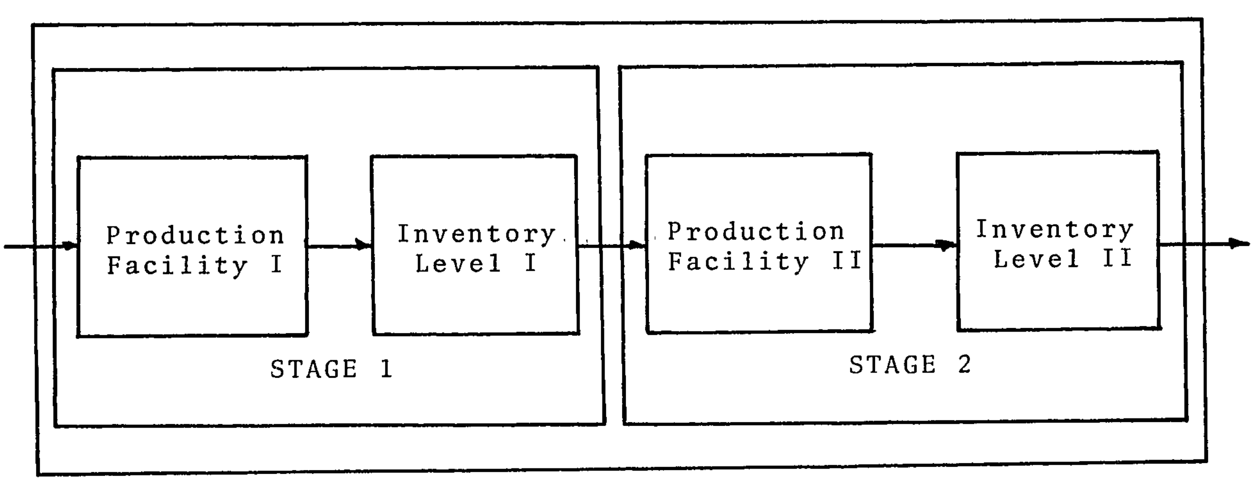


Figure 1.2

Multistage Operating System



Single stage inventory models consider only the consequences within the bounds of a basic operating stage. Multistage inventory models consider the effects of several basic operating stages linked together as shown in Figure 1.2, which for convenience is shown as a two-stage operation.

A second dimension for classifying inventory models is the number of products considered in the model. Some models consider one product in the analysis while others consider several products.

A third dimension for differentiating between inventory models is the nature of the demand patterns. Some models are capable of analyzing continuous demand conditions whereas others group the demands by time intervals and analyse the resulting intermittent or discrete demand.

The demand patterns can also be divided into deterministic and stochastic demand patterns, the fourth dimension. With deterministic demand patterns, the demand rates or quantities are known with certainty for as distant a planning horizon as is required for the analysis. Stochastic demand patterns have uncertain demand rates which require additional analysis compared to deterministic demand patterns.

This framework is diagramed in Figure 1.3 and is used in the discussion of inventory models in Chapter Two.

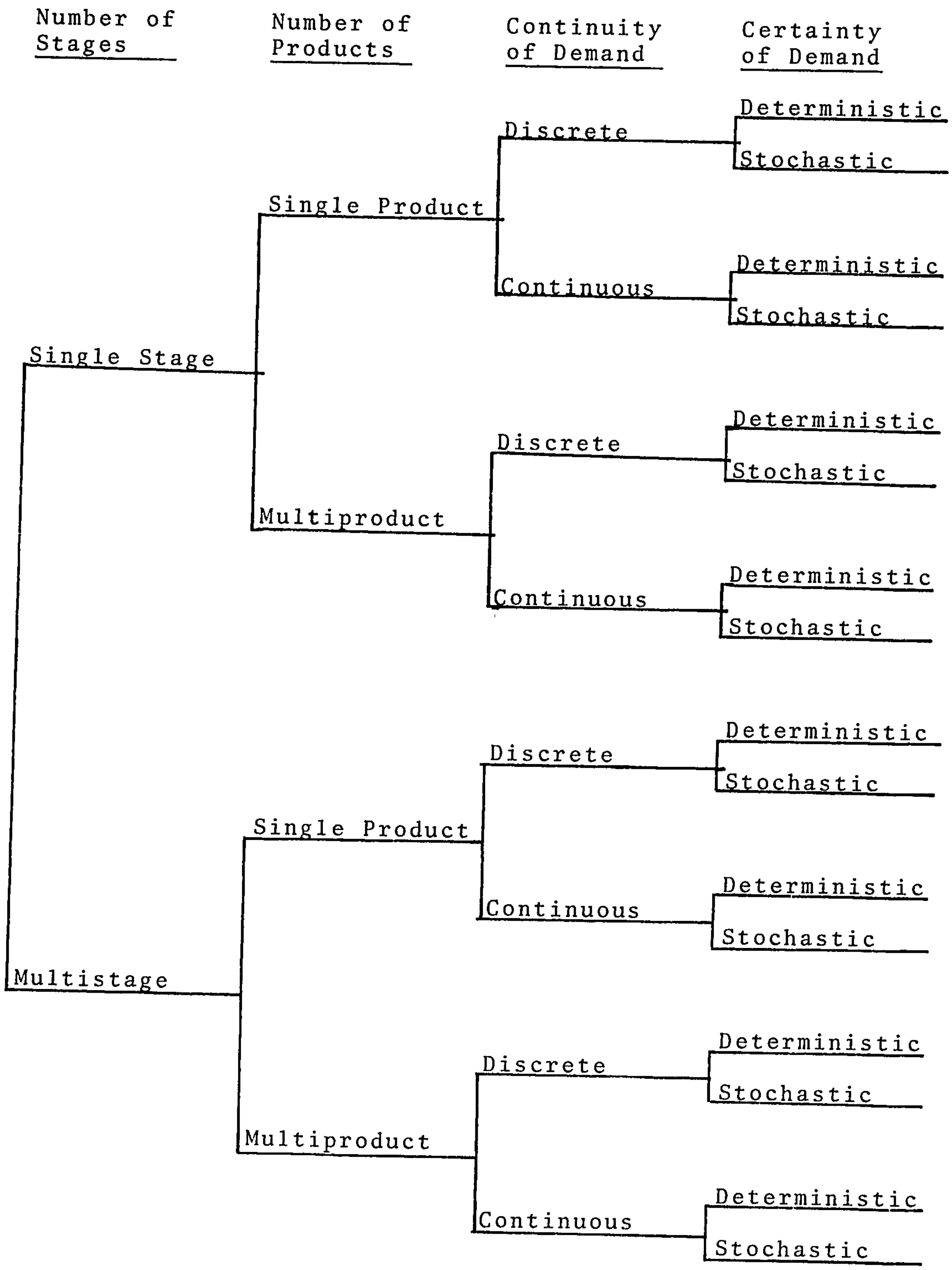
For the purpose of this research, the demand patterns discussed relate to the external market requirements for the finished product or products of the operating system. External demands for component products are not included in this analysis. The internal demands for products in multistage operating systems are incorporated in the multistage inventory models.

The analysis of multistage inventory control applications has been approached primarily by considering a multistage system to be a group of single-stage systems linked together. The solution procedure has been to consider each stage as a separate entity and to use single-stage inventory models for the analysis. Therefore both single-stage and multistage inventory models are discussed.

The Wilson economic lot-size model was one of the first inventory control models developed and it is a fundamental tool in inventory control. The "Wilson EOQ" model was developed for single-stage single-product continuous deterministic demand applications. Refinements to the model include the analysis of non-instantaneous receipt of the order quantity and price-breaks for large quantity purchases.

Figure 1.3

Framework for Categorizing Inventory Models



The Wilson EOQ model can also be used for stochastic demand applications by including safety stock in the calculation of the re-order point. Safety stock can be determined in several ways but the purpose of safety stock is to protect against demands greater than the mean expected demand during the replenishment phase of the inventory cycle.

Several methods have been developed to analyze single-stage single-product discrete deterministic demand conditions. Among the better known methods of this type are: (1) Least Total Cost; (2) Least Unit Cost; (3) Periodic Order Quantities; (4) Lot-for-Lot; and (5) The Wagner-Whitin Dynamic programming algorithm. These discrete lot-sizing techniques have become prominent in recent years with the development of requirements planning inventory systems.

Eilon (1962), Buchan and Koenigsberg (1963), and Buffa (1968) are amongst others who have developed models for analysing single-stage multiproduct continuous deterministic demand models. Buchan and Koenigsberg (1963) present a model which includes the analysis of cases in which capital and/or labour are restricted. The formulae for this analysis are relatively complex. The extension of this analysis to consider stochastic demands is not possible due to the method of formulating the problem.

Taha and Skeith (1970) have developed an inventory model to analyse multistage single-product continuous deterministic demand conditions. This model is based mainly on queuing theory dealing with sequential single-channel and cyclic queues.

Inventory models to analyse multistage multi-product discrete deterministic demand conditions have recently appeared. Zangwill (1966), Crowston, Wagner and Williams (1971), Love (1972) and Crowston and Wagner (1972) have been significant contributors. The models have examined acyclic networks. A major restriction of these models is that each facility can have only one successor.

Industrial organizations commonly operate multistage multiproduct systems which are subject to continuous stochastic demands. Requirements planning systems provide a framework for analysing a total system of this type. The concept of requirements planning segregates demands for products into independent and dependent classifications. Independent demands are the result of market forces. Dependent demands result from the demands of other inventory items. The rationale of requirements planning is to time-phase the production of component inventory items to facilitate the production of finished products and satisfy the market demand.

An alternative to requirements planning which has been widely promoted for many years is the use of the Wilson EOQ model for all inventory items in an operating system. This may be a misapplication of the Wilson model but the penalties from the misapplications may be relatively minor compared to the economic advantages gained from using the model.

1.3 Definition of this Research Project

The traditional use of the Wilson EOQ model for determining the lot-sizes of all inventory items is an obvious misapplication of the technique. However, it is a simple tool which may be relatively efficient even if some of the assumptions are violated.

In their special report for the American Production and Inventory Control Society, Plossl and Wight (1971) listed six areas requiring further research. The first three of these were:

1. Development of requirements planning simulator to demonstrate the obvious power of the technique.

2. Testing in a real world environment the various approaches to lot-sizing in a multi-level product structure to realize the benefits and minimize the problems of dynamic lot-sizing.

3. Develop methods for determining optimum safety stocks for lower level components in a requirements plan and the definition of service levels insured by these safety stocks.

Requirements planning is an inventory control concept that may have applications in small industrial companies. The basic concept of requirements planning is simple enough to be readily comprehended by industrial personnel without extensive training.

The major question that this research project attempts to answer are:

1. Given the relative simplicity of the Wilson model and of requirements planning systems, which method minimizes the total operating cost in a multistage multiproduct operating system?

2. What are the residual effects of using either method of controlling inventories?

The comparison of the use of these lot-sizing methods in a multistage multiproduct system will provide some guidelines for the selection of lot-sizing procedures in a multistage operating system. Such a comparison will also indicate whether or not there is support for the assertion Plossl and Wight (1971) made in their first recommendation listed above. This recommendation suggests that requirements planning has broad applications which have not been identified to date.

1.4 The Development of an Analytical Model

The flexibility of requirements planning as a concept of inventory control does not allow the development of an analytical model which has a universal solution procedure. The concept of requirements planning allows for the application of any inventory policy at any production stage to any inventory item. The underlying process of a requirements planning system is that the components within the system be time-phased to facilitate the assembly of finished products to meet the expected demand.

The objective for the organization stated above was the minimization of total operating costs over a planning horizon of a specified number of periods. Therefore the system can be considered to progress through a series of time periods such that during each period there are some inputs, some decision criteria, a transformation function and some outputs. The problem can be considered in terms of a sequential decision process model where the stages are the periods in the planning horizon, the states are the inventory status and the decision rules are the inventory control policies.

It will be shown in Chapter Three that even after substantial specialization, the solution procedure to this analytical model requires a trial and error search for a large number of variables to satisfy the optimization equation. A more direct approach is to use simulation to analyse the effectiveness of several inventory policies on a multistage production operation.

The use of simulation requires that a suitable model be used as the test vehicle. The "Production Control Project-DCIDE" simulation game was chosen as the model for testing the selected inventory policies. Some modifications were made to the game for this study. The simulation game has three production stages and 45 inventory items. These items are of four types: 15 finished goods, 7 subassemblies, 15 parts and 8 raw materials. The demands for the finished products can be either deterministic or stochastic. Stochastic finished product demands are generated by modifying the forecast demand by a standard unit normal random variate. Each finished product has a constant coefficient of variation, meaning that the standard deviation of the demand for each product is a proportional constant of the mean demand. The demands for finished products have base, trend, seasonal and random factors and the co-effecients fo these factorys are unique to each finished product. The simulation includes set-up costs for all assembled products, ordering costs for raw materials, production labour costs at both regular time and overtime rates, labour adjustment costs including hiring, firing and transfer costs, and backorder costs for demand not serviced directly from inventory. Backordered demands are retained until inventory is available to fill them. The production system is acyclic, indicating that components and raw materials can only be used for higher-order assemblies. The game requires 48 decisions per period: production or ordering decisions for each of the 45 inventory items, and workforce decisions for each of the three operating departments.

1.5 Description of Inventory Policies Tested

An inventory policy is defined as a set of decision rules which are applied to all inventory items in an operating system to determine the timing and quantities for ordering additional supplies. Several methods for determining the timing and quantities of inventory orders are discussed in Chapter Four.

Requirements planning inventory policies establish finished product requirements plans to supply the forecast demand for those products. The finished product requirements plans are then used to determine the ordering schedule for all component inventory items.

Statistical inventory policies determine the order points and order quantities individually for all inventory items based on the forecast demand and the variance of the demand for each item.

Combined statistical-requirements planning policies are conceptually the optimal inventory planning technique. Statistical policies are used to determine the finished product requirements. The finished products requirements schedule is then used to determine the ordering schedule for all component inventory items.

1.6 Research Methodology

The purpose of this research project was to analyse the effectiveness of requirements planning and statistical inventory policies on a multistage multi-product system. Therefore the decisions, with respect to the production quantities and size of the workforces, were determined from the inventory policies. The lot-sizing of the inventory policies were treated as the planned production quantities. The workforce in each department was adjusted to accommodate the planned production; workforce and production smoothing were not incorporated so the total impact of the inventory policies tested could be analysed.

The inventory policies were tested under two conditions: deterministic and stochastic demands for finished products. The difference in total cost between a policy tested under deterministic demands and under stochastic demands indicated the cost of uncertainty.

The total costs were compared for extended periods of application. The tests were run for the equivalent of 12 months to adjust the system to the policies and then the data for the comparisons was collected over the equivalent of the next 60 months. The total costs included labour adjustment costs which are not normally considered in inventory analysis, even though the selection of inventory policies has implications on labour adjustments. The labour adjustment costs were separated into components as was the utilization of labour.

The tests conducted under stochastic demands required the incorporation of safety stocks, however, the value of inventory held in safety stocks influenced the total costs. To compare inventory policies under stochastic demand, the minimum total cost must be found for the application of each policy.

An exhaustive evaluation of all reasonable combinations of safety stocks for each inventory item would have required an astronomic quantity of computer time. To reduce the analysis to an acceptable level, safety stock factors were determined by inventory levels, with each item at an inventory level using the same decision rules for the determination of the safety stock. Seven safety stock levels at each inventory level were considered. To increase the efficiency of computer usage, a "steepest ascent" search technique was used which located the minimum total cost in an average of approximately 100 runs per policy tested.

The "Production Control Project-DCIDE" simulation game has an internal forecasting routine which produces a perfect forecast. The actual demand for a finished product is equal to the forecast demand plus a random factor. When the coefficients of variation are set equal to zero, thereby eliminating the random factor, the forecast demand is equal to the actual demand.

A requirements planning inventory policy was tested under both deterministic and stochastic demand conditions. The deterministic test was used to develop a basis for a later comparison with the stochastic test, to determine the cost of uncertainty of demand. The stochastic demand test was of major importance. Its results were compared to the results from the statistical policies to determine the more appropriate policy for the test conditions.

The combined statistical-requirements planning policy was also tested under both deterministic and stochastic demand conditions. The deterministic test was used to determine the cost of uncertainty of demand for this policy. The stochastic tests were conducted for comparison with the results of the requirements planning and statistical policies tested under stochastic demand.

An (s,S) statistical inventory policy was tested under stochastic demand using two methods of determining the reorder point, s. The first used an (s,S) policy in which the order point was determined by the forecast demand during the lead time plus a multiple of the standard deviation of the demand. The order quantity was equal to the sum of the order point plus an economic order quantity less the stock on hand, plus, in the case of finished products, the current number of backorders.

The second statistical test was similar to the first except that the multiple of the standard deviation used was determined by the desired annual safety stock level. According to Brown (1967), this method should result in lower costs than the standard deviation approach.

The mean and standard deviation of total monthly costs were calculated for the inventory policies tested under stochastic demand conditions. These parameters were used to compare the results of the inventory policies tested to determine if there were statistically significant differences. The z-statistic was used for the statistical testing of differences. The tests were made on the total costs accumulated from the application of the inventory policies for a period of 60 months, which should have been sufficiently long to minimize most random effects.

A break-even analysis was used to develop formulae to compare the relative effectiveness of statistical and requirements planning inventory control policies. These formulae provide a guide in the selection of appropriate inventory control methods based on the length of the operating cycle.

The break-even analysis indicated that the total inventory related costs for a requirements planning Lot-for-Lot policy are sensitive to the length of the operating cycle. To test this sensitivity, the operating cycle was reduced to one week and the experiment was repeated. The reduction of the operating cycle to one week should approximate the assumption of the Wilson EOQ model more closely than the monthly cycle. The requirements planning policy will also require more set-ups, and therefore the statistical policy should have a lower operating cost.

1.7 Results and Analysis

The results and analysis of the research project are reported in Chapter Six. The total costs ranged from a low of approximately $27 million for the requirements planning policy under deterministic demand to a high in excess of $42 million for the statistical policy using safety stocks calculated from standard deviations.

The requirements planning policy tested under stochastic demand had a total cost of approximately $31 million. Thus the cost of uncertainty resulting from the difference between deterministic and stochastic demand conditions was approximately $4 million, a premium of nearly 15 percent. The major components of this difference were $2.5 million in labour adjustment costs and $1.0 million in inventory carrying costs.

The application of the combined statistical-requirements planning policy under deterministic demand resulted in a total cost of approximately $31 million which was approximately 15 percent more than the application of requirements planning policy under deterministic demand. This policy was expected to be less costly than the comparative requirements planning policy. The total cost of the combined policy under stochastic demand was also slightly more than $31 million indicating that the cost of uncertainty is very low for a combined policy.

The application of the statistical policies under stochastic demand resulted in much higher costs than the comparable requirements planning and combined policies. The minimum total cost using the standard deviation approach to safety stocks was approximately $42 million and the minimum total cost using the service level approach was approximately $41 million. These costs are more than 30 percent higher than the comparable requirements planning and combined policies. The two statistical policies had high labour adjustment costs, approximately $11 million and $10 million respectively. The service level approach did result in lower total costs than the standard deviation approach but not of the order predicted by Brown (1967).

The regular time hours were similar for all policies but the overtime hours were higher for the requirements planning and combined policies. Overtime was used to overcome the integer restrictions on the workforce level. If the desired production required a workforce level which was closer to the lower integer than the upper integer, the actual workforce employed was the lower integer number and, providing the materials were available for the desired production, overtime was used to produce the assemblies which would have been done by the "fractional" employee.

The set-up hours for the requirements planning policies were nearly twice the number needed for the statistical and combined policies. The EOQ lot-sizing was the reason for this result. There was very little difference in total set-up hours between the statistical and combined policies, indicating that almost all the savings in set-up hours resulting from the application of EOQ's were realized at the finished products assembly stage.

The idle-time hours were much lower for the deterministic demand applications than for the stochastic demand applications with the exception of the combined policy. The idle-time hours for the deterministic applications were due to integerizing the workforce upwards. The idle-time hours for the stochastic demand applications were primarily due to material shortages making assembly of the desired production lot-sizes impossible.

The run-time hours were similar for all policy-demand treatment combinations. This was expected since approximately the same quantity of all products was produced regardless of the inventory policy or the demand treatment applied.

The tests for significant differences between the inventory policies tested under stochastic demand indicated that the total cost using the requirements planning policy or the combined policy was significantly less than the total costs of either the standard deviation or the service level statistical policies. The level of confidence in these differences was effectively 1.0, indicating certainty. A comparison of the total costs of the two statistical policies indicated that the confidence level of the observed difference was only 0.655. The statistical policies had much larger variances in the monthly total costs than did the requirements planning policy under stochastic demand. Of the two statistical policies, the service level approach had the lower variance in monthly total costs.

A comparison of the total costs from the various tests was made excluding the labour adjustment costs. Under these conditions, the combined statistical-requirements planning policy resulted in a slightly lower total cost than the requirements planning policy under deterministic demand. Comparing the stochastic demand policies, excluding labour adjustment costs, indicated that the combined policy had a much lower total cost. However, under those conditions, the service level approach had a marginally higher total cost than the standard deviation approach, which is contrary to the expectations suggested by Brown (1967).

The safety stock profiles in terms of the amount of safety stock at each level at the minimum total cost combinations were similar for all policies tested under stochastic demand. The safety stock levels were largest at the finished products and raw materials levels. This supports Plossl and Wight (1970). The large finished product safety stocks protect against the high cost of stockouts, whereas, the large safety stocks of raw materials provide production flexibility.

The tests conducted using the weekly operating cycle confirmed the predictions of the break-even analysis developed in Chapter Two. The total operating costs less labour adjustment costs were less for the statistical policy than for the requirements planning policy under stochastic demand. This is opposite to the results obtained from the monthly operating cycle tests.

The simulation required an extensive amount of computer time due to the large number of runs required to find the minimum total costs for the policies tested under stochastic demand. Each run of 72 months took approximately one minute of CPU time on a CDC6400 computer. The search procedures required between 60 and 120 runs to find the minimum total costs per policy. A total of over 300 runs were made, including the initial runs used to test the computer programs.

1.8 Conclusions

A discussion of the implications of this research is given in Chapter Seven. The influence of inventory policies on labour adjustment costs was shown to be substantial in the tests conducted. The interaction between the inventory policies, the smoothing of production rates and labour adjustments indicates why the minimizing of inventory costs without the consideration of other functions may be insufficient for the total organization.

The statistical policies led to intermittent production which required large changes in the workforce levels. Some requirements planning policies may also lead to intermittent production but the Lot-for-Lot requirements planning policies used in this experiment performed much better than the statistical policies. This was due to the direct relationship between the demand functions and the workforce levels. The demand functions had relatively smooth changes from period to period thereby requiring limited changes in the workforce.

The requirements planning policies using the Lot-for-Lot, lot-sizing method resulted in much lower costs than the statistical policies, supporting the proponents of the requirements planning method of inventory control in multistage multiproduct systems. The major cost difference between the two approaches was in the labour adjustment costs and the inventory carrying costs. The relatively small difference in total costs between the safety stock and service level approaches to statistical inventory policies did not support Brown (1967). Apparently, the service level approach is primarily applicable to finished product inventories.

The "U" shaped safety stock profiles which minimized the total costs for all policies tested under stochastic demand is supportive of Plossl and Wight (1970). The relatively high costs of stockouts and the increased production flexibility from large raw material stocks make this an economical strategy.

Improvements in inventory control procedures must be considered in terms of the effect on the total system. The implementation of the combined statistical-requirements planning policy for controlling inventories resulted in lower total inventory-related costs than the requirements planning policy. However, the inclusion of the labour adjustment costs reversed the relative performance of the two policies indicating that the policy resulting in the lowest inventory-related costs did not result in the lowest total operating costs.

The length of the planning or operating cycle influences the choice of an inventory policy. The use of breakeven models indicated that requirements planning Lot-for-Lot methods are more economical for long cycles and the statistical models using EOQ lot-sizes are better for short operating cycles if only inventory-related costs are considered. The emperical results support the break-even analysis.

The "shape" of the system could also be an important criterion in the selection of a multistage-multiproduct inventory policy. For diverging systems where the number of inventory items increases progressively toward the final stage, the commonality of component items could favour the use of statistical inventory policies. Converging systems have a reduction of inventory items from the initial to the final stage in which case the demands for components are likely to indicate a definite dependent relationship. This would suggest the use of a requirements planning inventory policy.

Future research should be directed toward analysing the influence of the "shape" of the system to determine if the above assertions are valid. Another area which requires further exploration is the influence of the type of demand patterns on the selection of an inventory policy. Given the available technology, multistage-multiproduct system are too complex for an analytical evaluation forcing the researcher to use simulation as a research tool. Simulation is a very effective technique but it can be very costly; however, the alternatives would be much more difficult to administer and the research costs would probably be much greater.

Chapter Two

The Rationale of this Research Project

2.1 The Needs of Industry

The results of a previous survey[1] of 150 manufacturing companies in Southwestern Ontario indicated that one-third of these companies employed less than 50 manufacturing personnel and sixty percent of the companies had a productive workforce of under 100 people. Only ten percent of the sample employed more than 400 people in the production operation. The average manufacturing budget of this upper ten percent was approximately $10 million per year.

If the respondents of the survey were typical of Canadian industry, few companies are of sufficient size to employ personnel capable of comprehending the more sophisticated inventory models available. Plossl and Wight (1971) reported that they did not know of any companies using the Wagner-Whitin dynamic programming algorithm for determining economic lot sizes, even

---

1. Prout, H. W. "The State of the Art of Production Management - A Survey", Unpublished term paper, School of Business Administration, The University of Western Ontario, 1971.

though that algorithm was published in 1958.

The apparent needs of industry are for simple models which are easy to comprehend and which accomplish most of the economic benefits of the more sophisticated models with a low level of effort. Plossl and Wight (1971) report that even some of the most sophisticated industrial inventory control groups are using the Wilson model to determine economic order quantities for all inventory items.

Recently there has been a growing interest in requirements planning. Orlicky (1970) referred to requirements planning as the "Cinderella" of inventory control, claiming that it is a superior tool which cannot fail to flourish. Several articles relating to requirements planning have been published in the last few years. These articles have increased the awareness of requirements planning as an inventory control method, but at the same time, there is some skepticism about the method and some feeling that it must be very difficult to comprehend and to implement.

A review of some inventory models which are of interest in the discussion of inventory control in multistage multiproduct operating systems is presented in the following section.

2.2 A Review of Some Inventory Control Models

Several inventory control models have been developed and reported by previous researchers. Some of these models are relevant to this research project, while other models are less relevant. A framework for classifying the various inventory models was developed in Chapter One to group the available models into categories which are meaningful to this study.

The focus of this research project is the control of inventories in multistage multiproduct systems. Two relevant dimensions of the framework are the number of operating stages and the number of products included in the inventory models. The demand for the products of the system from external sources forms the remaining two dimensions of the framework. The demand for products may be considered as being either continuous or discrete, the latter being the accumulated demand for a specified time interval. In addition, the assumptions underlying inventory models may restrict the use of the models to either deterministic demand applications, in which case the demand for products is known with certainty for as distant into the future as is required for planning purposes, or to stochastic demand applications, in which case the mean demand for products can be forecast but the actual demand will vary above and below the forecast mean demand.

Multistage inventory analysis is frequently approached by using a set of single-stage single-product inventory models. Some of these relatively simple models are reviewed prior to the discussion of multistage inventory models.

2.2.1 Single-Stage Single-Product Continuous Deterministic Demand Inventory Models

The concept of a most economical lot-size for manufactured or purchased inventory items ordered in batches has been recognized since the early 1900's. Harty, Plossl and Wight (1963) state that the first recorded application of economic lot-sizing occurred at the H. H. Franklin Manufacturing Company in Syracuse, New York in 1912. According to Harty, Plossl and Wight (1963):

> One of the first published formulas is that described by Ford W. Harris in 1915; this formula was used at the Westinghouse Electric and Manufacturing Company. After that, very little was published on the subject of economic order quantities until W. E. Camp presented his well-known 'Camp Formula' in 1922. Mr. Camp was evidently not aware of the existence of the Harris formula because he said, 'No one has attempted to derive a formula for determining the production order quantity'. A few years later, R. H. Wilson...evidenced no knowledge of the existence of a formula for calculating EOQ's in his article, 'A New Method for Stock Control'.

Wilson later became a strong advocate of economic order quantities and that which Harty, Plossl and Wight (1963) refer to as the "Camp Formula" is frequently called the "Wilson Model" in the literature, and is developed in Appendix I.

The Wilson model is frequently referred to as the EOQ model and the resulting Q as the EOQ. The EOQ model is an integral part of several inventory policies which are described in detail in Chapter Three.

2.2.2. Single-Stage Single-Product Continuous Stochastic Demand Inventory Models

When the demand during the lead time is at a constant rate, the reorder point can be defined as the stock level required to service the demand during the lead time. A more typical case is when the demand is represented by an average rate with some variation in the actual demand rate from period to period. Under these conditions the reorder point consists of two terms: (1) the mean demand during the lead time, and (2) some safety stock to protect against higher than average demand during the lead time.

The amount of safety stock desired may be determined in a variety of ways. Four possibilities are: (1) fixed quantities; (2) time-based quantities; (3) quantities determined by some number of standard deviations in demand over the lead time, and;

(4) quantities related to the desired annual service level.

Fixed quantity safety stocks imply that the reorder point will be set at a quantity equal to the mean demand over the lead time plus an additional quantity which may be determined arbitrarily or analytically. This can be an effective method of determining safety stocks if the variation in demand over the replenishment lead time is small.

The use of time-based quantities for safety stocks is similar to the fixed quantity method since neither method relates to the actual variation in the demand for the inventory item. A time-based safety stock criterion implies that a replenishment order be issued whenever the stock on hand declines to a level equal to the mean forecast demand during the lead time plus some number of periods of mean demand. The term "safety-time" is also used to refer to time-based safety stocks.

A safety stock level related to the variation in the demand over the lead time can be based on the standard deviation of the demand function during the replenishment lead time. If the demand during the lead time is distributed according to a continuous probability distribution, a reorder level equal to the mean demand plus "z" standard deviations of demand can be used, where the value of "z" is dependent upon the desired level of customer service. A service level can be defined as the ratio of the number of units delivered immediately from stock, to the total number of units demanded. Define P to be the service level where:

$$(2.1)\quad P = \frac{\text{number of units delivered immediately from inventory.}}{\text{total number of units demanded.}}$$

If the desired service level during the lead time is selected at some value, the amount of safety stock in terms of the number of standard deviations of demand can be adjusted to provide that average service level. If the demand during the lead time is normally distributed, a safety stock equal to two standard deviations of demand variation will satisfy the demand during the lead time 97.5 percent of the time. Similarily for a reorder level of the mean demand plus three standard deviations, the probability of serving the demand during the replenishment lead time is 0.999.

The time period over which the service levels are measured has not been defined at this point and therefore can be chosen to be any convenient time duration. For the remainder of this study the time duration associated with service levels will be one year.

The use of the mean plus a constant number of standard deviations of demand during the lead time can result in substantially different annual service levels for the various inventory items. A corporate objective may be to provide a uniform service level for all finished products. Brown (1967) gives the following description of the procedure for calculating safety stocks to satisfy a given annual service level:

> ...we can represent the annual demand by S pieces per year, and the order quantity by Q pieces per order. The standard deviation of forecast errors is σ pieces per lead time. We want to find the safety factor k that will be used to compute the safety stock kσ. The expected quantity that will be backordered per order cycle is σE(k), where the function E(k) is the partial expectation. In the course of a year there will be S/Q orders, or opportunities for a shortage to occur. Therefore, the total annual quantity backordered is σE(k) S/Q, and we require that this be some fraction (1-P) of the annual sales. The fraction P is the normal way of expressing service, meaning the probability of filling the order. The fraction P will be treated as a policy variable. Therefore σE(k) S/Q = (1-P)S, or the safety factor satisfies the relationship
>
> $$E(k) = \frac{Q}{\sigma} (1-P).$$
>
> We know all the factors on the right side of the equation, so that we can compute the value of the partial expectation function.

The partial expectation as used by Brown (1967) is given by the formula:

(2.2) $$E(x) = \int_{x}^{\infty} (t-x)\, p\,(t)\, dt \;.$$

where x is a standard unit normal variable and p(t) is the height of the density function of the standard unit normal probability distribution at t standard deviations from the mean. An approximation of the partial expectation also given by Brown (1967) is:[1]

$$E(x) = -0.5x - \frac{9.8575631}{8.189133+x^2} + \frac{172025.85+6998.8869x^2}{107496.82+638.3668x^2+x^4} \;.$$

The service level approach normalizes the protection against stockouts to a yearly basis from a variable time base dependent upon the ratio of the order quantity to the yearly demand. Service levels are probably more meaningful to practitioners than the more esoteric concept of stockout costs. Brown (1967) suggests that this approach is more economical than using the multiple of the standard deviation approach.

2.2.3 Single-Stage Single-Product Discrete Deterministic Demand Inventory Models

Several models have been developed for analysing the single-stage single-product discrete deterministic demand conditions. Among the better known models in

---

1. Brown credits Mr. P. F. Strong of Arthur D. Little Inc., Cambridge, Massachusetts with the derivation of the approximation formula.

this category are: (1) Least Total Cost; (2) Least Unit Cost; (3) Periodic Order Quantities; (4) Lot-for-Lot, and; (5) the Wagner-Whitin dynamic programming algorithm. Discrete lot-sizing techniques do not require the assumption of a uniform demand rate. Orders for replenishment stocks of inventory items are issued as they are required to satisfy the actual demand. The objective of discrete lot-sizing models is to determine how the period requirements should be combined into lots.

The Least Total Cost model assumes that the total costs related to the lot-size decision will be minimized when the ordering costs are equal to the inventory carrying costs. As a matter of interest, it should be noted that the resulting Q from the Wilson model equates the ordering costs and the inventory carrying costs. A trial and error solution method is used in the Least Total Cost approach of lot-sizing to aggregate future period demands into one lot until the total cumulative carrying cost is most nearly equal to the set-up cost. Since interval demands are considered, the inventory carrying costs and the set-up costs can seldom be equated but a reasonably close approximation can usually be achieved. An example of Discrete Requirements is given in Figure 2.1. Figure 2.2 gives an example of Least Total

Figure 2.1

Example of Discrete Requirements

Month	Quantity	Month	Quantity
1	500	7	500
2	600	8	400
3	700	9	300
4	800	10	200
5	700	11	300
6	600	12	400

Unit Cost - $5.00 each

Set-Up Cost - $50.00

Inventory Cost - 2% per month

$$EOQ = \sqrt{\frac{2AS}{Ci}} = \sqrt{\frac{2x6000x50}{12x0.02x5}} = 707 \text{ units.*}$$

* See Appendix I

Figure 2.2

Example of Least Total Cost Lot-Sizing

Month Req'd	Net Req't	Cumul. Lot-Size	Excess Inv.	Months Carried	Carrying Costs Unit	Carrying Costs Cum.	Set-Up Costs	Total Costs
1	500	500	0	0	0	0	50	50
2	600	1100*	600	1	60	60	50	110
3	700	700	0	0	0	0	50	50
4	800	1500*	800	1	80	80	50	130
5	700	700	0	0	0	0	50	50
6	600	1300*	600	1	60	60	50	110
7	500	500	0	0	0	0	50	50
8	400	900*	400	1	40	40	50	90
9	300	1200	300	2	60	100	50	150
9	300	300	0	0	0	0	50	50
10	200	500*	200	1	20	20	50	70
11	300	800	300	2	60	80	50	130
11	300	300	0	0	0	0	50	50
12	400	700*	400	1	40	40	50	90

* Indicates adopted lot-sizes.

Required Receipt Schedule

Month	Net Requirement	Required Receipt
1	500	1100
2	600	0
3	700	1500
4	800	0
5	700	1300
6	600	0
7	500	900
8	400	0
9	300	500
10	200	0
11	300	700
12	400	0

Note: The lot-size of 500 in month 9 could have been 800 as the total cost of 70 in month 10 and 130 in period 11 are equally distant from the optimal total cost of 100. The smaller lot-size was arbitrarily adopted in consideration of the lot-size in month 11.

Cost lot-sizing method as applied to the discrete requirements of Figure 2.1. The Least Total Cost lot-sizing method indicated that the requirements be grouped into six lots, each incorporating the requirements of two months.

The Least Unit Cost technique attempts to determine the economic lot-size on the basis of the least cost-per-piece. The cumulative carrying cost and the set-up costs are added and the total divided by the number of pieces in the lot to get a unit cost. The lot-size is set at the quantity which gives the least unit cost. The application of the Least Unit Cost lot-sizing method to the discrete requirements of Figure 2.1 is shown in Figure 2.3. In the example the requirements for the 12 months are grouped into 8 lots.

Gorham (1968) analysed the Least Total Cost and Least Unit Cost approaches and concluded that the Least Unit Cost approach was not as economical as Least Total Cost. Plossl and Wight (1970) do not recommend the use of the Least Unit Cost technique emphasizing that the Black & Decker Manufacturing Company has also compared the two techniques in simulations and confirmed the results of Gorham (1968).

The Periodic Order Quantity technique is developed from the Wilson Economic Order Quantity (EOQ). The EOQ is calculated and then divided by the annual requirements

Figure 2.3

Example of Least Unit Cost Lot-Sizing

Month Req'd	Net Req't	Cum. Lot-Size	Excess Inv.	Months Carried	Carring Cost Unit	Carring Cost Cum.	Set-Up Costs	Total Costs	Cost per Unit
1	500	500	0	0	0	0	50	50	0.1000
2	600	1100*	600	1	60	60	50	110	0.1000
3	700	1800	700	2	140	200	50	250	0.1388
3	700	700*	0	0	0	0	50	50	0.0714
4	800	1500	800	1	80	80	50	130	0.0866
4	800	800*	0	0	0	0	50	50	0.0625
5	700	1500	700	1	70	70	50	120	0.0800
5	700	700*	0	0	0	0	50	50	0.0714
6	600	1300	600	1	60	60	50	110	0.0846
6	600	600*	0	0	0	0	50	50	0.0833
7	500	1100	500	1	50	50	50	100	0.0909
7	500	500	0	0	0	0	50	50	0.1000
8	400	900*	400	1	40	40	50	90	0.1000
9	300	1200	300	2	60	100	50	150	0.1250
9	300	300	0	0	0	0	50	50	0.1667
10	200	500*	200	1	20	20	50	70	0.1040
11	300	800	300	2	60	80	50	130	0.1625
11	300	300	0	0	0	0	50	50	0.1667
12	400	700*	400	1	40	40	50	90	0.1285

* Indicates adopted lot-sizes.

Required Receipt Schedule

Month	Net Requirement	Required Receipt
1	500	1100
2	600	0
3	700	700
4	800	800
5	700	700
6	600	600
7	500	900
8	400	0
9	300	500
10	200	0
11	300	700
12	400	0

Note: The lot-sizes in months 1 and 7 could have been 500 due to equal costs per unit at the adjacent lot-sizes. In both cases the larger lot-size was arbitrarily adopted.

to give the average number of years demand represented by the lot-size. The resulting number of years is then adjusted to the nearest number of integer periods. The demand requirements are accumulated for that number of periods and the result is the Periodic Order Quantity lot-size. An example of the application of the Periodic Order Quantity lot-sizing method to the discrete requirements of Figure 2.1 is given in Figure 2.4. The Periodic Order Quantity of 1.414 months is integerized to one month, and therefore the requirements of each month are the adopted lot-sizes.

In a special report prepared for The American Production and Inventory Control Society, Plossl and Wight (1971) noted that 3 of the 8 companies reported used Periodic Order Quantities to determine lot-sizes. This was the most frequently used lot-sizing method, the next was EOQ's.

The Lot-for-Lot approach has several advantages, particularily for assemblies, sub-assemblies and components which have very low set-up costs. This approach sets the lot-size equal to the quantity demanded in each time period. Figure 2.5 shows the lot-sizes of the Lot-for-Lot, lot-sizing method for the monthly requirements of Figure 2.1. The lot-sizes adopted from

Figure 2.4

Example of Periodic Order Quantity Lot-Sizing

EOQ = 707 units*

Total Annual Demand = A = 6000 units/yr.

Periodic Order Quantity $= \frac{EOQ}{A} = \frac{707}{6000/\text{year}} = 0.1178$ year

$= 1.414$ months

$= 1$ month (integerized)

Therefore:

Required Receipt Schedule

Month	Net Requirement	Required Receipt
1	500	500
2	600	600
3	700	700
4	800	800
5	700	700
6	600	600
7	500	500
8	400	400
9	300	300
10	200	200
11	300	300
12	400	400

* See Figure 2.1

Figure 2.5

Example of Lot-for-Lot, Lot-Sizing

Month	Net Requirement	Required Receipt
1	500	500
2	600	600
3	700	700
4	800	800
5	700	700
6	600	600
7	500	500
8	400	400
9	300	300
10	200	200
11	300	300
12	400	400

the application of the Lot-for-Lot method are always equal to the demand of one period. The lot-sizes adopted using this method are identical to those adopted using the Periodic Order Quantity method for this example. Plossl and Wight (1970) suggest that the Lot-for-Lot approach tends to minimize the grouping of requirements from lower-level inventories, thereby leveling the production rates in the manufacturing facilities.

Wagner and Whitin (1958) developed a dynamic programming algorithm for determining economic lot-sizes. This is a sophisticated technique which explores the various alternatives in setting order quantities to minimize the total costs over the planning horizon. Plossl and Wight (1970) noted that no one at a workshop they conducted knew of any company using the Wagner-Whitin technique. In their 1970 paper, Plossl and Wight stated that the Wagner-Whitin algorithm "...may find use in 'fine tuning' a system in the future when its potential economies warrant the extra complexity".

Figure 2.6 shows the calculations for determining the optimal lot sizes of the monthly requirements of Figure 2.1 using the Wagner-Whitin algorithm. It is interesting to note that the optimum plan determined by Wagner-Whitin algorith required 9 set-ups as compared to 6 for the least Total Cost lot-sizing method and 8 set-ups for the Least Unit Cost lot-sizing method.

Figure 2.6

Example of Wagner-Whitin Lot-Sizing

Month Req'd	Net Req't	Ordering Cost	Calculated Costs	Minimum Cost	Optimal Policy
1	500	50	50	50	1
2	600	50	50+50=100 50+60=110	100	2
3	700	50	100+50=150 50+50+70=170	150	3
4	800	50	150+50=200 100+50+80=230	200	4
5	700	50	200+50=250 150+50+70=270	250	5
6	600	50	250+50=300 200+50+60=310	300	6
7	500	50	300+50=350 250+50+50=350	350	7 or 6,7
8	400	50	350+50=400 300+50+40=390 250+50+50+80=430	390	7,8
9	300	50	390+50=440 350+50+30=430 300+50+40+60=450	430	8,9
10	200	50	430+50=480 390+50+20=460 350+50+30+40=470	460	9,10
11	300	50	460+50=510 430+50+30=510 390+50+20+60=520	510	11 or 10,11
12	400	50	510+50=560 460+50+40=550 430+50+30+80=590	550	11,12

Required Receipt Schedule

Month	Net Requirement	Required Receipt
1	500	500
2	600	600
3	700	700
4	800	800
5	700	700
6	600	600
7	500	900
8	400	0
9	300	500
10	200	0
11	300	700
12	400	0

Single-stage single-product interval deterministic demand inventory control models are used extensively as sub-models in the requirements planning method of controlling inventories which will be described below.

2.2.4 Single-Stage Multiproduct Continuous Deterministic Demand Inventory Models

Inventory models to determine the lot-sizes for multiple products in a single stage system have been developed by several authors including Buchan and Koenigsberg (1963), Buffa (1968), and Eilon (1962). Buchan and Koenigsberg (1963) point out that if there are no restrictions on machine capacity, capital invested in inventory and/or warehouse space, then the multiple products case can be treated as several single-product cases. They suggest that such assumptions are not valid in most cases of practical interest, and therefore, some modification in lot-sizes must be made to accommodate the binding restrictions. Their model is developed in Appendix II.

2.2.5 Single-Stage Multiproduct Continuous Stochastic Demand Inventory Models

No literature relating to multiple product single-stage continuous stochastic demand inventory models was found. This was expected given the restrictions required to develop the deterministic demand model. In that model, the products are produced in cycles, each product having a production run in each cycle. The analysis of stochastic demands would require the determination of safety stocks for each item. However, these safety stocks could not be used to determine a useful order point. The lot-size quantities developed in Equation (II-11) assume a fixed number of cycles per year with each product requiring a given fraction of each cycle. There is no flexibility for production runs of quantities other than those determined by the equation.

With the current method of formulating the multiproduct single-stage continuous deterministic demand inventory model, there is a little prospect of the development of a similar stochastic model. A need for such a model may exist in warehousing operations where the production function is the handling of orders. Some divisions of large integrated companies may also act as single-stage operations producing several products.

A single-stage multiproduct continuous stochastic demand inventory model would be useful for analysing multistage multiproduct operations. The model could be applied sequentially to the operating stages of a multistage system.

2.2.6 Multistage Single-Product Continuous Deterministic Demand Inventory Models

Taha and Skeith (1970) have developed an inventory model to analyse multistage single-product continuous deterministic demand conditions. They assume that the product moves between stages in a serial fashion.

> In this model, it is assumed that unfilled demand is backlogged. Overproduction (within limitation) is allowed at the different stages so that each stage i may produce k_i batches once every k_i cycles. The decision variables for each stage i are the number of batches per run k_i, the batch size Q_i, and the shortage quantity of the finished product Q_s. These are determined by minimizing the total cost per unit time which includes the inventory holding costs, the shortage cost, and the set-up cost. The problem is also considered for the case with storage constraints at the different stages.

The analysis used by Taha and Skeith (1970) is based mainly on queuing theory dealing with sequential single-channel and cyclic queues. The model yields a detailed feasible production schedule and the optimal lot-sizes for each stage. They state that they intend to extend the model so that the restriction on the number of lots produced at each stage will be dictated by balancing the successive stages.

2.2.7 Multistage Multiproduct Discrete Deterministic Demand Inventory Models

Zangwill (1966) developed a model for analysing a deterministic multiproduct, multifacility, multiperiod production and inventory planning problem. He considers the multifacility model to be essentially a linking together of single facility models. The linking is in the form of an acyclic network indicating that there is an echelon of stages and that materials can only flow from lower level stages to higher level stages.

The model considers concave production costs, which can depend upon the production in several different facilities, and piecewise concave inventory costs. Backlogging of unsatisfied demand is permitted. Algorithms are developed for a parallel facilities case and for a series facilities case.

Crowston and Wagner (1972) have developed two dynamic lot-size models for multistage assembly systems. The application of these models is restricted in that each facility may have any number of predecessors but only one successor. This type of system could be called a "converging system" since the number of units decreases from stage to stage. Their first model is a dynamic programming algorithm which is indifferent to the assembly structure, provided that the above conditions are met. Their second model is a Branch and Bound algorithm intended for cases where the number of periods is large but the structure is near serial.

The models presented by Crowston and Wagner (1972) are direct extensions of the model discussed by Love (1972) for the special case of facilities arranged in series.

In an earlier paper, Crowston, Wagner and Williams (1971) prove that for a multistage assembly system characterized by the restriction that each stage can have at most one immediate successor but any number of immediate predecessors, an optimal set of lot-sizes exists such that the lot-size at each facility is a positive integer multiple of the lot-size at its successor facility. This is reassuring, otherwise the production of higher level stages could be restricted by insufficient materials.

The extension of these multistage multiproduct discrete deterministic demand inventory models to consider the continuous stochastic condition is unlikely. The solution procedures require that the external demands for components be known in discrete quantities per time period for the required planning horizon. The use of an expectation operator could be used to convert the stochastic demands to equivalent deterministic demands, but the solution procedure would be very complex.

2.2.8 Multistage Multiproduct Continuous Stochastic Demand Inventory Models

Multistage, multiproduct continuous stochastic demand is a most important combination since it relates to the conditions faced by most industrial organizations. A traditional approach to inventory control for these conditions has been to consider each inventory item as a separate entity and use the Wilson EOQ model for each inventory item. A recent development is the concept of material requirements planning systems, which is applicable to multistage multiproduct continuous demand operating systems. In spite of the necessity for a model to analyse these conditions, requirements planning appears to be the only analytical method applicable to these conditions.

The literature pertaining to requirements planning can be very confusing on the first reading. The number of articles that have been written on requirements planning systems is relatively small. The length of these articles precludes an in-depth discussion of the multi-stage time-phasing of component lot-sizes to facilitate the production of finished products. Therefore the subject matter of most articles about requirements planning is the comparison of EOQ lot-sizing to discrete lot-sizing for single stage single-product discrete deterministic applications. The examples given tend to be extremely biased in favour of discrete lot-sizing and grossly violate the assumptions of the Wilson model.

Requirements Planning is based on the concept that the demand functions for inventory items are of one of two types: Independent or Dependent. Orlicky (1970) states:

> Demand for a given inventory item is considered INDEPENDENT when such demand is unrelated to the demand for other items, particularilly higher level assemblies or products. Demand is defined as independent when it is not a function of demand for other inventory items. Independent demand must be forecast...

He goes on to say:

> ...demand is considered DEPENDENT when it is directly related to, or derives from, the demand for other items or end products. Such demand can, of course, be calculated. Dependent demand need not, and should not, be forecast. It can be determined from the demand for those items to which it is component...

According to Plossl and Wight (1970) the principle of independent/dependent demand states:

> ...where demand is independent (finished goods items or service parts not used in current production) the order point technique is applicable. Where demand is dependent, as it is for all components and semi-finished inventory items that 'go into' another inventory item, material requirements planning is a far more satisfactory technique...

An operational definition of requirements planning is not given in the literature. Berry (1971) came closest when he stated that requirements planning "...systems reduce a master schedule of finished products to a time-phased schedule of requirements for intermediate assemblies and component parts...". It may be more meaningful to refer to requirements planning as a concept rather than a technique. There is no mathematical expression or relationship which can be applied to all requirements planning applications. Requirements planning systems are applicable to operating systems which have time lags in the assembly of finished products.

The concept of requirements planning is to combine the parts explosion matrix[1] of finished products with the time-series matrix[2] of components of the finished products so that once a production plan for finished products has been determined, the time-phasing of assembling the various component parts and materials can be organized to facilitate the final assembly of the

---

1. A parts explosion matrix is a listing in tabular form of all the materials required for each part to show the total number of each component required to manufacture each part or assembly.

2. A time-series matrix is a listing in tabular form of the required lead times for components to be available for use prior to the final assembly of the finished product.

finished product on schedule. The accuracy of the finished product production schedule is vitally important to the successful application of requirements planning.

An example of a time-phased multistage production system is shown in Figure 2.7. The product explosion matrix indicates the number and the identity of components used at the various stages. For example each unit of X requires 1 subassembly A, 2 subassemblies B, 1 part D and 1 raw material F. Similarily each unit of part D requires 2 raw materials E and 1 raw material F. The time-series product explosion diagram shows how the product explosion matrix is combined with the lead times in the system structure to provide a time series of total requirements for all components in product X. It is particularily important to note that some components, such as raw material F, can be required in several periods. The time phased requirements table shows how the time-series product explosion diagram is combined with the product requirements to determine the quantities of the various components required in each time period.

Figure 2.7

Example of Time-Phased Multistage Production

Assume Product X requirements are given by Period:

Period	1	2	3	4	5	6
Requirements	0	0	0	100	1000	100
Period	7	8	9	10	11	12
Requirements	100	100	0	0	0	0

Product Explosion Matrix

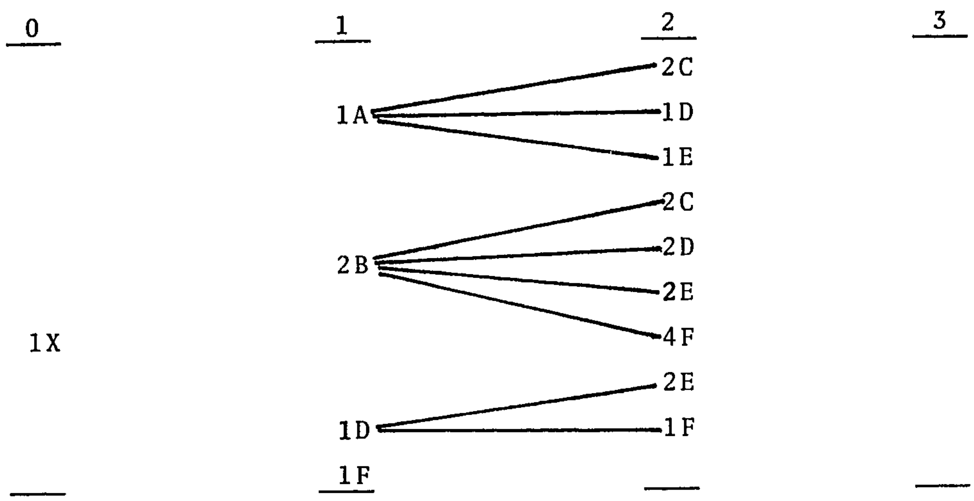
	Product X	Subassembly A	Subassembly B	Part C	Part D	Raw Material E	Raw Material F
Product X	0	1	2	0	1	0	1
Subassembly A	0	0	0	2	1	1	0
Subassebmly B	0	0	0	1	1	1	2
Part C	0	0	0	0	0	1	2
Part D	0	0	0	0	0	2	1
Raw Material E	0	0	0	0	0	0	0
Raw Material F	0	0	0	0	0	0	0

Assume components must be available at start of period to facilitate production.

Figure 2.7 (Cont'd)

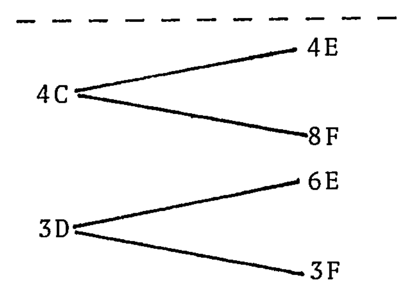
Time-Series Product Explosion Diagram

Lead Time in Periods



Sub-Total Requirements

0	1	2	3
1X	1A	4C	
	2B	3D	
	1D	5E	
	1F	5F	



Total Requirements

0	1	2	3
1X	1A	4C	10E
	2B	3D	11F
	1D	5E	
	1F	5F	

Figure 2.7 (Cont'd)

Time-Phased Requirements Table

	PERIOD 1	2	3	4	5	6	7	8
Product X	0	0	0	100	1000	100	100	1000
Subassembly A	0	0	100	1000	100	100	1000	0
Subassembly B	0	0	200	2000	200	200	2000	0
Part C	0	400	4000	400	400	4000	0	0
Part D	0	0	100	1000	100	100	1000	0
	0	300	3000	300	300	3000	0	0
	0	300	3100	1300	400	3100	1000	0
Raw material E		500	5000	500	500	5000	0	0
	1000	10000	1000	1000	10000	0	0	0
	1000	10500	6000	1500	10500	5000	0	0
Raw material F			100	1000	100	100	1000	0
		500	5000	500	500	5000	0	0
	1100	11000	1100	1100	11000	0	0	0
	1100	11500	6100	1600	11500	5000	0	0

Plossl and Wight (1970) claim that requirements planning:

> ...is the proper inventory management technique to use in controlling the following types of material:
>
> 1. Components
> 2. Raw Material
> 3. Service parts also used in current production.
> 4. Semi-finished materials.
> 5. Branch warehouse inventory feedback.
> 6. Lumpy 'independent' demand.

The implementation of a requirements planning system requires that some form of lot-sizing be used for finished products and for components. Plossl and Wight (1970) list the following techniques:

Square root EOQ.

Discrete lot-sizing:
- Least Total Cost.
- Least Unit Cost.
- Periodic Order Quantity.
- Lot-for-Lot.
- Wagner-Whitin.

Fixed quantity.

Each of these lot-sizing techniques is useful in certain applications. It is the responsibility of the user to determine which technique is the most appropriate for the particular application. In general, the lot-sizing method should be individually selected for each inventory item.

Plossl and Wight (1970) also discuss the use of safety stocks in requirements planning systems. They state:

> As with the order point/order quantity system, safety stocks are necessary in material requirements plans to protect against demand variations for the end products and against supply variations for components. A sound approach is to carry safety stocks at the top level in end products and at the bottom level in raw materials. This provides shelf stock to meet normal demand variations for the finished products and orders raw material to permit releasing manufacturing orders early to level production on the plant and to react quickly to unexpented scrap losses or unusually high finished goods demands. Safety stocks might also be carried on semi-finished components which can be converted quickly into a wide variety of higher-level items. This approach, combined with reduced part manufacturing lead times, results in greater flexibility to meet changing demand with minimum inventories...

Plossl and Wight (1970) list the following three methods of setting safety stocks in materials requirements planning systems.

1. Safety time.
2. Increase master schedule.
3. Fixed quantity.

Safety time and fixed quantity safety stocks were discussed in Section 2.2.2. Safety stock attained by increasing the master schedule is similar to both safety time and fixed quantity safety stocks as none of these methods consider the magnitude of the variation in demand. Increasing the master schedule by an arbitrary factor implies that inventory in excess of the expected demand be available to supply the demand when it arises without explicit consideration of the relevant costs.

A general model of the multistage inventory problem is formulated in Chapter Three. This model is useful for conceptualizing multistage inventory problems and for appreciating the complexity of optimization in multistage systems.

Chapter Three

A General Multistage Inventory Model

The multistage inventory models discussed in the previous chapter can be considered as either specific or general models. The Taha and Skeith model was limited to multistage single-product systems subject to deterministic continuous demands. The Zangwill, and Crowston and Wagner models were developed for multistage multiproduct systems which interact with discrete deterministic demand functions. These models are limited to applications under specific conditions.

Multistage inventory control applications are not limited to the restrictions of the models discussed above. A general model of a multistage multiproduct production system and a solution framework are developed below.

3.1 Statements of the Problem

It has been stated previously that the assumption underlying this research project is that the organizations' objective is to minimize the total cost of operations.

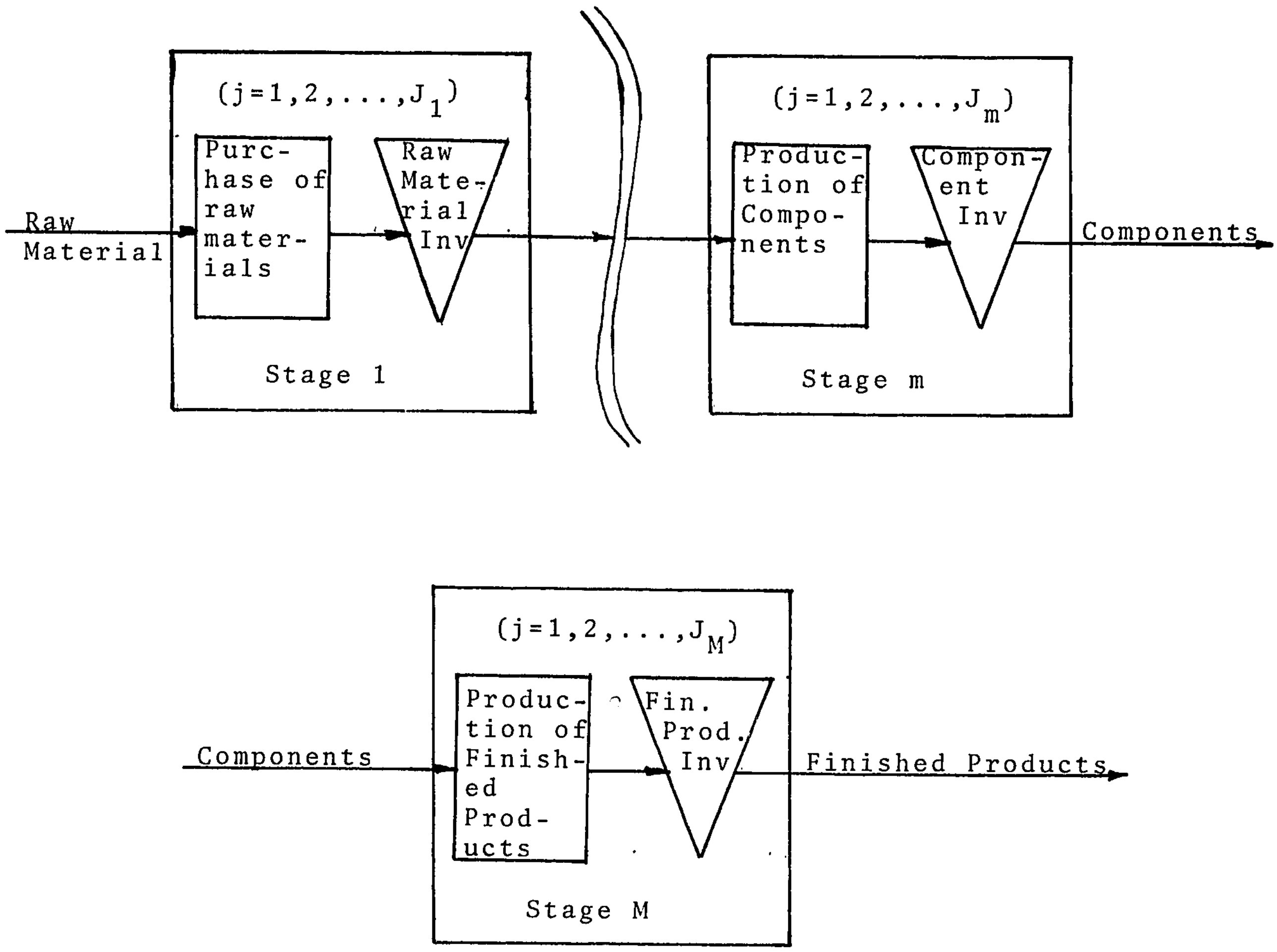
This can be rephrased to state that the objective is to minimize the total cost of operating a multistage multi-product manufacturing facility over a planning horizon of N periods.

Assume a general M-stage manufacturing facility which process J_m products at each stage M (m=1,2,...,M). Each stage consists of a production facility and an inventory level as defined in Chapter One. The initial facility represents the purchasing function and the associated raw materials inventory. The final facility processes the finished products and stores the finished product inventories. Figure 3.1 illustrates such a system.

An inventory policy is required for each product or component in the system and a workforce decision rule is required for each facility. The demand for products and components in the system can either be exogenous (from extenral sources) or endogenous (resulting from the demands of other products or components). It is convenient to assume that the units produced in period n (n=1,2,...,N) can be used in subsequent operations only in succeeding periods but that they can be used to supply exogenous demands in the current period n.

Figure 3.1

Typical Multistage Multiproduct System



Since the problem is to minimize the total cost over a planning horizon of N periods, it can be described in terms of a sequential decision problem of the form shown in Figure 3.2, where:

X_{n-1} = a state vector which gives all the relevant information about the inputs to period n.

X_n = a state vector which gives all the relevant information about the outputs from period n, which become the inputs to period n+1.

D_n = the decision vector which controls the operations of period n.

r_n = the return scalar of period n which measures the utility of the period as a single-valued function of inputs, decisions and outputs.

(3.1) $$r_n = r_n \ (X_{n-1}, D_n, X_n).$$

The stage coupling function, t, expresses the output of each component as a function of the input state and the decision vectors.

(3.2) $$X_{n+1} = t_n \ (X_n, D_n).$$

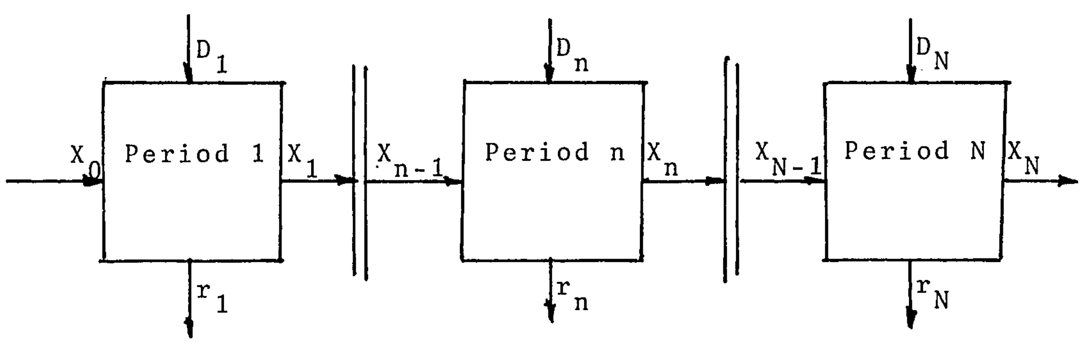
3.2 Development of the General Model

For the general multistage multiproduct problem, let:

X_n = the vector of inventories and workforces on hand at the end of period n (for all M production stages).

Figure 3.2

Typical Sequential Decision Problem



D_n = the vector of inventory policies and workforce decision rules incorporated in period n (for all M production stages).

R_n = the vector of demand for all products and components in period n (for all M production stages).

r_n = the cost of operations in period n (for all M production stages).

The transformation during any time period n can be represented by Figure 3.3.

The model can be expanded by substituting for the vector of inputs, X_n, and the decision vector, D_n. X_n is composed of two parts:

I_n = the vector of inventory quantities on hand at the beginning of period n. The elements of I may be negative indicating outstanding backorders.

G_n = the vector of workforces for each stage m (m=1,2,...,M) available in period n. The elements of G_n must be non-negative.

D_n is also composed of two parts:

W_n = the vector of workforce decision rules.

Z_n = the vector of inventory policies.

An inventory policy is composed of two parts, decision rules regarding when to order and how much to order. These are referred to as order points and order quantities. The vector Z_n can therefore be subdivided into:

B_n = the vector of order points decision rules in period n.

Figure 3.3

Transformation During a Typical Period n

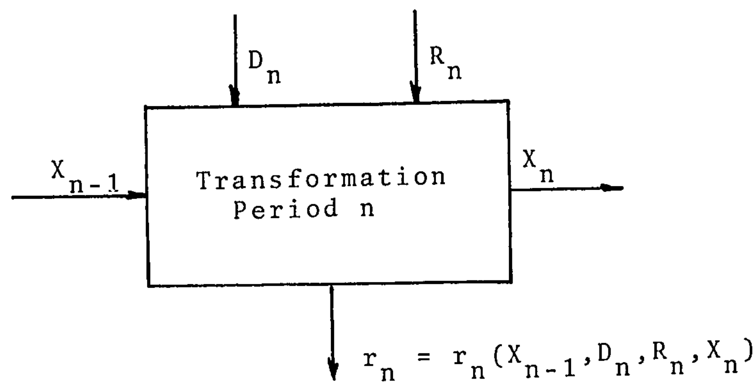
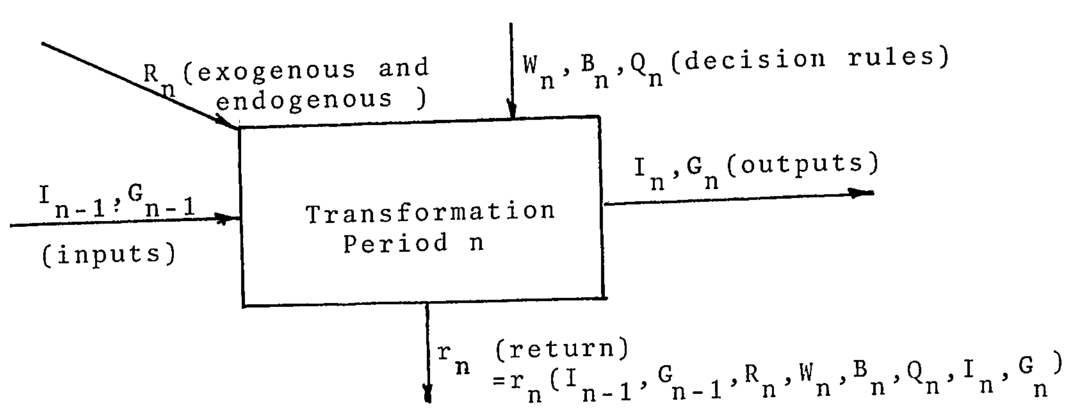


Figure 3.4

Typical Period in the Expanded Problem



Q_n = the vector of order quantity decision rules in period n.

The diagram in Figure 3.4 represents a typical period in the problem as it is now defined.

Assume that the vector of demands, R_n, can be divided into a vector of expected demands, F_n, and a vector of stochastic random variations, E_n.

F_n = the vector of mean expected demands for all products and components. This is equivalent to the actual demands without the stochastic random variation.

E_n = the vector of stochastic random variations of the actual demands from the mean expected demands in period n.

The vector of order point decision rules, B_n, can be subdivided into two components V_n and S_n in accordance with the division of the demand vector.

Let: V_n = the vector of order point decision rules which would be used if there were no stochastic random variations, E_n, in the demand for products and components.

S_n = the vector of safety stocks employed to reduce the costs of backorders resulting from demands in excess of available inventories.

The transformation and cost functions can now be written as:

(3.3) $(I_n, G_n) = t_n (I_{n-1}, G_{n-1}, F_n, E_n, W_n, S_n, Q_n)$.

(3.4) $r_n = r_n (I_{n-1}, G_{n-1}, F_n, E_n, W_n, S_n, Q_n, I_n, G_n)$.

The objective is to maximize the total cost of operations, C_N, over a planning horizon of N periods.

(3.5) Minimize $C_N = \min \sum_{n=1}^{N} r_n$.

Assume that N is sufficiently large so that the ending state conditions I_N and G_N will not influence the decision structure and that the initial state conditions I_0 and G_0 are known.

The general production-inventory identity states:

(3.6) $I_n = I_{n-1} + P_n - R_n$.

where: I_n = inventory on hand at the end of period n.

P_n = production in period n.

R_n = demand in period n.

In the general problem outlined above:

(3.7) $P_n = P_n\ (I_{n-1}, V_n, S_n, Q_n, G_n)$

assuming that the available workforce and inventories are the only constraining factors on production.

But (V_n, S_n, Q_n) describe an inventory policy. Therefore define a vector δ_n such that the elements of δ_n are given by:

(3.8) δ_n = 1 if the decision rules of the inventory policy of an inventory item indicate that production of the item is necessary in period in $(Q_n > 0)$.

= 0 if the decision rules of an inventory policy indicate that no production of the item is necessary in period n $(Q_n = 0)$.

Therefore:

(3.9) $P_n = \delta_n Q_n \mid I_{n-1}, G_n$.

This relationship indicates that the production quantity is equal to Q_n if there is to be any production of the item in period n provided that there is sufficient inventory and manpower available to facilitate the production.

The general nature of the model as it is presently developed encompasses a wide variety of applications but by being so general, it lacks tractability. To become more operational, assumptions must be made which restrict the generality but enhance the solution procedure.

3.3 An Inventory Dominant Model

The objective of this research project is to compare the relative performance of the application of inventory policies in a multistage manufacturing system. The influence of other decision rules must be eliminated wherever possible. Therefore, assume that the vector of inventory policies is to dominate the production stage such that the workforce at each stage will be adjusted each period to provide sufficient personnel to facilitate the desired production on regular time.

Let: U_n = the vector of unit manufacturing times for all products and components in the system. The elements of U_n for raw materials represent the unit costs of purchased materials.

A_n = the vector of set-up time for all products and components in the system. The elements of A_n for raw materials represents the cost of placing orders for purchased materials.

H_n = the number of regular time hours available per employee in period n.

Therefore:

$$(3.10) \quad G_n = \frac{\sum_{j=1}^{J_m} (U_{j,m,n,}\ Q_{j,m,n} + A_{j,m,n}\ \delta_{j,m,n})}{H_n}, \quad \begin{matrix}(m=2,3,\ldots,M).\\(n=1,2,\ldots,N).\end{matrix}$$

Since the manpower decision rules are derived from the order quantity decision rules resulting from the inventory policies, the actual production quantities are given by:

$$(3.11) \quad P_n = \delta_n \quad Q_n \mid I_{n-1}.$$

The formulation can therefore be expressed as:

$$(3.12) \quad I_n = t_n\ (I_{n-1},\ P_n,\ F_n,\ E_n) = I_{n-1} + P_n - F_n - E_n.$$

$$(3.13) \quad G_n = \frac{\sum_{j=1}^{J_m} U_{j,m,n}\ \ Q_{j,m,n} + A_{j,m,n}\ \ \delta_{j,m,n})}{H_n}$$

$$(3.14) \quad r_n = r_n\ (G_{n-1},\ F_n,\ E_n,\ P_n,\ G_n)$$

The model still lacks operationality which can only come from specifying the inventory control policies which are to be used for each product or component. An example of such a model is developed in the next section.

3.4 A Lot-for-Lot Inventory Policy Model

As an example of an inventory dominant model, assume that a Lot-for-Lot inventory policy is to be used for all inventory items. Also assume that only finished products (products from production stage M) have exogenous demands. All other products and components have endogenous demands resulting from the assembly of other products. In addition, assume that a product explosion matrix XM is defined such that each element of $XM_{a,b,c,d}$ indicates the number of components of product c stage d which are used in the assembly of product a stage b.

The Lot-for Lot decision rule for the desired production, P^1, of finished products is:

(3.15) $$P^1_{j,M,n} = F_{j,M,n} - I_{j,M,n-1} + S_{j,M,n}$$

where: M = finished product stage

j = product number at the production stage.

n = time period.

If it is assumed that S is a function of the standard deviation, σ, of the random variation of the deamnd, E, then the decision variable is the multiple z of the standard deviation of demand.

Therefore:

(3.16) $$P^1_{j,M,n} = F_{j,M,n} - I_{j,M,n-1} + z_{j,M,n}\,\sigma_{j,M,n}.$$

$(j=1,2,\ldots,J_M)$.

$(n=1,2,\ldots,N)$.

The decision rules for component products make use of the product explosion matrix and the demand for finished products as follows:

$$(3.17)\quad P^{1}_{j,m,n} = \sum_{a=1}^{J_m} P^{1}_{j,M,n}\ XM_{a,M,j,m} + \sum_{b=1}^{M}\sum_{c=1}^{J_b} F_{c,b,n+1}\ XM_{c,b,j,m} - I_{j,m,n-1} + z_{j,m,n} \sum_{b=1}^{M}\sum_{c=1}^{J_b} \sigma^{2}_{c,b,n+1}\ XM_{c,b,j,m}.$$

(m=1,2,...,M-1)
(n=1,2,...,N)
$(j=1,2,...,J_m)$

The workforce decision rules are:

$$(3.18)\quad G_{m,n} = \frac{\sum_{j=1}^{J_m} (A_{j,m,n} + U_{j,m,n}\ P^{1}_{j,m,n}\ \delta_{j,m,n}).}{H_n}$$

(n=1,2,...,N)
(m=2,3,...,M)

The actual production quantity of any item will depend upon the availability of the required inventories. Assume that the production progresses iteratively by production stages from stage M to stage 2. Assume that in the event of insufficient inventory to facilitate all the desired production at any stage, the production quantities of all products requiring the constraining materials are reduced according to the ratio of the available to the desired quantities. Define a matrix MR such that each element of $MR_{a,b,c,d}$ represents the quantity of product a stage b required for the assembly of the desired production at stage c in time period d. Then:

$$(3.19) \quad MR_{j,m,b,n} = \sum_{a=1}^{J_b} P^1_{a,b,n} \; XM_{a,b,j,m} \; .$$

Define a matrix FA such that each element of $FA_{a,b,c,d}$ indicates the ratio of available to required inventories of product a stage b needed for desired production at stage c in period d.

$$(3.20) \quad FA_{j,m,b,n} = \begin{bmatrix} \dfrac{I_{j,m,n-1}}{MR_{j,m,b,n}} & \text{if} \quad I_{j,m,n-1} < MR_{j,m,b,n} \; . \\ 1 & \text{if} \quad I_{j,m,n-1} \geq MR_{j,m,b,n} \; . \end{bmatrix}$$

The quantity of material used in the assembly of products at state b in period n is therefore:

(3.21)

$$MR^1_{j,m,b,n} = MR_{j,m,b,n} + \sum_{a=1}^{J_b} P^1_{a,b,n} \; XM_{a,b,j,m} \; (FA_{j,m,b,n} - 1) \; .$$

The inventory remaining for further use in assemblies is:

$$(3.22) \quad I_{j,m,n-1} = I_{j,m,n-1} - MR^1_{j,m,b,n} \; .$$

The actual production in period n is:

$$(3.23) \quad P_{a,b,n} = P^1_{a,b,n} \; FA_{j,m,b,n} \; .$$

(b progresses sequentially M,N-1,...,2)
($a = 1,2,\ldots,J_b$)
($m = 1,2,\ldots,M-1$)
($j = 1,2,\ldots,J_m$)
($n = 1,2,\ldots,N$)

Following the above adjustments, the ending inventory is given by:

$$(3.24) \quad I_{j,m,n} = I_{j,m,n-1} + P^{1}_{j,m,n} - F_{j,m,n} - E_{j,m,n}$$

where $P''_{j,m,n}$ is the final production decision following adjustments for material availability.

The costs associated with the decision rules are as follows:

labour costs.
hiring costs.
firing costs.
transfer costs (workforce adjustments between production stages).
inventory carrying costs.
backorder costs.
raw material ordering costs.

These costs are the elements of r_n which can be expressed as a function of the inputs, outputs and decisions. The objective is to minimize the total costs over a planning horizon of N periods. The decision variable for the minimization is z, the vector of the multiples of the standard deviations of the stochastic random variations of demand.

3.5 Solution Procedures

One solution procedure for a problem of the type outlined above is a nested dynamic programming method using a recursive relationship of the form:

(3.25) $$f_N(X_N) = \max_{D_N} r_N(X_N, D_N) = f_{N-1}\{t_N(X_{N-1})\}.$$

Let:

(3.26) $$f_n(z_n) = \min_{z_n} \{r_n(I_n, G_{n-1}, z_n) + f_{n-1}(t_n(I_n, G_{n-1}, z_n))\}.$$

subject to: $I_n = I_{n-1} + P_N^1 - F_n - E_n$.

$$G_n = \frac{\sum_{j=1}^{J_m} (A_{j,m,n} + U_{j,m,n} P_{j,m,n}^1)^{\delta_{j,m,n}}}{H_n}$$

The objective is to minimize $f_N(z_N)$ where:

(3.27) $$f_N(z_N) = \min_{z_n} R_n(I_n, M_{n-1}, z_n), \quad (n = 1,2,\ldots,N).$$

(3.28) $$R_n(I_n, M_{n-1}, z_n) = r_n(I_n, G_{n-1}, z_n), \quad (n=1).$$

$$= r_n(I_n, G_{n-1}, z_n)$$

$$+ f_{n-1}(t_n(I_n, G_{n-1}, z_n)), \quad (n=2,3,\ldots,N).$$

A solution procedure for the above model requires the evaluation of an expectation function to determine the optimum safety stock for each inventory item. The expectation function incorporates the decision variable, z_n, in both the limit of an integral and in a density function. A trial and error procedure is therefore necessary to find an optimal value of z_n. This amounts to a search for the cost minimizing safety stock level.

The above solution procedure is conceptually possible but impractical for problems involving several products and several production stages. Since the solution of the analytical model requires a search for the cost minimizing safety stocks, a more direct procedure is to use simulation in the search for the cost minimizing safety stock decision rules.

The use of simulation as a method of determining optimal decision rules can utilize the analytical model of the production system but does not require the analytical solution. Therefore if a satisfactory model of the production system can be formulated, simulation can be used directly as a solution procedure. Simulation techniques do not guarantee optimal solutions, but search procedures can be incorporated to determine near optimal solutions.

Another advantage of simulation techniques is that the influence of variables can be tested on the model at relatively low cost as compared to tests conducted on an industrial system. The experimental testing of variables on simulation models can be controlled so that the effects of each variable can be tested with no interference from random or extraneous inputs.

The complexity of the analytical solution procedures and the flexibility of simulation indicated that the use of simulation was an appropriate technique for this research project. The following section discusses the simulation model.

3.6 The Test Environment

The experimental study required a model of a multistage multiproduct facility. Two alternatives which were immediately apparent were: (1) to solicit the cooperation of an industrial firm and model the system of that firm, or; (2) adopt a simulation model which had previously been developed. Several simulation models were available, and since some of these have been available for a few years and have been generally accepted as representative of industrial situations, it was decided to use such a model.

The model chosen for the experiment was the operations management game "Production Control Project-DCIDE" which was originally developed by Dr. P. F. Winters. Several versions of the game are in existence, the differences being adjustments in the programming to suit computer hardware. The structure of "Production Control Project-DCIDE" is the same in all versions. The game was modified by Britney (1972) for use on an IBM 1130 computer. The Britney version has added some monitoring facilities for information about the progress of all players. A players manual for the Britney version is in Appendix III. A version of the game which was written for CDC computers was adapted for use in this research.

"Production Control Project-DCIDE" is a simulation model of a three stage manufacturing operation. The model includes 15 finished products, 7 subassemblies, 15 parts and 8 raw materials. Finished products can be made and sold in the same time period. Subassemblies and parts are made in one time period and are available for use in the succeeding periods. Four of the raw material items are available for use in the period after they have been ordered while the other four raw material items have a two time period lag between ordering and availability for use. There is no constraint on the quantity of raw materials which can be ordered in any time period. A schematic of the factory and a flow diagram of the simulation are shown in Figures 1 and 7 of the players manual in Appendix III respectively. The program listing and flow diagram of the modified game follow the players manual in Appendix III.

The simulation model requires forty-eight data inputs in each time period. These inputs are decisions regarding the quantities of products to be made for each of the 15 finished products, 7 subassemblies and 15 parts, the quantities of the 8 raw materials to be ordered and the size of the labour force to be employed in each of the 3 operating stages of the manufacturing facility. There is no limitation on the availability of personnel for the work force. The volume of materials processed in any time period is limited only by the labour availability in man-hours and the materials in inventory. Overtime can be used at a fifty percent premium to a maximum of thirty percent of the total regular time hours. Each product requires a predetermined number of set-up hours before production can start in each time period as well as a standard number of labour hours per piece produced.

There are charges for hiring, firing, and transfering employees from one department to another. A cost of $500.00 per employee is assessed for every addition to the workforce and $700.00 is charged for every employee discharged. A charge of $100.00 is made for transferring employees from department to department. Additions to or deletions from the total workforce are made after the possibilities of transfers have been exhausted. The time period is taken as one month and there are 160 regular time man-hours in a month. A charge of $3.50 per man-hour is assessed for every man-hour contracted and employees cannot be hired for less than a month. Other versions of the game allow fractional employees in the labour force indicating part-time employment. This feature was altered for this research so that the workforce in any time period was an integer number of employees.

An inventory carrying charge of two percent per month is made on the value of the month end inventory. The value of each product is updated each period to maintain a current per unit valuation.

A stockout cost of $140.00 per unit is assessed for each unit of finished product demand which cannot be delivered during the month the demand is observed. If the order cannot be delivered immediately, it is backordered and held until such time as there is inventory available. An ordering cost of $50.00 per order placed per product has been added to the game for this research in order to facilitate the calculation of economic order quantities for raw materials.

The demand functions for products 1 to 12 are given by base, trend, seasonality and error parameters. Products 13 to 15 originally had exponentially distributed demand functions but these were revised for this research in order to conform with the demand functions of other finished products. The revised parameters of the demand functions are shown in Table 3.1.

The definition of an inventory policy is explained in Chapter Four together with the selection of inventory policies tested.

Table 3.1

Revised Demand Parameters for Products 13 to 15

Product I	Basic Demand Factor AS(I)	Trend Factor BS(I)	Seasonality Factor CS(I)	Random Factor SIG(I)
13	400.00	20.00	100.00	0.05
14	700.00	13.00	200.00	0.01
15	1500.00	-25.00	300.00	0.35

Chapter Four

Description of Inventory Policies Tested

4.1 Definition of an Inventory Policy

An inventory policy is a set of decision rules which are applied in the analysis of inventory items to determine the timing and the quantity of orders to be placed to replenish the item's stock. This definition is sufficiently flexible to permit the existence of a separate inventory policy for each inventory item in the system. However, unless otherwise indicated, it will be assumed for the remainder of this study that the adoption of an inventory policy implies that the policy will be used for all inventory items in the system. This does not mean that the parameters for ordering replenishment stocks will be the same for all items, but rather, the method of determining the parameters will be common to all inventory items in the system.

The above inventory policy definition indicated that there are two types of parameters in an inventory policy: timing parameters and quantity parameters. There are several methods of determining both types of parameters which will be discussed below.

The timing parameters of inventory policies can be either fixed or variable. A fixed timing parameter stipulates that the inventory stock be replenished at fixed time intervals regardless of the quantity of inventory in stock. A variable timing parameter implies that a replenishment order be dispatched whenever the stock on hand is reduced to a quantity related level, usually referred to as an order point. Depending on the prevailing conditions, the order point can be either a fixed or a variable quantity.

Order points are calculated using the mean expected demand during the lead time and the variation in demand to determine the timing of the order placement for replenishment stock. Inventory policies related to deterministic demand functions do not require safety stocks for protection against random variations in the demand functions. The order point is simply equal to the known demand during the replenishment lead time. Inventory policies which are designed to control inventories subject to stochastic demands require safety stocks to protect against backorders from demands larger than the mean expected demands during the replenishment lead times. The calculation of the desired safety stock is dependent upon the inventory policy being administered. Given the desired quantity of safety stock for each inventory item, the order points are calculated by adding the mean expected demand during the replenishment lead times and the desired safety stocks for each inventory item.

The order quantities may also be fixed or variable. A fixed order quantity implies that whenever a replenishment order is issued for an inventory item, the order is for a standard quantity regardless of the prevailing conditions. A variable order quantity requires the calculation of the quantity to be ordered each time an order is placed. The amount of inventory presently in stock and the forecast demand for the item are used to determine the order quantity. The method of determining lot-sizes will also influence the order quantity.

In the previous chapters, the need to compare the application of requirements planning and statistical inventory policies was established. It was also noted, that according to the literature pertaining to inventory control, a combined statistical-requirements planning policy should be optimal for a multistage multiproduct system. These policies will be discussed in the following sections.

4.2 Requirements Planning Policies

As discussed previously, Plossl and Wight (1970) suggest the following lot-sizing procedures for requirements planning systems:

- EOQ.
- Discrete lot-sizing:
  - Least Total Cost.
  - Least Unit Cost.
  - Periodic Order Quantities.
  - Wagner/Whitin.
- Fixed Quantity.

A comparison of the Economic Order Quantities, Q_i, and the base demand factors, AS_i, for the 15 finished products in the simulation game is presented in Table 4.1. The values of Q_i and AS_i were taken from Figure 6 of Appendix III. The weighted average ratio of $\frac{Q_i}{AS_i}$ of 1.909 indicates that the Economic Order Quantities are approximately equal to two months of base demand. Since the ratio was slightly less than 2.0, it was decided to use a Lot-for-Lot, lot-sizing method for all products. The choice of a Lot-for-Lot method also facilitated the calculation of safety stocks, which will be discussed below.

Table 4.1

Comparison of Economic Order Quantities to Base Demands for Finished Products

Product i	Q_i	AS_i
1	2000	1000
2	1500	500
3	600	200
4	400	100
5	250	50
6	600	300
7	1000	500
8	1000	1000
9	400	100
10	1500	500
11	600	200
12	500	100
13	1500	400
14	1200	700
15	1600	1500
TOTAL	13650	7150

$$\text{WEIGHTED AVERAGE} = \frac{13650}{7150} = 1.909$$

The calculation of safety stocks in requirements planning systems has been recognized as a weakness of this system. Plossl and Wight (1970) suggested the following three methods for determining safety stocks in requirements planning systems:

Safety Time.

Increase Master Schedule.

Fixed Quantity.

The use of a "safety time" safety stock implies that, in addition to the inventory required to satisfy the demand during the lead time, some extra periods of expected demand be carried to protect against unusually high demands during the replenishment cycle. Plossl and Wight (1971) commented on the use of safety time as follows:

> Putting in safety time really doesn't tell the system the truth and can be particularly harmful if the safety stock is varied for different items... Priorities are distorted and by such cushions, work-in-process inventories are inflated and operating people soon learn that they have more time to get parts than the due dates indicate. The resulting 'credibility gap' can easily offset the benefits of having safety allowances.

Increasing the master schedule requires that the demand requirements of finished products be over-forecast. This can inflate the inventory of components common to several finished products by exploding the bill of materials to the level of these common components.

It is highly unlikely that the actual demand for all finished products will be at the maximum level simultaneously, therefore, the requirements for common components will be overstated.

The use of a fixed quantity safety stock was discussed previously. It was stated that this procedure might be applicable to demand functions which had relatively small variations in demand over time.

Plossl and Wight (1970) indicated their dissatisfaction with the above methods of calculating safety stocks by stating:

> More theoretical work remains to be done to develop a rational basis for setting safety stocks on dependent demand items similar to the statistical approaches used for independent demand items. The definition of 'Customer service' and 'forecast error' are yet unclear when applied to dependent demand items.

The references to customer service and forecast error in the above statement are taken to mean the percentage of demands satisfied for inventory and the variation in demand for component items. In their 1971 publication, Plossl and Wight stated: "Demand variation could be measured at the component level and a safety stock calculated for the component, using statistical order point methods". This is equivalent to considering the demand for components as being independent, which is what they argue against.

If the lot-sizing techniques aggregate the demands for two or more periods, safety stocks could be determined by recording the demands by period, and calculating the mean and standard error of the demand function. However, if the Lot-for-Lot, lot-sizing method is used throughout the materials requirements planning system and the standard error of the forecast demands for the finished products are known several periods in advance, it is then possible to calculate safety stock for all inventory items quite simply.

Given that each finished product has a constant coefficient of variation, meaning that the standard error of the demand is a constant proportion of the mean demand over time for each finished product, the problem is simplified even further. The only information required to calculate safety stock under these conditions is the future demands for the finished products, the coefficients of variation, and the product explosion matrix. Peterson, Thomas and Loiseau (1971) have suggested that for similar circumstances, the standard error of demand for several periods is equal to the square-root of the sum of the squared standard errors of the forecast demand in each period. Therefore the standard error for component parts can be determined by calculating the square-root of the sum of the squared standard errors of the forecast demand for finished products offset by the lead time required between the components' production and the assembly of the finished products. Consider the following example.

If an inventory item is used twice in finished product A and the lead time between the production of the item and the final assembly of finished product A is three periods, and the same item is used three times in finished product B, the lead time for two usages being two periods and the lead time for the third usage being four periods, the standard error of the demand for the component can be calculated as:

$$\sigma_C = \sqrt{2.(\sigma_{A,3})^2 + 2.(\sigma_{B,2})^2 + (\sigma_{B,4})^2} \quad (4.1)$$

where: σ_C = standard error of the demand for the component.

$\sigma_{A,3}$ = standard error of the forecast demand for finished product A three periods hence.

$\sigma_{B,2}$ = standard error of the forecast demand for finished product B two periods hence.

$\sigma_{B,4}$ = standard error of the forecast demand for finished product B four periods hence.

If it is assumed that each finished product has a constant coefficient of variation, then Equation (4.1) could be written:

$$(4.2) \quad \sigma_C = \sqrt{2.(V_A.f_{A,3})^2 + 2(v_B.f_{B,2})^2 + (v_B, f_{B,4})^2}.$$

where: v_A = coefficient of variation for product A.

$f_{A,3}$ = mean forecast demand for product A three periods hence.

v_B = coefficient of variation for product B.

$f_{B,2}$ = mean forecast demand for product B two periods hence.

$f_{B,4}$ = mean forecast demand for product B four periods hence.

Calculating the standard errors of the demands for component items as illustrated in Equations (4.1) and (4.2) does not require the retention of past data to enable the compilation of statistics. These equations are responsive to changes in the forecasts, thereby minimizing safety stocks.

The production schedule for the requirements planning policies using Lot-for-Lot, lot-sizing methods was determined from the forecast demands for each finished product during each time period over the required planning horizon. The gross requirements of the finished products were equal to the forecast demand over the replenishment lead time plus the desired safety stock level calculated as a multiple of the standard error of the forecast demand. The net requirements of these finished products, which became the planned production quantities, were equal to the gross requirements, less the stock on hand, plus the current number of backorders.

The gross requirements for the component items were determined from the forecast demands for the finished products and the time-series product explosion matrix. The standard error of demand for each inventory item was determined using the square-root method discussed above. The net requirements for each inventory item was equal to the gross requirements, including safety stock less the stock on hand.

In summary, the requirements planning inventory policy used in this research project incorporated a Lot-for-Lot, lot-sizing method. This required that every inventory item be assembled or ordered every period in which a demand existed. The replenishment quantity of each inventory item was equal to the forecast demand, plus the desired safety stock, less the existing stock.

The listing and a flow chart of the computer subroutine, DCID1, used for the implementation of the requirements planning policies is given in Appendix IV. An analytical solution to the Lot-for-Lot inventory policy application to the "Production Control Project-DCIDE" game is given in Appendix V.

4.3 Statistical Policies

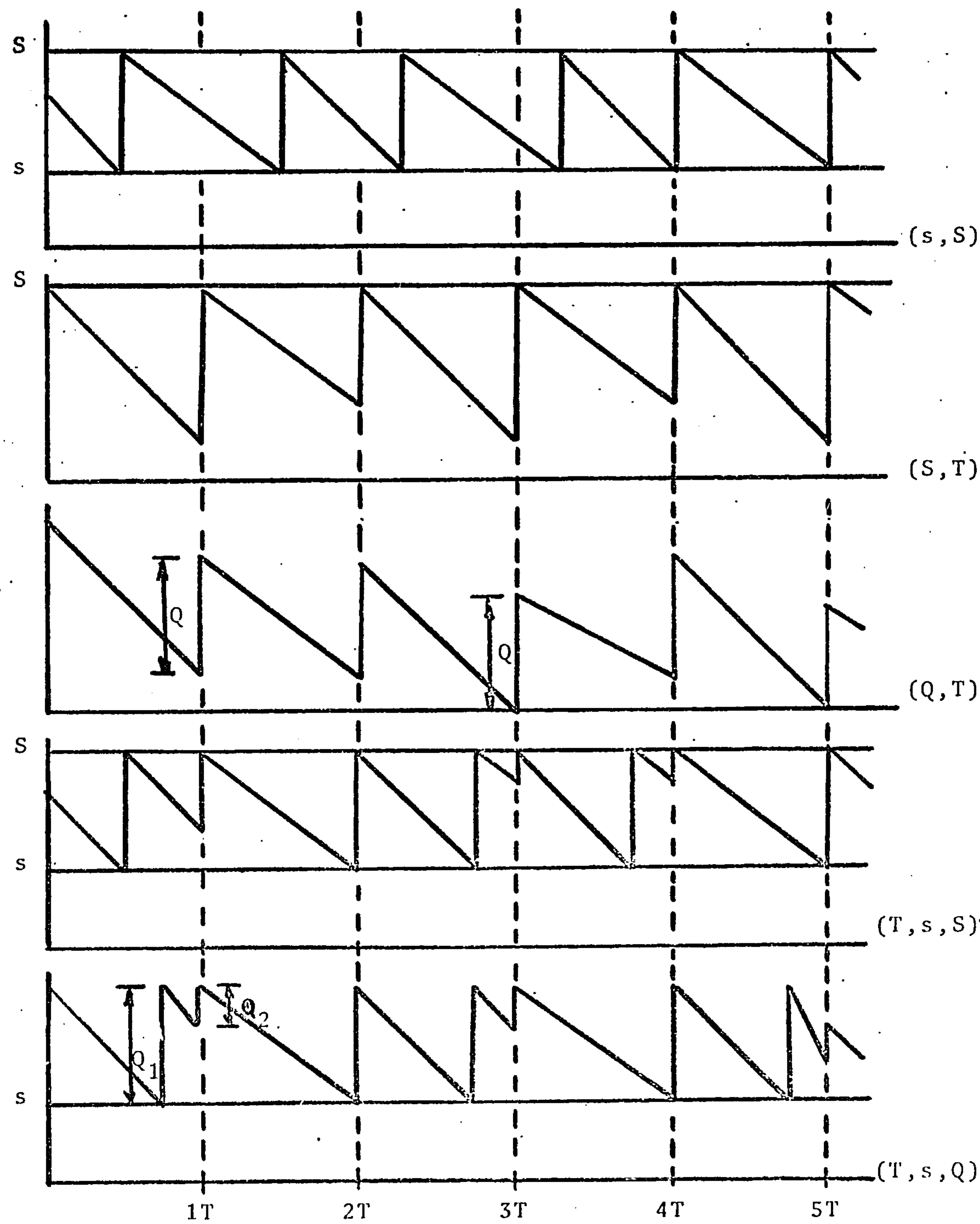
There are several statistical inventory policies which could be tested. Eilon and Elmaleh (1968) investigated the following policies: (s,S), (S,T), (Q,T), (T,s,S) and (T,s,Q). These policies are illustrated in Figure 4.1 and discussed below.

There is some confusion in the literature concerning the definition of an (s,S) policy. The confusion is in relation to the timing and the quantity of the replenishment order. Starr and Miller (1962) decided that the confusion was so diffuse that they chose not to discuss the use of (s,S) policies as such and referred to them as a special case of the "Q-system". A Q-system is defined as ordering a Wilson economic order quantity whenever the stock on hand is depleted to the order point.

A more common definition of an (s,S) policy could be described as an "order-up-to" policy. This is the policy type described by Scarf (1960), Iglehart (1962) and Veinott and Wagner (1965). An "order-up-to" policy requires that the inventory be reviewed at the end of every period. If the stock on hand is larger than the order point, s, no replenishment order is placed. However, if the stock on hand is less than or equal to s,

Figure 4.1

Illustration of Some Statistical Inventory Policies



a replenishment order is placed such that the stock on hand and on order is equal to S. The order quantity is S-s-O where O is the quantity of material on order at the time of the inventory review. The order point s is determined by the forecast demand over the lead time plus the desired safety stock. The upper limit S is the sum of the order point and the economic order quantity, Q as determined by the Wilson formula. This "order-up-to" interpretation is the definition of the (s,S) policies used in this research. The (s,S) policy is often referred to as the "two-bin system".

An (S,T) inventory policy is a cyclical reorder system. Replenishment is effected in every time interval T (which consists of an integer number of time periods) to build the stock to an upper limit S. Both parameters T and S can be varied to control the average inventory and to protect against backorders.

The (Q,T) system is similar to the (S,T) policy except the replenishment quantity Q is always equal to a fixed value, such as the Wilson EOQ.

The (T,s,S) policy is a combination of the (s,S) and (S,T) policies. Eilon and Elmaleh describe it as follows:

> ...replenishment is provided every interval T to bring the stock level up to the upper limit S, but if in between review periods the stock declines to s, an order for replenishment of a batch Q is made...
>
> The main advantage of the T,s,S procedure is that it provides a better insurance against runouts than the pure cyclical review and is particularly effective in clearing any backlogged demand.

The value of the order point, s, and the values of Q and S are the same as for an (s,S) policy.

A (T,s,Q) policy was also tested by Eilon and Elmaleh (1968) which they describe in the following way:

> This is a combination of the s,S and Q,T policies, and is therefore similar to T,s,S except that the replenishment takes the form of fixed quantities Q_1 and Q_2...
> There is a danger that, in the absence of an upper control limit, the average stock would tend to soar. To alleviate this effect, two fixed order quantities were defined:
> Q_1 - the quantity ordered when the order is triggered by s (i.e. when the stock level falls below or is equal to s);
> Q_2 - the quantity ordered when at the cyclical review the stock level is found to be above s. In this study we generally took $Q_2 = \frac{1}{2}Q_1$ although in principle other relationships may be considered.

The value of s was determined in the same way as for the other policies discussed above.

Eilon and Elmaleh (1968) concluded that the (T,s,S) policy was the most promising policy tested and their analysis indicated that the implementation of a (T,s,S) policy was less costly than the implementation of an (s,S) policy at both a 90 and a 95 percent service level.

A criticism of the Eilon and Elmaleh (1968) study is the method they used to calculate the order point s. They state:

> The value of s is taken as
>
> $$s = D(L) + k\bar{D}$$
>
> where D (L) is the forecast demand for the lead time L; $\bar{D}$ is the mean demand per period; k is a safety factor, which determines the size of the buffer stock and hence the likely frequency of stock runouts.

The use of $k\bar{D}$ to determine the desired amount of safety stock is an example of the use of "safety time" discussed above. The disadvantage of such a procedure is that it does not consider the variation in the demand. In separate articles discussing the optimality of (s,S) inventory policies, Scarf (1960) and Iglehart (1963) consider s as a function of the demand distribution. The use of $k\bar{D}$ in the determination of s leaves the generality of the Eilon and Elmaleh (1968) study in doubt.

Buffa (1969) states the "...the normal distribution has been found to describe adequately many demand functions at the factory level...". In his 1965 edition, Buffa stated that the rational determination of safety stocks was a function of the probability distribution of demand combined with a decision regarding the acceptable probability of a stock-out. This is the rationale for the commonly accepted use of the standard deviation in the calculation of safety stocks. The desired protection against stockouts can be achieved by varying the number of standard deviations of demand used in calculating the desired safety stock. The safety stock is therefore a function of the variation in the demand.

Scarf (1960), Iglehart (1963) and Veinott and Wagner (1965) have discussed the optimality of (s,S) policies and the method of computing the optimal values of the two parameters. On the basis of these arguments, the (s,S) policy was chosen for testing the statistical policies in this project.

An (s,S) statistical inventory policy was tested using two methods of determining the reorder points, s. In both cases, EOQ's of Appendix I were used to determine the upper limit, S=s+Q.

One application of the (s,S) inventory policy used multipliers of the standard deviation to determine safety stocks. This inventory policy is referred to as the "standard deviation approach". This policy required the determination of an order point for each item. The order point was the sum of the expected demands during the replenishment lead time, plus a multiple of the standard deviation of demand during the replenishment lead time. The multiplier of the standard deviation was dependent upon the desired protection against stockouts during the replenishment cycle.

The listing and flow diagram of the computer subroutine used to implement the standard deviation statistical inventory policy is given in Appendix VI. This subroutine is labelled DCID4.

The second application of the (s,S) statistical inventory policy incorporated the use of service levels as outlined by Brown (1967). This policy is referred to as the "service level approach" because the safety stock was calculated in multiples of the standard deviation to provide a desired average annual service level. The order point, s, was the sum of the expected demand during the replenishment lead time, plus the safety stock, calculated in standard deviations to give an annualized service level.

For both applications of the (s,S) statistical policy, the upper limit, S, was the sum of the order point, plus the economic order quantity, EOQ. The order quantity was equal to the difference between the upper limit, S, and the actual inventory level at the end of the period. Replenishment orders were issued only if the stock on hand at the end of the review period was less than or equal to s. When replenishment orders were issued, the order quantities were greater than or equal to the applicable EOQ.

Appendix VII gives a listing and the flow diagram for the computer subroutine DCID5 which was used to implement the service level statistical inventory policy.

Both the standard deviation and the service level approaches to statistical inventory policies require the standard deviation of the demand for each component. This data cannot be derived from the demand distributions of the finished products as was done for the requirements planning Lot-for-Lot policy. The difference is that in the Lot-for-Lot case, there is a demand every period from all assemblies requiring the item, whereas the statistical policies precipitate intermittent demand for component parts.

A file was created to store the historical demands by period for each component. This data was used to calculate the standard error in the demand for each component.

The implementation of the statistical policies required some adjustments to the EOQ's for component parts. The EOQ's for some components were less than the EOQ's for the assemblies in which they were used, thus creating a materials shortage problem. Crowston, Wagner and Williams (1971) showed that for a system having a constant demand over an infinite horizon, the cost-minimizing ratio of the lot-size of a supply item to the lot-size of a demand item was a positive integer. This result was used in the implementation of the statistical policies to alleviate material shortage due to lot-size inconsistencies. A check was made of all components in an assembly to make sure that the EOQ of the component was at least equal to the lot-size of the assembly. If the EOQ of the component was less than the lot-size of the assembly, the lot-size of the component was increased to equal the lot-size of the assembly.

In summary, given the forecast demand, the application of the statistical inventory policies required the calculation of the desired safety stock and order point for each item. The upper control level was equated to the order point plus the EOQ for each inventory item. If the stock on hand at the end of each period was less than the order point, a replenishment order equal to the upper control limit less the stock on hand was issued. If the stock on hand at the end of the review period was larger than the order point, a replenishment order was not issued.

4.4 Combined Statistical-Requirements Planning Policies

Technically what is herein called a combined statistical-requirements planning policy is a form of material requirements planning. However, to avoid possible confusion the "combined" nomenclature has been adopted.

For this policy, the finished products are controlled by an (s,S) policy. This results in a production plan by period for these products. The time-series product explosion matrix is then implemented to determine the time-phased requirements of components which are produced on a Lot-for-Lot basis. This policy combines a statistical method for the items which have independent demands with discrete methods for the items which have dependent demands.

The implementation of this policy requires that the production plan for finished products be determined several periods in advance so that the production of components can be coordinated to facilitate the assembly of finished products. The longest lead time between the ordering of a raw material and the assembly of the finished product is the dominant factor in determining the time period for which the finished product assembly schedule must be committed.

The commitment to an assembly schedule for a finished product several periods in advance requires the calculation of safety stocks for the required lead time. The approach used to determine the safety stock was to take the square-root of the sum of the finished product demand variances for the number of periods committed. The order points also considered the mean forecast demand over the lead time. The order points for the finished products in the combined policy were very large in comparison to the other policies tested. This was partially balanced by the lack of safety stocks for component items.

The combined statistical-requirements planning policy was implemented by determining the safety stock and order point for each finished product. If the stock on hand was less than the order point, a replenishment order was issued for a quantity equal to the order point, plus an EOQ, less the stock on hand and, less the material on order. If the stock on hand was greater than the order point, no replenishment order was issued. For component items, production was planned by the Lot-for-Lot method to facilitate the time-phased finished product production plan.

Appendix VIII gives a listing and a flow diagram of the computer subroutines DCID3 used to implement the combined statistical-requirements planning policy.

Chapter Five

Research Methodology

5.1 Method of Making Decisions

The objective of this research project was to make a comparative study of the influence of inventory policies on the total cost of operating a multistage manufacturing system. Given this objective, it was imperative that whenever possible, the differences in results be attributable to the influence of the inventory policies. Production planning in terms of smoothing the production requirements and workforce levels interact with inventory policies. If production planning had been incorporated in the decision rules used in the experiment, the results would have been difficult to interpret. Therefore it was decided that the inventory policies would dominate the manufacturing system.

The production requirements were taken directly from the inventory policies and the workforce levels were adjusted to facilitate those production requirements on regular time. The workforce levels were rounded to the nearest integer, which resulted in the possibility of overtime or undertime, although these quantities were expected to be small. The workforce levels were calculated from the original production quantities resulting from the inventory decision rules. These production quantities were based on the assumption that there would be sufficient inventory of the component items to make the desired quantities. If there was insufficient inventory of component items to assemble the original production quantities, the lot-sizes were reduced and the workforce experienced idletime.

The requirements planning Lot-for-Lot inventory policies were expected to require relatively few labour adjustments due to the requirement to produce each product every period. Therefore the labour adjustment costs were expected to be relatively low for the tests which implement Lot-for-Lot inventory policies.

The statistical (s,S) inventory policies were expected to require many more labour adjustments due to the intermittent production requirements. It was previously noted that the weighted average economic lot-size of finished products was equal to approximately 1.9 months of base demand. The cost of hiring and firing an employee was $1,200.00, which was equivalent to 344 man-hours at regular time wages, or approximately 2.14 months wages on regular time. This analysis indicates that it would be less expensive to maintain a regular workforce for finished products than hire and fire as required. However, the decision was made that the workforce levels would be adjusted to facilitate the production quantities generated by the inventory policies, and this was adhered to. The above analysis is based on the requirements of the average product. The interaction between products at each stage was expected to smooth the labour adjustments to some extent. The labour adjustment costs for the statistical policies were expected to be larger than those for the Lot-for-Lot policies, but the magnitude of the differences was difficult to hypothesize.

The combined statistical-requirements planning policies were expected to require more labour adjustments than the requirements planning policies, due to the intermittent production of finished products, but less labour adjustments than the statistical policies, due to the interaction of the cascaded demands for component items from the finished products. Therefore the labour adjustment costs for the combined policies were expected to be higher than those for the requirements planning policies but less than those for the statistical policies.

5.2 Method of Comparing Results

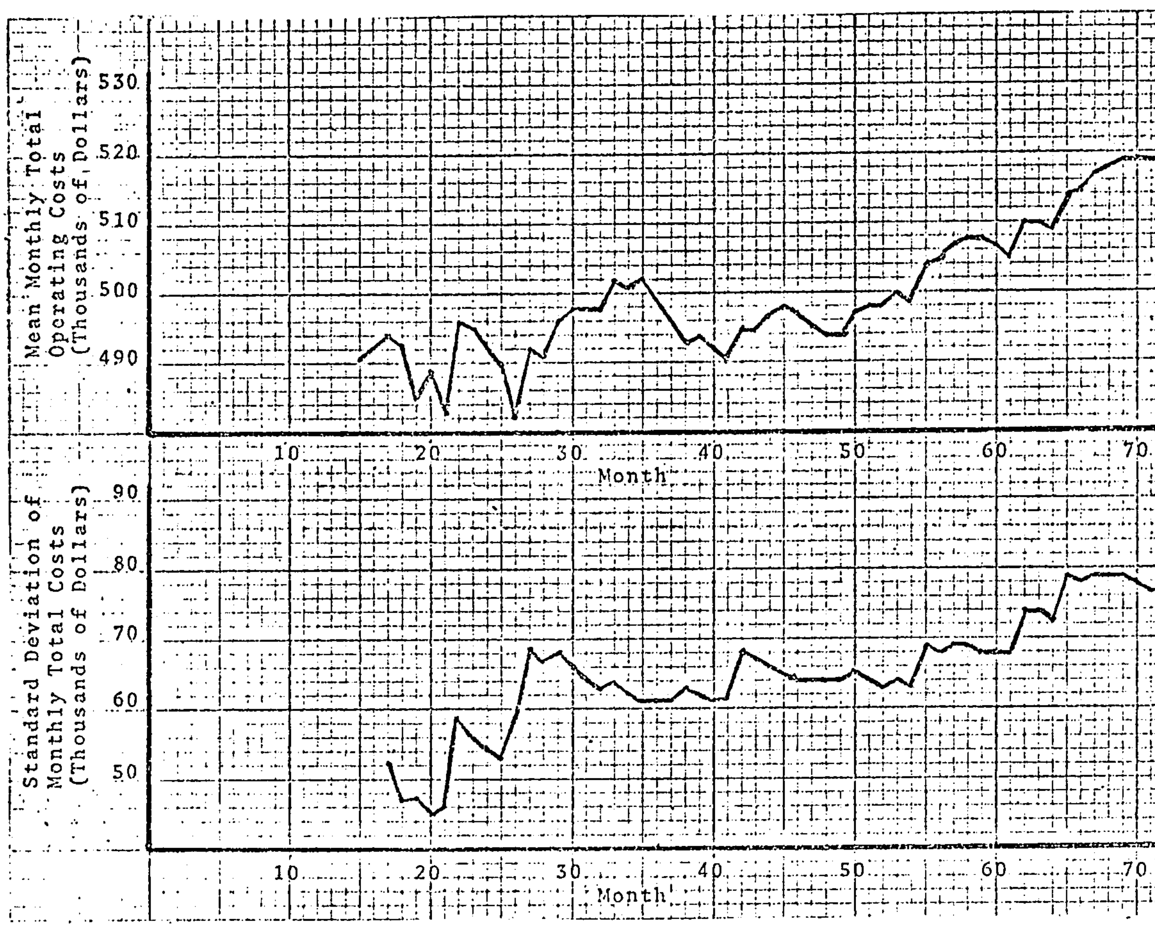
5.2.1 The Test Period

The dynamic interaction of the implementation of the inventory policies and the simulated operation of the manufacturing facility imposed a problem in comparing the results of the various tests. It would be virtually impossible to control the experiment so that the conditions were identical for all tests. The most obvious disparity was the inventory quantities at the end of the test period. To overcome this problem, the experiment was run for an extended time period on the assumption that the ending conditions would not significantly influence the results of the various tests.

An initialization period of 12 months was used to adjust the system to the various inventory policies and the costs of operations were accumulated from month 13 to the end of the test horizon. A graph of the mean monthly total operating costs and the standard deviation of the monthly total costs is shown in Figure 5.1. It can be seen from these graphs that aside from the increasing trend factor due to the overall growth in the demand for products, the mean and standard deviation of the monthly costs stabilized in 3 to 4 years after the initialization period. For this to occur, the process must have stabilized much earlier. It was therefore decided to make the comparisons on the basis of the

Figure 5.1

Typical Plot of Mean Monthly Total Operating Costs and Standard Deviation of Monthly Total Operating Costs by Month



accumulation of five year's data after the one year initialization period.

5.2.2 The Data Collected

Inventory policies do not take into consideration the costs of labour adjustments even though the labour requirements are directly influenced by inventory policies. Since the purpose of this research was to compare the total costs of operating the simulated system, the labour adjustment costs were included in the total costs. The inclusion of these labour adjustment costs should indicate the total influence of the inventory policies on the operating system.

The total operating costs were collected in five major groups to facilitate the analysis. These groups were: (1) labour adjustment costs - hire, fire and transfer costs; (2) payroll costs; (3) inventory carrying costs; (4) backorder costs, and; (5) ordering costs. The labour hours were also accumulated in five categories, namely: (1) set-up hours; (2) running hours; (3) standard time hours; (4) overtime hours, and; (5) idletime hours. The collection of this data was useful for determining the differences in the characteristics of the various inventory policies tested.

5.2.3 The Test Treatments

The implementations of the inventories policies were tested under two treatment conditions: deterministic and stochastic demand. For the deterministic demand tests, the future demands of the finished products, and therefore all components, were known by period with certainty for as distant a horizon as required for planning purposes. The deterministic tests were made for two reasons. The first was to check the computer programs to ensure they were functioning properly. The second reason for the deterministic tests was to determine the cost of uncertainty associated with the randomness of the stochastic demands.

The stochastic demands for finished products were generated by modifying the forecast demands by a normally distributed random coefficient. The standard deviations of the stochastic demands were proportional to the mean forecast demand, that is each product had a constant coefficient of variation. The coefficients of variation for the finished products were unique to each finished product.

5.2.4 Search for Minimum Total Costs

The comparison of the implementation of the various inventory policies required that the minimum total cost of operations be found under each test condition. The comparison of minimum total costs is unbiased and allows each policy to be tested at its best performance level. The total costs determined from testing a policy under deterministic demand are by definition invariant from test to test. However, the tests conducted under stochastic demands required that safety stocks be carried to protect against backorders. The variations in the demands for finished products creates variations in the demands for all components, and therefore, it may be desirable to carry safety stocks at all inventory levels. Plossl and Wight (1971) suggest that the largest safety stocks should be carried for finished product and raw material items. The finished product safety stocks provide protection against backorders and the raw materials safety stocks provide production flexibility.

The amount of safety stock carried for each item influences the total cost of operating the system. Therefore the safety stocks for each inventory item should be chosen to minimize the total operating costs. An exhaustive evaluation of all reasonable safety stock levels for the 45 inventory items in the simulated system would require n^{45} computer runs for each stochastic policy tested, where n is the number of safety stock levels tested for each inventory item. If only three levels of safety stocks are chosen for each inventory item, approximately 2.95 x 10^{21} computer runs would be required. This is obviously infeasible.

If the inventory items at each inventory level are treated homogeneously in terms of the safety stock levels, then the exhaustive evaluation of all reasonable safety stock levels for each inventory level requires n^4 computer runs. Three levels of safety stocks per inventory level are insufficient to give a reasonable assurance of near minimum total operating costs. A better indication of the minimum total costs would be obtained if 5 or 7 safety stock levels were tested at each inventory level. However, the exhaustive testing of 7 safety stock levels at each inventory level would require $7^4 = 2401$ computer runs. The initial test runs on a CDC6400 computer cost approximately $20.00 per run. Therefore approximately 2400 runs per stochastic policy tested was also infeasible.

Exhaustive testing of all reasonable safety stock levels at all inventory levels is not necessary to locate the minimum total cost combination. Wilde (1964) discusses several optimum seekin g methods, two of which were of interest in this study. The steepest ascent (descent) search method and the Hooke-Jeeves pattern or direct search method. Due to the high cost per computer run, it was decided to use a "hands on" steepest descent approach. The strategy was to choose a reasonable starting combination of safety stocks and use the slopes of the total cost surface to locate the minimum total cost combination of safety stock levels.

The total cost surface was expected to be relatively flat and well-behaved so that a coarse grid could be used for the search procedure. A reasonable scale for determining safety stocks appeared to be in terms of the standard deviation of demand. A convenient range on that scale was from 1.0 to 4.0 standard deviations in step sizes of 0.5 standard deviations. This allowed for 7 grid points at each safety stock level.

The combined statistical-requirements planning policy only required safety stocks for finished products and therefore a search was needed at only one level. For consistency, the same scale of safety stock levels was used for this policy as for those policies having safety stocks at four levels.

The statistical policy using the annual service level approach suggested by Brown (1967) required a different scale than that used for the other policies. The service level approach uses the desired service level as the decision variable rather than a multiple of the standard deviation, and therefore the scale for the search procedure had to be in terms of the desired service level. It was decided that for the service level tests, the grid points would be service levels of 0.90, 0.93, 0.95, 0.97, 0.99, 0.995 and 0.999. These levels encompass the range of service levels that are usually considered desirable.

5.3 Forecast of Finished Product Demands

A forecast of finished product demands was required for the planning of production quantities. The simulation game has an internal forecasting system which gave a perfect forecast of demand. The actual demands for the finished products were equal to the forecast demand plus a normally distributed random variable.

An exponential smoothing forecasting routine was introduced which searched for the smoothing parameters minimizing the standard error of the forecast demands. This forecasting routine required a total run time of several times the regular computation time.

The forecast demands from the exponential smoothing routine were similar to the internally generated forecasts but the standard errors of the forecasts were somewhat larger for the exponential smoothing method. This was to be expected since the exponential smoothing routine was trying to duplicate the internal forecasts. Since the forecasting routines generated comparable forecasts, it was decided to use the internal forecasts to save computation time.

5.4 Inventory Policy/Demand Treatments Tested

The requirements planning Lot-for-Lot inventory policy was tested under both deterministic and stochastic demand treatments. The deterministic test was used primarily to determine a base for evaluating the cost of uncertainty. It was also used to verify the operation of the computer program.

The requirements planning Lot-for-Lot inventory policy tested under stochastic demand was one of the original tests to be conducted. The results are to be compared with those from the statistical policies.

The statistical policies were tested under stochastic demand conditions only. The "standard deviation" statistical policy used the standard deviation of the demand for inventory items at all levels to determine the safety stock levels. The "service level" statistical policy used the desired annual service levels of all inventory items to set the safety stock levels. Brown (1967) claims that the service level approach is more economical than the standard deviation approach and that it gives a consistency of service which is lacking in the standard deviation method.

The combined statistical-requirements planning inventory policies were tested under both deterministic and stochastic demand treatments. The deterministic demand treatment again served the dual role of verifying the operation of the computer program and evaluating the cost of uncertainty.

The combined policy was tested under stochastic demand to evaluate the principle of independent/dependent demand postulated by Orlicky (1970). According to the theory proposed by Plossl and Wight (1970) and Orlicky (1970), the combined policy should lead to lower operating costs than either the requirements planning or statistical inventory policies.

5.5 Experimental Design

The comparison of the total costs of implementing the various inventory policies was based on the minimum total costs for each test conducted. The mean monthly total costs and the standard deviation of those monthly total costs were also collected and recorded. This facilitated the testing for statistically significant differences between the minimum mean monthly total costs. Therefore statements can be made regarding the confidence that differences do exist between policies.

5.6 Development of Comparison Formulae

A situation may arise when it would be advantageous to have some insight into relative efficiencies of alternative lot-sizing methods. Of particular interest is a comparison of EOQ lot-sizing and requirements planning methods. A Lot-for-Lot, lot-sizing method is useful for this comparison. Assume that the demand for an inventory item is known in interval quantities for at least a one year planning horizon. The use of a Lot-for-Lot method will require a set-up every period to produce the product, whereas the EOQ method will require a variable number of set-ups. Assume that the inventory carrying cost is levied against the average inventory in stock for both lot-sizing methods.

The total cost equation for the requirements planning Lot-for-Lot method is:

$$(5.1) \qquad TC_{RP} = N_1 S + \frac{A}{2N_1} Ci + CA.$$

where N_1 is the number of intervals or operating cycles per year. For the requirements planning Lot-for-Lot methods to be more economical than the EOQ method, the total cost from Equation (5.1) must be less than that from Equation (I.1), that is

$$(5.2) \qquad N_1 S + \frac{ACi}{2N_1} + CA < \frac{AS}{Q} + \frac{QCi}{2} + CA.$$

Simplifying Equation (5.2) gives:

$$(5.3) \quad S\left(N_1 - \frac{A}{Q}\right) < \frac{Ci}{2}\left(Q - \frac{A}{N_1}\right).$$

This can be reduced to:

$$(5.4) \quad \frac{S}{Q}(N_1Q - A) < \frac{Ci}{2N_1}(N_1Q - A).$$

or

$$(5.5) \quad N_1 < \frac{QCi}{2S}.$$

Substituting for Q and rearranging the variables gives

$$(5.6) \quad N_1 < \frac{A}{Q}.$$

Equation (5.6) indicates that requirements planning Lot-for-Lot, lot-sizing will result in a relative advantage if the number of operating cycles is less than the number of set-ups required by the economic lot-size method. A greater number of set-ups than operating cycles would require multiple set-ups in some periods. This would increase the quantity of inventory in stock to greater than a EOQ lot-size and the inventory carrying costs would be increased accordingly.

If it is assumed that the inventory item under consideration is a finished product and that inventory carrying costs are calculated on the value of the inventory at the end of each period, the Lot-for-Lot method will supply the deterministic interval demand as it occurs, and there will be no inventory carrying costs. Conversely, the EOQ method would still carry an average inventory of approximately half the lot-size. For the requirements planning Lot-for-Lot method to be more economical than the EOQ method:

(5.7) $$N_2S + CA < \frac{AS}{Q} + \frac{QCi}{2} + CA$$

where N_2 is the number of operating cycles per year. This can be simplified to:

(5.8) $$N_2 < \frac{A}{Q} + \frac{QCi}{2S}.$$

But:

(5.9) $$\frac{QCi}{2S} = \frac{A}{Q}.$$

Therefore:

(5.10) $$N_2 < \frac{2A}{Q}$$

A comparison of Equation (5.6) and (5.10) indicates that the critical number of operating cycles is twice as large when there is no inventory carrying charge. This is to be expected since when a charge is levied for carrying the average inventory, the critical number of operating cycles is equal to the number of set-ups for the EOQ method. The minimum cost for the EOQ method occurs when the set-up costs are equal to the inventory carrying costs. Since there are no carrying costs assumed for the requirements planning Lot-for-Lot method in Equation (5.10), it is logical that $N_2 = 2N_1$. The implication of Equation (5.10) is that if inventory costs are based on the ending inventory of each period, the requirements planning Lot-for-Lot method will be a less expensive inventory control procedure than the EOQ method provided that the number of operating cycles is less than twice the number of set-ups required by the EOQ method.

The break-even equations developed above may be considered as a guide to the selection of an appropriate lot-sizing method for any given inventory item. However, the selection of a lot-sizing method appropriate for a single inventory item does not imply that it is the best procedure for all inventory items at all stages in a multistage multiproduct operating system.

5.7 Sensitivity Analysis

The formulae developed in the previous section indicated that the length of the operating cycle influences the applicability of inventory models to production systems. It was therefore decided to test the application of the previously selected inventory policies on the simulated factory to determine if the predicted results would be confirmed.

The requirement of a Lot-for-Lot inventory policy to have production runs in every operating cycle suggests that as the operating cycle is shortened, the number of set-ups in any given time period increases, and the unit cost per item assembled increases if the lot-size is less than an EOQ.

It was previously noted that the weighted average base monthly requirements for finished products was approximately one-half of an EOQ lot-size. The use of components in several products may result in the monthly demand for some components to be greater than the EOQ for those components. It was suspected that on an average over the total system, the monthly demand for inventory items was approximately equal to the EOQ lot-size.

To test the sensitivity of the inventory policies to the length of the operating cycle, the operating cycle was reduced to an equivalent time period of one week. The parameters of the game were revised to accommodate the reduced operating cycle. In particular, the demand functions were adjusted so that the total demands over the six year horizon used in the experiment were comparable to the total demands using a monthly cycle. A year was considered to be composed of 48 weeks for the weekly operating cycle tests.

Chapter Six

Results and Analysis

The governing of production and workforce decision rules by inventory control policies was intended to demonstrate the difference in total operating expenses using several inventory policies so that conclusions could be made about the relative effectiveness of these policies.

Some players of the "Production Control Project-DCIDE" simulation game have been able to develop decision rules which resulted in total costs of approximately $4.2 million for the first year and five year total costs of $23 to $24 million[1]. In these cases, the demands for finished products were stochastic. These costs cannot be used as a strict comparison with the results of this research due to changes which have been made in the game but they do provide a guideline. The applications of the inventory policies did not consider trade-offs usually associated with production

1. This information was received from R. R. Britney in a private communication.

control such as the use of overtime or undertime and changes in the level of the workforce. Consequently, the costs resulting from the research were expected to be higher than those normally associated with good management of the game.

The operating cost for the cases tested are shown in Table 6.1. The range of total operating cost is relatively large, especially under stochastic demand functions. This range of costs demonstrates the relative effectiveness of the inventory policies tested.

6.1 Requirements Planning - Deterministic Demand

The application of the requirements planning Lot-for-Lot inventory policy under demand certainty to the simulated operating system of "Production Control Project-DCIDE" resulted in a total operating cost of $27.157 million for the 60 month test period. The major component of the total cost was the payroll cost which accounted for $24.430 million. The only other cost of major significance was the inventory holding cost of $2.286 million. The cost of $0.417 million for hiring, firing, and transferring workers indicates that there were few labour force adjustments during the period being considered. If each of the 8 raw materials were ordered in each of the 60 periods at

Table 6.1

Cost Comparisons for the Monthly Operating Cycle

	Labour Adjustment Costs	Payroll Costs	Inventory Carrying Costs	Backorder Costs	Ordering Costs	Total Costs	Total Costs less Labour Adjustment Costs
Requirements Planning - Deterministic Demand	$ 417,400.00	$24,430,280.00	$ 2,286,453.74	$ 0.00	$ 23,200.00	$27,157,333.74	$26,739,933.74
Requirements Planning - Stochastic Demand	$ 2,964,200.00	$24,801,362.25	$ 3,314,406.06	$ 22,400.00	$ 22,600.00	$31,124,968.31	$28,160,768.31
Combined Stat.- Req. Plan - Deterministic Demand	$ 6,017,000.00	$21,177,297.75	$ 3,907,726.87	$ 0:00	$ 19,900.00	$31,121,924.62	$25,104,924.62
Combined Stat.- Req. Plan - Stochastic Demand	$ 5,243,900.00	$21,465,847.37	$ 4,620,076.34	$ 26,880.00	$ 19,650.00	$31,376,353.72	$26,132,453.72
Statistical (s,S) - Standard Deviations	$11,222,800.00	$23,427,262.25	$ 7,441,817.11	$ 278,880.00	$ 14,700.00	$42,385,429.36	$31,162,629.39
Staristical (s,S) - Service Levels	$10,178,200.00	$24,194,150.75	$ 6,371,969.87	$ 617,400.00	$ 14,700.00	$41,376,420.62	$31,198,220.62

cost of $50.00 per order placed, the maximum ordering charge would have been $24,000.00 The ordering costs of $23,200.00 is $800.00 less than the maximum possible charge for ordering costs, indicating that there were no demands for some products in certain periods. There were no backorder charges as would be expected under demand certainty.

The total costs of over $27 million indicate that the use of requirements planning Lot-for-Lot inventory policies as they were applied in this research are not effective production scheduling rules. The addition of demand risk will increase the total costs of operations even more as will be described in the following section. Even though the total costs may not be as low as could be expected from the use of more general production scheduling techniques, the costs are of the same order of magnitude as the $23 to $24 million total costs attained by other players.

6.2 Requirements Planning-Stochastic Demand

As shown in Table 6.1, the minimum total operation cost using requirements planning Lot-for-Lot inventory policies under stochastic demand conditions was $31.125 million. This is approximately $3.968 million more than the total cost under deterministic demand conditions. The cost associated with risk is approximately $0.8 million per year and represents 12.9 percent of the total operating cost.

A comparison of the observed costs under deterministic and stochastic demands is made in Table 6.2. The largest difference in cost was due to adjustments of the labour force. The other major cost difference was in the inventory holding costs. These two categories together accounted for over 92 percent of the difference in total costs. The only category which did not show an increase under stochastic demand from deterministic demand was the ordering cost for raw materials.

The combinations of safety stock levels surrounding the minimum total cost is shown in Table 6.3. The safety level of 2.5 standard deviations for finished goods indicates large safety stocks of these items. The cost of backordered demand is very large relative to the cost of carrying inventory, therefore heavy protection is warranted. The 1.5 standard deviations of safety stock carried for subassemblies indicates that the level of protection required for subassemblies is less than that for finished goods. However the safety stock for parts at 2.5 standard deviations and for raw materials at 4.0 standard deviations means that large stocks of these items are warranted, presumably due in part to the lesser item values. The

Table 6.2

Cost Comparison for Requirements Planning

(Monthly Operating Cycle)

	Deterministic Demand (1)	Stochastic Demand (2)	Difference (2)-(1)
Labour Adjustment Costs	$ 417,400.00	$ 2,964,200.00	$ 2,546,800.00
Payroll Costs	$24,430,280.00	$24,801,362.25	$ 371,082.25
Inventory Carrying Costs	$ 2,286,453.74	$ 3,314,406.06	$ 1,027,952.32
Backorder Costs	$ 0.00	$ 22,400.00	$ 22,400.00
Ordering Costs	$ 23,200.00	$ 22,600.00	$ -600.00
Total Costs	$27,157,333.74	$31,124,968.31	$ 3,967,634.57

Table 6.3

Total Operating Costs for Requirements Planning

at Selected Safety Stock Levels

(Monthly Operating Cycle)

(Costs in Millions of Dollars)

		ZFG = 2.0			ZFG = 2.5			ZFG = 3.0		
		ZSA = 1.0	ZSA = 1.5	ZSA = 2.0	ZSA = 1.0	ZSA = 1.5	ZSA = 2.0	ZSA = 1.0	ZSA = 1.5	ZSA = 2.0
ZPT = 2.0	ZRM = 3.5	31.435	31.237	31.195	31.172	31.137	31.160	31.230	31.244	31.273
	ZRM = 4.0	31.370	31.195	31.176	31.136	31.109	31.152	31.203	31.227	31.274
ZPT = 2.5	ZRM = 3.5	31.385	31.179	31.142	31.177	31.126	31.135	31.237	31.231	31.257
	ZRM = 4.0	31.316	31.150	31.133	31.117	31.102*	31.134	31.189	31.213	31.256
ZPT = 3.0	ZRM = 3.5	31.396	31.221	31.153	31.205	31.175	31.150	31.286	31.282	31.278
	ZRM = 4.0	31.329	31.162	31.139	31.142	31.121	31.145	31.217	31.239	31.277

* indicates minimum cost combination

ZFG = finished products safety stock in standard deviations
ZSA = subassemblies safety stock in standard deviations
ZPT = parts safety stock in satandard deviations
ZRM = raw materials safety stocks in standard deviations

search for the minimum cost combination was not carried past 4.0 standard deviations for raw materials due to the inconceivability of carrying every stock item at higher protection levels. It is doubtful that a significantly lower total cost would have been found at higher levels of safety stock for raw materials.

6.3 Combined Statistical-Requirements Planning-Deterministic Demand

The combined statistical-requirements planning model tested under deterministic demand resulted in a total cost of $31.122 million for the 60 month test period. This was approximately $3.965 million more than the requirements planning Lot-for-Lot test under deterministic demand.

A comparison of the component costs between the implementation of requirements planning and combined statistical-requirements planning policies is shown in Table 6.4. The difference in labour adjustment costs amounted to $5.600 million. This indicates that the use of economic order quantities for finished products resulted in substantial fluctuations in the workforce whereas the use of requirements planning allowed the workforce to remain relatively constant. Significantly, the use of the combined policy resulted in a $3.253 million dollar saving in payroll costs. It will be shown below that this difference is almost entirely attributable

Table 6.4

Cost Comparison Between Requirements Planning and Combined Policies under Deterministic Demand (Monthly Operating Cycle)

	Requirements Planning (1)	Combined Stat.-Req.Plan. (2)	Difference (2)-(1)
Labour Adjustment Costs	$ 417,400.00	$ 6,017,000.00	$ 5,599,600.00
Payroll Costs	$24,430,280.00	$21,177,297.75	$-3,252,982.25
Inventory Carrying Costs	$ 2,286,453.74	$3,907,726.87	$ 1,621,273.13
Backorder Costs	$ 0.00	$ 0.00	$ 0.00
Ordering Costs	$ 23,200.00	$ 19,900.00	$ -3,300.00
Total Costs	$27,157,333.74	$31,121,924.62	$ 3,964,590.88

to a reduction in set-up hours.

The inventory carrying costs were approximately $1.621 million more when the combined policy was used than for the requirements planning Lot-for-Lot policy. An increase in inventory carrying costs was expected as the use of economic order quantities requires that finished goods be carried in inventory. In addition, the inventories at preceding levels will also be increased due to the larger lot-sizes.

The deterministic demand condition facilitated the planning of finished goods inventory to eliminate backorder charges in both test conditions. The remaining difference in cost was a $3,300 saving in ordering costs resulting from the application of the combined inventory policy.

6.4 Combined Statistical-Requirements Planning-Stochastic Demand

The combined statistical-requirements planning policy tested under stochastic demand resulted in a minimum total cost of $31.376 million for the 60 month test period. The total operating costs at the safety stock factors tested are shown in Table 6.5. The values shown indicate that the cost surface for this test treatment is not as flat as for the requirements planning Lot-for-Lot test under uncertainty. The sharp

Table 6.5
Total Operating Costs for the Combined Policy at Selected Safety Stock Levels
(Monthly Operating Cycle)

Safety Factor z in Standard Deviations	Total Cost
0.5	$33,474,201.43
1.0	$32,769,995.10
1.5	$32,757,987.66
2.0	$31,376,353.72*
2.5	$31,653,980.07
3.0	$31,998,668.72
3.5	$31,946,493.98
4.0	$32,074,563.34

* indicates minimum total cost

rise in total operating costs at safety factors less than 2.0 are primarily due to backorder costs.

A comparison of the total operating costs for the combined statistical-requirements planning policy under deterministic and stochastic demand is given in Table 6.6. The difference in total operating costs of only $0.254 million is substantially different than the cost of uncertainty associated with the requirements planning policy. As can be seen from Table 6.6, the labour adjustment costs were $0.773 million less under the stochastic test conditions than under the deterministic test conditions. The safety stock appears to have served the dual purpose of protecting against backorders and smoothing the workforce adjustments.

The payroll costs were approximately $0.289 million higher under the stochastic test conditions, which may be due to materials not always available to facilitate the desired production quantities. As expected, the inventory carrying costs were higher under the stochastic test conditions due to the additional inventory carried in safety stocks.

The total operating costs for the requirements planning and the combined statistical-requirements planning policies are compared in Table 6.7. Contrary to the predictions of Plossl and Wight (1970) and Orlicky (1970), the combined policy did not result in the least

Table 6.6

Cost Comparison for Combined Policy

(Monthly Operating Cycle)

	Deterministic Demand (1)	Stochastic Demand (2)	Difference (2)-(1)
Labour Adjustment Costs	$ 6,017,000.00	$ 5,243,900.00	$ -773,100.00
Payroll Costs	$21,177,297.75	$21,465,847.37	$ 288,549.62
Inventory Carrying Costs	$ 3,907,726.87	$ 4,620,076.34	$ 712,349.47
Backorder Costs	$ 0.00	$ 26,880.00	$ 26,880.00
Ordering Costs	$ 19,900.00	$ 19,650.00	$ -250.00
Total Costs	$31,121,924.62	$31,376,353.71	$ 254,349.09

Table 6.7

Cost Comparison Between Requirements Planning and Combined Policies under Stochastic Demand

(Monthly Operating Cycle)

	Requirements Planning (1)	Combined Stat.-Req. Plan. (2)	Difference (2)-(1)
Labour Adjustment Costs	$ 2,964,200.00	$ 5,243,900.00	$ 2,279,700.00
Payroll Costs	$24,801,362.25	$21,465,847.37	$-3,335,514.88
Inventory Carrying Costs	$ 3,314,406.06	$4,620,076.34	$ 1,305,670.28
Backorder Costs	$ 22,400.00	$ 26,880.00	$ 4,480.00
Ordering Costs	$ 22,600.00	$ 19,650.00	$ -2,950.00
Total Costs	$31,124,968.31	$31,376,353.71	$ 251,385.40

total cost. The combined policy had a total operating cost $0.251 million higher than the requirements planning policy total cost. However, when the labour adjustment costs are removed from the comparison, the combined policy results in a much lower cost. Labour adjustment costs are not normally included in discussions regarding inventory policies, and even though the results show how significantly inventory policies influence labour adjustment costs, some comparisons of the relative effectiveness of inventory policies should exclude labour adjustment costs. It is interesting to note that the difference between the total costs less labour adjustment costs for the requirements planning and the combined statistical-requirements planning tests were very similar under both deterministic and stochastic demand treatments. These differences were $1.635 million for the deterministic treatments and $2.028 million for the stochastic treatments.

The payroll costs were approximately $3.336 million less for the combined policy than the requirements planning policy. As explained previously, this difference is due to the reduced number of set-up hours resulting from the application of the combined policy. The inventory carrying costs were approximately $1.306 million higher for the combined policy than the requirements planning policy, presumably due to the additional carrying costs resulting from the larger lot sizes. The backorder costs and the ordering costs were similar for both types of policies.

6.5 Statistical Inventory Policies-Stochastic Demand

The use of statistical inventory policies under stochastic demand resulted in considerably higher costs than was observed for either the requirements planning or the combined policies. When safety stocks in multiples of standard deviations were used, the total cost was $42.385 million. The service level approach resulted in a total cost of $41.376 million. These total costs are in excess of 30 percent larger than the total costs of operations using requirements planning policies.

Table 6.8 shows the comparison of the costs associated with requirements planning and the two statistical approaches: standard deviations and service levels. The pattern of the differences in the cost components is similar for both statistical treatments. The service level approach is less costly than the standard deviation approach for all cost components except backorder charges. This was anticipated as the service level approach tends to reduce inventories and increase backorders.

The two major categories of cost differences between the requirements planning Lot-for-Lot and statistical inventory approaches are the labour adjustment

Table 6.8

Cost Comparison Between Requirements Planning and Statistical Policies under Stochastic Demand (Monthly Operating Cycle)

	Requirements Planning (1)	Statistical Std. Dev. (2)	Difference (2)-(1)	Statistical Ser. Levels (3)	Difference (3)-(1)
Labour Adjustment Costs	$ 2,964,200.00	$11,222,800.00	$ 8,258,600.00	$10,178,200.00	$ 7,214,000.00
Payroll Costs	$24,801,362.25	$23,427,262.25	$-1,374,100.00	$24,194,150.75	$ -607,211.50
Inventory Carrying Costs	$ 3.314,406.06	$ 7,441,817.11	$ 4,127,411.05	$ 6,371,969.87	$ 3,057,563.81
Backorder Costs	$ 22,400.00	$ 278,880.00	$ 256,480.00	$ 617,400.00	$ 595,000.00
Ordering Costs	$ 22,600.00	$ 14,700.00	$ -7,900.00	$ 14,700.00	$ -7,900.00
Total Costs	$31,124,968.31	$42,385,429.36	$11,260,491.05	$41,376,420.62	$10,251,452.31

charges and the inventory carrying charges. The workload in manhours fluctuated much more when the production was undertaken in economic lot sizes than when quantities were made during each period to meet the monthly demands. These labour adjustment costs accounted for nearly 75 percent of the difference in the total costs between the requirements planning and statistical tests. The inventory carrying costs were approximately twice as large for the statistical tests as for the requirements planning tests.

The backorder charges were considerably higher for the statistical tests than for the requirements planning tests although the cost differences for backorder charges were a minor portion of the total cost difference. The payroll charges were lower for both statistical tests than for the requirements planning application. This reduction was due primarily to reduced set-up hours for the statistical applications.

Table 6.9 is similar to Table 6.8 except that the combined policy rather than the requirements planning policy is compared to the statistical policies. The comparisons are similar with the exception of the differences in payroll costs. Whereas the requirements planning policy had higher payroll costs than both of the statistical policies, the combined policy had lower payroll costs than both statistical policies.

Table 6.9

Cost Comparison Between Combined

and Statistical Policies under Stochastic Demand

(Monthly Operating Cycle)

	Combined (1)	Statistical Std. Dev. (2)	Difference (2)-(1)	Statistical Ser. Levels (3)	Difference (3)-(1)
Labour Adjustment Costs	$ 5,243,900.00	$11,222,800.00	$ 5,978,900.00	$10,178,200.00	$ 4,934,300.00
Payroll Costs	$21,465,847.37	$23,427,262.25	$ 1,961,414.88	$24,194,150.75	$ 2,728,303.38
Inventory Carrying Costs	$ 4,620,076.34	$ 7,441,817.11	$ 2,821,740.77	$ 6,371,969.87	$ 1,751,893.53
Backorder Costs	$ 26,880.00	$ 278,880.00	$ 252,080.00	$ 617,400.00	$ 590,520.00
Ordering Costs	$ 19,650.00	$ 14,700.00	$ -4,950.00	$ 14,700.00	$ -4,950.00
Total Costs	$31,376,353.71	$42,385,459.36	$11,009,105.65	$41,376,420.62	$10,000,066.91

The total costs of combinations of safety stocks surrounding the minimum total cost combination are shown in Tables 6.10 and 6.11 for the standard deviation and service level approaches respectively. In both cases, a pattern of protection levels similar to the one found for the requirements planning test was in evidence. The minimum total cost combinations had a high safety stock protection for finished products and raw materials and lower safety stock protection for subassemblies and parts. In the requirements planning case, the subassemblies had the least safety stock protection of all levels whereas the parts had the least protection for the standard deviation test. The subassemblies and parts had equal safety stock protection in the case of the service levels.

The slopes increased monotonically along all four axes from the minimum total costs found. This suggests that the absolute minima are near the observed minima and not appreciably different in value.

The vacant cells in Tables 6.10 and 6.11 attest the efficiency of the search procedure. The requirement of the value selected as the minimum on the cost surface was that all neighbouring costs be higher than the minimum. The three search procedures which were conducted on four dimensional grids required a total of approximately 300

Table 6.10

Total Operating Costs for Statistical Standard Deviations at Selected Safety Stock Levels

(Monthly Operating Cycle)

(Costs in Millions of Dollars)

		ZFG = 2.5			ZFG = 3.0			ZFG = 3.5		
		ZSA = 1.0	ZSA = 1.5	ZSA = 2.0	ZSA = 1.0	ZSA = 1.5	ZSA = 2.0	ZSA = 1.0	ZSA = 1.5	ZSA = 2.0
ZPT = 0.5	ZRM = 2.0					47.329				
	ZRM = 2.5					46.524	50.879		44.644	48.946
	ZRM = 3.0					47.576	51.270		47.344	49.626
ZPT = 1.0	ZRM = 2.0					44.497	43.602		45.676	45.113
	ZRM = 2.5	43.433	42.889		43.247	42.385*	44.963		44.626	44.115
	ZRM = 3.0					45.639	45.284		45.390	45.279
ZPT = 2.0	ZRM = 2.0									
	ZRM = 2.5	43.383	44.689		45.058	44.329				
	ZRM = 3.0									

* indicates minimum cost combination

ZFG = finished products safety stock in standard deviations
ZSA = subassemblies safety stock in standard deviations
ZPT = parts safety stock in standard deviations
ZRM = raw materials safety stock in standard deviations

Table 6.11

Total Operating Costs for Statistical Service Levels at Selected Safety Stock Levels

(Monthly Operating Cycle)

(Costs in Millions of Dollars)

		PFG=0.97			PFG=0.99			PFG=0.995		
		PSA=0.95	PSA=0.97	PSA=0.99	PSA=0.95	PSA=0.97	PSA=0.99	PSA=0.95	PSA=0.97	PSA=0.99
PPT = 0.95	PRM = 0.97									
	PRM = 0.99		43.620		44.181	46.350	42.861			
	PRM = 0.995		46.537			44.704	42.204			45.136
PPT = 0.97	PRM = 0.97	47.287	45.982	46.581	46.896	45.270	47.965	48.626	45.673	46.291
	PRM = 0.99	43.251	45.767	45.594	44.149	41.376*	44.711	44.917	44.405	43.731
	PRM = 0.995		45.430			46.220	42.008			44.766
PPT = 0.99	PRM = 0.97									
	PRM = 0.99		48.034		45.646	46.087	48.044			
	PRM = 0.995		44.128			45.456				

* indicates minimum cost combination

PFG = finished products safety stock service level
PSA = subassembly safety stock service level
PPT = parts safety stock service level
PRM = raw materials safety stock service level

runs. Thus approximately four percent of all possible safety stock combinations were tested. It was usually possible to straddle the minimum costs fairly quickly and then the gradients were followed to the minimum cells. Consequently, many cells surrounding the minimum were not evaluated since the gradients indicated that they were unlikely candidates for the minimum total costs.

6.6 Total Costs Less Labour Adjustment Costs

Some interesting results are shown in Table 6.1 when the labour adjustment costs have been removed from the total costs. Labour adjustment costs are usually associated with production control rather than inventory control, therefore the comparison excluding these costs may give a better reflection of the differences related to the implementation of inventory policies.

The difference in total cost between the $25.105 million for the combined policy and the $26.740 million for requirements planning under deterministic demand conditions is in the expected direction. The constraints of the game require the demand for finished products to be in blocks rather than in individual units which hampers the effectiveness of the economic order quantities. The economic order quantities were frequently equal to approximately two months demand so the statistical approach for finished products may be appreciably in error. The difference in total cost between the requirements planning and combined statistical-requirements planning approaches is not sufficient to make any definite statement, but there is a strong indication that the combined policy approach may result in lower operating costs if the labour adjustment costs are ignored.

The difference in the total costs between requirements planning, combined statistical-requirements planning and statistical inventory policies tested under stochastic demand conditions are considerably reduced if the labour adjustment costs are not included. As previously noted, the total cost less labour adjustment costs for the combined policy was $2.029 million less than the same cost for the requirements planning Lot-for-Lot policy under stochastic demand. The total costs less labour adjustment costs for the statistical policies were both approximately $31.2 million which was $6.2 more than the combined policy and $3 million more than the requirements planning policy. Therefore, under the experimental test conditions, the statistical policies resulted in the largest total costs, with or without the inclusion of labour adjustment costs.

6.7 Comparison of Total Man-hours

Table 6.12 shows the breakdown of the total man-hours used in the test conditions being compared. Two differences between tests are worth noting. The first is that the set-up hours for the two requirements planning tests are approximately double the set-up hours for the other tests. These differences of approximately 800,000 man-hours result in a cost savings of the order of $2.8 million for the five year term. The other major difference is in the idle man-hours. The two statistical tests employed is excess of 500,000 idle man-hours as compared to 60,000 idle man-hours for the requirements planning - stochastic demand test condition and 4,000 idle man-hours for the combined policy tested under stochastic demand. These idle man-hours are primarily due to the inavailability of materials to produce the products scheduled.

6.8 Tests for Significant Differences

The mean and standard deviation of the monthly total costs are shown in Table 6.13 by the type of policy tested. These statistics were collected to test for significant differences between the total operating costs resulting from the application of the inventory policies tested.

Table 6.12

Comparison of Labour Hours

for the Inventory Policies Tested

(Monthly Operating Cycle)

	Standard Time Hours	Overtime Hours	Set-Up Hours	Run Time Hours	Idle Hours
Requirements Planning - Deterministic Demand	6,975,040.0	3,360.0	1,639,165.0	5,335,442.0	3,793.0
Requirements Planning - Stochastic Demand	7,081,920.0	2,789.0	1,612,602.0	5,410,587.0	61,520.0
Combined Stat.-Req. Plan. - Deterministic Demand	6,222,080.0	3,288.0	859,492.0	5,361,866.0	4,010.0
Combined Stat.-Req. Plan. - Stochastic Demand	6,128,480.0	3,079.5	686,809.0	5,440,513.0	4,237.5
Statistical Standard Deviations	6,691,360.0	1,429.0	763,961.0	5,365,396.0	533,431.0
Statistical Service Levels	6,910,720.0	1,263.0	825,641.0	5,407,347.0	678,995.0

Table 6.13

Mean and Standard Deviation of Total Monthly Operating Costs By Inventory Policy Tested

	Mean Monthly Total Cost	Standard Deviation of Total Monthly Costs
Requirements Planning -Deterministic Demand	$ 452,622.	$ 29,795.
Requirements Planning -Stochastic Demand	$ 518,373.	$ 76,804.
Combined Stat.-Req. Plan. -Deterministic Demand	$ 518,699.	$ 99,158.
Combined Stat.-Req. Plan. - Stochastic Demand	$ 522,939.	$117,807.
Statistical (s,S) -Standard Deviations	$ 706,424.	$245,774.
Statistical (s,S) -Service Levels	$ 689,607.	$206,805.

The hypotheses being tested are that there are no differences between the total costs of the policies tested for the same operating cycles, and that the differences which were observed were due to random chance.

The information available for statistical analysis is the sample means, $\bar{X}_1$ and $\bar{X}_2$, the sample standard deviations, s_1 and s_2, and the sample sizes, N_1 and N_2. If the samples were assumed to be drawn from normally distributed populations, the t statistic defined are:

$$t = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}} \tag{6.1}$$

could be used to compare the sample means.

If the assumptions of normality were correct, the statistic would have approximately a t distribution.

The degrees of freedom for the statistic would be:

$$df = \frac{\left[\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}\right]^2}{\frac{\left[\frac{s_1^2}{N_1}\right]^2}{N_1} \quad \frac{\left[\frac{s_2^2}{N_2}\right]^2}{N_2}} \tag{6.2}$$

For these hypotheses, the population means, μ_1 and μ_2, are assumed to be equal and therefore $(\mu_1 - \mu_2) = 0$.

For large sample sizes, the t distribution approaches the normal distribution. The degrees of freedom for the comparison of sample means were calculated according to Equation (6.2), and the range of value obtained was from 70 to 117. With such large values for the degrees of freedom, the t distribution can be approximated by the normal distribution. Dixon and Massey (1969) state that a z-statistic can be used to test for the difference between two population means for large sample sizes even when the populations are not normally distributed. The sampling distribution of the statistic:

$$z = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{\sigma_1^2}{N_1} + \frac{\sigma_2^2}{N_2}}} \qquad (6.3)$$

is normal with mean 0 and variance 1. Since the population variances, σ_1^2 and σ_2^2, are not known, the sample variances, s_1^2 and s_2^2, can be substituted into Equation (6.3) with little loss of accuracy. The z-statistic was used to test for significant differences in the total costs of the various policies.

For many of the comparisons, the differences between the means were relatively large and statistically significant differences could be expected. However, the standard deviations of the total monthly costs were also very large for most test conditions, and therefore what was first thought to be substantially different results may not have been statistically significant differences.

For the observed data, the z-statistic for the difference between the application of the requirements planning and the combined policies tested under deterministic demand was 4.94. Therefore, the hypothesis that the distribution of total operating costs for the requirements planning and combined policies under deterministic demand can be rejected at a significance level of 0.9999^{+}. A similar statement regarding the distributions of total operating costs for the requirements planning and combined policies under stochastic demand cannot be made. For this comparison, the z-statistic is 0.25, which has a significance of 0.60.

The comparison between the distributions of total costs for requirements planning and statistical standard deviations under stochastic demand resulted in a z-statistic of 5.66. The similar comparison between requirements planning and statistical service levels under stochastic demand gave a z-statistic of 6.01. In both cases, the hypotheses that the total costs came from the same distribution are rejected at a significance level of 0.9999^{+}. Comparisons between the combined policy and the statistical policies under stochastic demand gave z-statistics of 5.21 and 5.42 for the standard deviation and service level policies respectively. These statistics also have a significance level of 0.9999+. The z-statistic for the comparison between the standard deviation and service level policies was 0.40, which has a significance level of 0.66.

The results of these tests of significance indicate that there was a significant difference between the total costs resulting from the application of requirements planning and combined inventory policies under deterministic demand. There was insufficient evidence to make a similar statement for the same comparison under stochastic demand.

The application of the statistical inventory policies resulted in significantly higher total operating costs than for either the requirements planning or the combined inventory policies. There was insufficient evidence to reject the hypothesis that there was no difference in total operating costs resulting from the application of statistical standard deviation and statistical service level inventory policies.

6.9 Safety Stock Profiles

For the purposes of this discussion, a safety stock profile was considered as the locus of safety stocks at the four inventory levels which resulted in the minimum total operating costs for the inventory policies tested under stochastic demand. The requirements planning and statistical standard deviation policies measured safety stocks in terms of the standard deviation of demand. The statistical service level policy measured safety stocks in terms of service levels. The combined policy did not have a safety stock profile since safety stock was only carried at the finished products level.

The safety stock profiles for all the policies tested were "U" shaped, which means that the safety stocks for finished products and raw materials were relatively larger than the safety stocks of subassemblies and parts. This supports the opinion of Plossl and Wight (1971) that safety stocks should be concentrated at the finished products level for protection against backorders from varying demand and at the raw materials level to give production flexibility.

6.10 Sensitivity Analysis to the Length of the Operating Cycle

The break-even analysis developed in Chapter Five indicated that as the length of the operating cycle shortened, the requirements planning policies would become relatively more expensive to implement than the statistical inventory policies. The simulation game was adjusted to operate with an operating cycle equivalent to one week as opposed to the regular operating cycle of one month to test the validity of the break-even analysis. The requirements planning and combined policies were both tested under deterministic and stochastic demands. The statistical standard deviation policy was tested only under stochastic demand. The results of these tests are shown in Table 6.14.

The requirements planning policies again result in substantially lower costs than either the combined or statistical policies. However, the total costs include labour adjustment costs which, as discussed above, are not a normal consideration of inventory policies and were not included in the break-even analysis. A comparison of the total costs less labour adjustment costs in Table 6.14 indicates exactly what the break-even analysis predicted. The requirements planning policies have much higher payroll costs due to the additional number of set-ups required compared to the other policies. It is interesting to note that the combined policy results in the lowest total cost when the labour adjustment costs are not included. This is again supportive of Plossl and Wight (1970).

Table 6.14

Cost Comparisons for the Weekly Operating Cycle

	Labour Adjustment Costs	Payroll Costs	Inventory Holding Costs	Backorder Costs	Ordering Costs	Total Costs	Total Costs less Labour Adjustment Costs
Requirements Planning - Deterministic Demand	$ 627,400.00	$ 41,662,726.87	$ 1,234,573.05	$ 0.00	$ 93,400.00	$ 43,618,099.92	$ 42,990,699.92
Requirements Planning - Stochastic Demand	$ 24,503,900.00	$ 42,030,117.50	$ 2,527,696.15	$ 163,520.00	$ 91,150.00	$ 69,316,383.65	$ 44,812,483.65
Combined Stat.- Req. Plan. - Deterministic Demand	$ 55,788,800.00	$ 22,961,333.50	$ 3,180,684.79	$ 0.00	$ 58,300.00	$ 81,989,118.29	$ 26,200,318.29
Combined Stat.- Req. Plan. - Stochastic Demand	$ 54,291,600.00	$ 22,914,317.12	$ 3,590,437.56	$ 354,060.00	$ 58,150.00	$ 81,208,564.68	$ 26,916,964.68
Statistical (s,S) - Standard Deviations	$114,701,900.00	$ 24,258,564.87	$ 8,340,948.61	$ 93,100.00	$ 24,970.00	$147,419,483.48	$ 32,717,583.48

6.11 Discussion of the Simulation

The computational time per computer run in the original form of the "Production Control Project-DCIDE" simulation game was approximately one minute per run on a CDC 6400 computer using a FORTRAN extended compiler. The standard output was approximately three pages per period. The detailed reporting normally given in the game was not required for this research project so the output was condensed to a one page summary per run. This not only saved costs for computer printing but also reduced the CPU time required to process each run.

The search for the minimum total operating costs required an extensive number of computer runs. However as previously mentioned, the search technique was sufficiently efficient to find a total cost which was lower than the total costs at any neighbouring grid point in an average of 100 runs. The minimum total costs were found by testing only four percent of the total combinations of safety stock considered feasible.

6.12 Summary of Key Results

The results of this research project have shown that substantial differences in total operating costs result from the type of inventory policy selected for a multistage manufacturing system. A major difference is the labour adjustment costs. The results of this experiment show that the labour adjustment costs can be sufficient to reverse the relative economies of two inventory policies.

The inventory policies were tested under conditions which allowed the inventory policies to dominate. The relative magnitudes of the labour adjustment costs frequently indicated that the inventory policies were not efficient production control procedures. The interaction of production and workforce smoothing decision rules would have reduced the total operating costs under most of the test conditions, but the comparisons between the inventory policies would not have been as distinct.

The effectiveness of requirements planning policies was clearly demonstrated under both deterministic and stochastic demand functions for monthly and weekly operating cycles. The major cost differences between requirements planning and statistical inventory policies are labour adjustment costs and inventory carrying costs. The statistical policies saved substantially on set-up costs but these savings were less than sufficient to compensate for the higher labour adjustment costs.

The combined statistical-requirements planning policy resulted in higher total costs than the requirements planning policy under deterministic demand but about the same total costs under stochastic demand. When labour adjustment costs were excluded, the combined policy had lower total costs than either the requirements planning Lot-for-Lot or statistical policies. This is supportive of Plossl and Wight (1970).

The service level statistical policy did not result in substantially less total operating costs than the standard deviation policy. This is contrary to the results suggested by Brown (1967) and may indicate that further comparisons of these two approaches should be made.

The safety stock profiles were "U" shaped as hypothesized by Plossl and Wight (1970).

The change of the planning cycle from one month to one week resulted in a shift of total costs which confirmed the break-even analysis developed earlier, provided that the labour adjustment costs were excluded.

The results of this research project are further discussed in the next chapter where some conclusions are made. In addition, some comments are offered on areas where further research should be directed to further the understanding of inventory control in multistage manufacturing facilities.

Chapter Seven

Conclusions and Suggestions for Further Research

The focus of this research has been the comparison of operating costs resulting from the implementation of statistical and requirements planning inventory policies. A simulated manufacturing system was used as the test vehicle. This permitted the duplication of exactly the same conditions for the various tests performed. The variable inputs were governed entirely by the inventory control policies which emphasized the differences between the policies tested.

7.1 Inventory Policies as Production Control Procedures

Inventory policies do not consider the costs of production and workforce changes, which as was determined in the research project, can be a major component of the total operating costs. The statistical policies had the largest labour adjustment costs for both the monthly and weekly operating cycles. The intermittent production requirements of the economic lot-sizes was the major reason for these adjustment costs.

The requirements planning policies contrasted the statistical policies by requiring much fewer labour adjustments. These relatively low labour adjustment costs are due to the production requirements for finished products following the demand functions for finished products. The demand functions for finished products changed gradually from period to period and the labour requirements followed.

The combined statistical-requirements planning policy required more labour adjustments than the requirements planning policy but less labour adjustments than the statistical policies. The production of finished products in economic lot-sizes required substantial labour adjustments at all levels but the direct association of component production to finished product production reduced the labour adjustments at the component assembly stages.

The requirements planning Lot-for-Lot policy did not consider production smoothing in the normal context of the term, but by producing to demand for smooth demand functions, some production smoothing was included. Other discrete lot-sizing techniques such as the Least Total Cost or Least Unit Cost methods would have required intermittent production of components. This would have required larger labour adjustment costs than the Lot-for-Lot policy.

The exlusion of production control procedures from inventory policies deletes a set of cost functions which can have a significant influence on the total operating costs. The results of this research have shown that the inventory policy resulting in the lowest total costs when labour adjustment costs are not included need not be the inventory policy which results in the lowest total operating costs when labour adjustment costs are included. In general, the inclusion of production control procedures would have reduced the total operating costs under most test conditions. In addition, the total cost of operations between policy applications would have been substantially reduced.

Dzielinski and Gomory (1965) formulated a linear programming model for analysing single-stage multiproduct systems. Their model includes labour adjustment costs for increasing and decreasing the size of the workforce from period to period. Unfortunately the Dzielinski and Gomory (1965) model is limited to single-stage multiproduct discrete deterministic demand applications. The application of this type of model would result in substantially lower total operating costs than the inventory policies tested since all the relevant costs are explicitly included in the optimization function.

7.2 The Cost of Uncertainty of Demand

The comparison of the total costs resulting from the application of requirements planning inventory policies under deterministic and stochastic demand functions indicated that the total cost of operating under stochastic demand was approximately $4 million more than under deterministic demand. The tests using stochastic demand in this research used perfect forecasts.

The cost of uncertainty of demand from the implementation of the combined statistical-requirements planning policy was much less than for the requirements planning policy. For the monthly operating cycle, the cost of uncertainty of demand for the combined policy was only $250 thousand for the 60 month test period. This is primarily due to the fewer labour adjustments under stochastic demand as compared to deterministic demand.

A comparison of the costs of uncertainty of demand for the weekly operating cycle were somewhat different than for the monthly operating cycle. For the weekly cycle, the cost of uncertainty of demand for the requirements planning policy increased to approximately $25 million. The similar comparison for the combined policy was $-800,000 indicating that the total operating cost was less under stochastic demand than under deterministic demand. The safety stock smoothed the labour adjustment costs by a larger amount than the cost of carrying the safety stock.

In practical applications, the future demands of the products are usually unknown and some type of forecasting is used. The forecasting techniques can vary from relatively simple relationships to complex mathematical routines. Modifications of the forecasting methods used can usually be made to improve the accuracy of the forecasted demands. With increased forecast accuracy, less safety stock is required, and therefore, lower operating costs can be expected.

There are two approaches to increasing the accuracy of the forecast demands. The first is to improve the forecasting technique so that the misestimation errors are reduced. This may require a comprehensive evaluation of the underlying process to determine whether or not other forecasting techniques should be used. Alternatively, efforts can be made to decrease the uncertainty in the demand. For example, obtaining commitments from customers for future orders would reduce the uncertainty.

Any action taken to reduce the uncertainty of demand should be subjected to a cost-benefit analysis. Uncertainty of demand has a varying cost associated with it, and any attempt to reduce that uncertainty will have a related cost. It would be economical to allocate funds to the reduction of uncertainty as long as the marginal benefit was greater than the marginal cost, but this may not be the best strategy for the organization. The investment required to improve the forecast accuracy may be applied to other opportunities where the benefit would be greater than the benefit from improved forecast accuracy.

7.3 Cost Minimizing Safety Stock Profiles

The results of this experiment have shown that when safety stock is carried at several levels, the cost minimizing profile of safety stock concentrates the safety stock at the finished products and raw materials levels. Plossl and Wight (1970) proposed that such a profile would be desirable without an indication of the optimality of such a profile. The argument offered Plossl and Wight (1970) was that large safety stocks of finished products gave protection against demands larger than the mean expected demands and the large safety stocks of raw materials gave production flexibility.

In all the tests conducted in this project where safety stock levels could be set at each inventory level, the least total operating cost combination of safety stock levels had a "U" shaped profile. While the results of this research do not prove the optimality of the "U" shaped safety stock profile, the concept of the desirability of such a configuration is strongly supported.

7.4 Relative Effectiveness of Inventory Policies

Statistical inventory policies are based upon the assumption of unit withdrawals, a constant rate of demand, and a known distribution of random variation in demand. An order quantity is determined which minimizes the total cost of carrying inventory and set-up or ordering charges.

Requirements planning inventory policies are based upon the time-phasing of materials necessary to have the finished product available to meet the demand. Requirements planning does not use any optimization analysis, but is based upon the assumption that for items subject to dependent demands, the set-up cost of producing the items is relatively low compared to the cost of carrying inventory.

Requirements planning and statistical inventory control policies should be considered as supporting methods rather than opposing alternatives. The two approaches were developed for different purposes. The data generated in this research indicated that if labour adjustment costs are ignored, under deterministic demand the combined statistical-requirements planning inventory was approximately $1.635 million less costly than the standard requirements planning policy. This difference is in the direction expected.

The test which implemented requirements planning inventory control policies under stochastic demand had a significantly lower total cost than either application of statistical inventory control policies. These results also confirm what was anticipated. The demands for all subassemblies, parts and raw materials are dependent upon succeeding products and occur in blocks. These conditions are consistent with the concept of requirements planning.

The occurence of the demands for the finished products in blocks reduced the effectiveness of the statistical policies in favour of the requirements planning policies. The performance of the requirements planning policies was also enhanced by the calculation of inventory changes on the basis of month end inventory.

The application of the combined statistical-requirements planning policy did not result in the lowest total operating costs as predicted by Plossl and Wight (1970). However, they probably were not considering the total cost of operations, but rather the total inventory related costs. When the labour adjustments costs are excluded from the total costs, the combined policy did result in the lowest total costs under both deterministic and stochastic demand functions. This result held for both the monthly and the weekly operating cycles.

In general, the requirements planning Lot-for-Lot policies resulted in the lowest total operating costs for both deterministic and stochastic demands. The application of the statistical policies resulted in the largest total operating costs, and the combined statistical-requirements planning policies resulted in total operating costs which were between the other two types of policies. When the labour adjustment costs were removed from the total operating costs, the combined policies had a lower total cost than the requirements planning policy and the statistical policies were the most costly.

The selection of an appropriate inventory policy is therefore dependent upon the objective which is being pursued. Under the test conditions, the requirements planning Lot-for-Lot policy was the most effective in minimizing the total operating costs. The combined statistical-requirements planning policy was the most effective in minimizing inventory related costs.

7.5 The Influence of the Operating Cycle

The break-even analysis developed in Chapter Five indicated that the length of the operating cycle would influence the type of inventory policy which resulted in the lowest inventory related costs. The subsequent comparison of the simulated system using monthly and weekly operating cycles confirmed the break-even analysis.

For the monthly operating cycle tests, the requirements planning Lot-for-Lot policy resulted in lower total operating costs, less labour adjustment costs than the statistical policies. However, when the operating cycle was reduced to one week, the statistical policy had a lower total cost, less labour adjustment costs than the requirements planning policies.

The reason for the sensitivity of these inventory policies is the trade-off between set-up costs and inventory carrying costs. Other inventory policies may also be sensitive to the length of the operating cycle. Therefore the length of the operating cycle can be an important element in the selection of an appropriate inventory policy.

7.6 Inventory Control in the Total System

Inventory control as a function must be considered in terms of the total system being analysed. Depending on the boundaries of the system under consideration, the inventory related costs may form a part of the total cost structure or they may represent the total costs of the system. If the inventory related costs are the only costs being considered, then the minimization of those inventory related costs is a reasonable objective.

In many instances, inventory related costs are only a part of the total cost structure. Under these conditions, the minimization of inventory related costs may not be the optimal strategy for the system. The results of this research have shown that when the scope of the system is larger than the inventory system, the minimum total operating costs do not minimize inventory related costs. Therefore the total relevant system must be kept in focus so that the optimization of one segment of the system does not lead to sub-optimization of the total system.

7.7 The Influence of the Shape of the System

The shape of the system is defined in terms of the flow of components from stage to stage. In a converging system, the number of inventory items decreases by operating stage from several at the raw materials stage to few at the finished products stage. A diverging system has several finished products and few raw materials. There are many possibilities between these two extreme conditions, but the extremes may be useful for determining the applicability of inventory policies.

Diverging systems have substantial commonality of components in finished products. At some degree of commonality, the effective demand for components at the assembly stages may approach an "independent" type of demand. This would indicate that in the limit, a statistical inventory policy may be the best type of inventory policy for a diverging system.

Converging systems generally have much less commonality of components than diverging systems so that the demand for components is likely to typify dependent demand conditions. Therefore, requirements planning techniques are likely to be the most efficient for converging systems.

Systems which are a combination of diverging and converging systems, such as the "Production Control Project-DCIDE" are difficult to analyse, but the results of this research indicate that requirements planning policies are likely to be the most effective for all system shapes except for the extreme diverging systems.

7.8 The Implementation of Inventory Policies

The ability of a company to successfully implement various inventory policies is dependent upon many factors. One of the most important factors is the number of inventory items in the system. In general, the larger the number of inventory items in a system, the simpler the implementation of the inventory control procedures must be. Small companies which have thousands of inventory items must rely on simple heuristic procedures whereas larger companies can use computers to assist in the implementation of the more sophisticated methods of inventory control.

The basic concept of using a product explosion matrix to time-phase the assembly of components has been in existence for many years but the implementation of the technique for large scale applications was delayed until high-speed computers were available.

Even though the concept of the time-phasing of component assemblies in requirements planning systems is readily comprehendable, the implementation of a requirements planning inventory control system requires the use of a computer for all but the most basic production operations. In contrast, the implementation of an (s,S) system requires less computations, although for operations having thousands of products, computer assistance is essential.

The implementation of requirements planning systems requires that the production operation be considered as composed of several stages. The components of the system flow from stage to stage in discrete blocks. This is in contrast to the assumption of the EOQ model which specifies a continuous demand pattern.

The independent/dependent demand criterion introduced by Orlicky (1970) is a reasonable criterion for determining the applicability of continuous or discrete lot-sizing methods. The use of continuous lot-sizing methods at all stages is possible in a requirements planning system, but by doing so, the time-phasing of component assemblies is overlooked. In most cases, requirements planning systems convert the continous demands for finished products into discrete demands for components at or near the final assembly stage.

Requirements planning is a useful concept for production systems which have several distinct stages and where the demand for components is generated from the demands for finished products. Statistical (s,S) inventory policies are most applicable to inventory items which have continuous demand patterns. The combination of statistical policies for finished products and discrete lot-sizes for other components may be an economical approach depending upon the other costs included in the analysis.

7.9 Suggestions for Further Research

As a result of this research project, a number of other aspects of inventory control theory which require investigation have become apparent. As outlined in the previous section, the shape of the system is expected to be a key variable in the selection of an inventory policy for a multistage system. Diverging systems are likely to respond favourably to statistical systems whereas converging systems are expected to be controlled more effectively by requirements planning systems.

Another area which requires investigation is the effect of demand functions on the choice of an inventory policy. The trend coefficients can be either positive or negative and can vary in slope from shallow to steep. Similarily the seasonality of the demand functions can vary from very minor fluctuations to large seasonal amplitudes. The random element in the demand functions can also take on a wide range of values. The influence of these variations in demand functions is not known and may represent a basic variable for the selection of an appropriate inventory policy.

A multistage multiproduct model similar to the Dzielinski and Gomory (1965) model would be a significant contribution to both the theory and practice of inventory control. The explicit inclusion of labour adjustment costs in the objective function would smooth these adjustments while determining the optimum lot-sizes and production schedules.

The stochastic demands for finished products were generated from demand patterns which were normally distributed. The effect of other demand patterns may be quite different from the effect of normally distributed patterns. A beta distribution would be particularly useful for testing the effect of the demand patterns since the flexibility of the beta distribution facilitates the approximation of most other continuous distributions.

The operating system used for this research project had fixed lead times for all stages of the system. This may have been a very severe restriction which has influenced the conclusions of this research project. An investigation of the effect of variable lead times on the selection of inventory policies may indicate that these conditions are sufficiently different to support inventory policies different from those selected for constant lead time systems.

Further research is also warranted for additional investigation of the system studied in this project. The costs of the various functions should be varied to determine the sensitivity of the total costs to these specific costs and to the selection of the cost minimizing inventory policies. The hiring and firing costs can be adjusted to investigate various corporate policies of underutilizing and overutilizing production personnel as well as test the effect of labour turnover policies. The costs of carrying inventories and labour costs can also be adjusted to determine their influence on the selection of cost minimizing inventory policies.

There are several types of inventory policies which have not been tested in this research project which warrant inventigation. Several methods of discrete lot-sizing methods were discussed but were not tested.

A similar situation occurred with respect to statistical inventory policies. To date, these policies have only been discussed in terms of single-stage operating systems. It may be that they are compatible with multistage systems.

The investigation of the influence of the above variables would be virtually impossible without the use of simulation as a research tool. Simulation has several disadvantages: it can be very expensive as evidenced by this research project in which approximately $5,000 of computer funds were required. The more detailed the simulation model becomes, in general the more expensive it is to run. However, simulation has the advantage that it reduces industrial complexity to a manageable level at which experiments can be undertaken to determine the influence of selected variables. The other major advantage is that the results of the experiment are available in a very short period of time.

The alternative to simulation as a research tool is to study industrial applications. The study could be passive in which the researcher would attempt to identify comparable companies using different inventory policies and make comparisons of their results. The other approach would be to attempt an experimental design to test the influence of differing inventory policies on participating companies. Neither of these approaches are likely to be very successful.

The passive study would have considerable difficulty in finding comparable companies to conduct the required tests. The active role would have a similar difficulty gaining the cooperation of companies to participate in such an experimental project.

Therefore, although simulation has many drawbacks as a research tool, there are a number of situations in which it is the best technique available. Research into the influence of inventory control policies is one of those areas where simulation can be used to great advantage.

Bibliography

Arrow, K., Karlin, S. and Scarf, H. (eds.). Studies in the Mathematical Theory of Inventory and Production. Stanford, California: Stanford University Press, 1959.

Balintfy, J. L. "On a Basic Class of Multi-Item Inventory Problems". Management Science, Vol. 10, No. 2, January 1964. pp. 289-297.

Berry, William L. "Lot Sizing for Requirements Planning Systems: A Framework for Analysis". Paper No. 322. Herman C. Krannert Graduate School of Industrial Administration, Purdue University, Lafayette, Indiana. August 1971.

Britney, R. R. Production Control Project - DCIDE, The University of Western Ontario, Revised July 1972.

Brown, R. G. Decision Rules for Inventory Management. New York: Holt, Rinehart and Winston, 1967.

Buchan, J. and Koenigsberg, E. Scientific Inventory Management. Englewood Cliffs, New Jersey: Prentice-Hall, 1963.

Buffa, E. S. Production-Inventory Systems: Planning and Control. Homewood Illinois: Richard D. Irwin, 1968.

Crowston, W. B. and Wagner, M. H. "Dynamic Lot Size Models for Multi-Stage Assembly Systems". A paper presented at TIMS XIX International Meeting. Houston, Texas, 1972.

Crowston, W. B., Wagner, M., and Williams, J. F. "Economic Lot Size Determination in Multistage Assembly Systems". Working paper 566-71. Alfred P. Sloan School of Management, Massachusetts Institute of Technology: October, 1971.

Dixon, W. J. and Massey, F. J. Jr. Introduction to Statistical Analysis. McGraw-Hill, Inc. 3rd Edition, 1969.

Doll, R. E. "Requirements Planning Simulation 'The New Frontier of Inventory Management'". APICS Conference Proceedings, 1971.

Dzielinski, B. and Gomory, R. "Optimal Programming of Lot Sizes, Inventories and Labour Allocations". Management Science, Vol. 11, No. 9, July 1965. pp. 874-890.

Eilon, S. Elements of Production Planning and Control. London: Macmillan, 1962.

Eilon, S. and Elmaleh, J. "An Evaluation of Alternative Inventory Control Policies". The International Journal of Production Research, Vol. 7, No. 1, 1968.

Eilon, S. and Elmaleh, J. "Adaptive Limits in Inventory Control". Management Science, Vol. 16, No. 8, April 1970.

Factor - Integrated Management Information and Control System for Manufacturing. Honeywell, 1969.

Geisler, M. A. "The Sizes of Simulation Samples Required to Compute Certain Inventory Characteristics with Stated Precision and Confidence". Management Science, Vol. 10, No. 2, January 1964. pp. 261-286.

Geisler, M. A. "A Test of a Statistical Model for Computing Selected Inventory Model Characteristics by Simulation". Management Science, Vol. 10, No. 4, July 1964. pp. 709-715.

Gorenstein, S. "Some Remarks on EOQ vs. Wagner-Whitin". Production and Inventory Management, Second Quarter, 1970.

Gorham, T. "Dynamic Order Quantities". Production and Inventory Management, First Quarter, 1968.

Haeling von Lanzenauer, C. "Production and Employment Scheduling in Multistage Production Systems", Naval Research Logistics Quarterly, Vol. 17, No. 2, June 1970. pp. 193-198.

Haeling von Lanzenauer, C. "A Production Scheduling Model by Bivalent Linear Programming". Management Science, Vol. 17, No. 1, September 1970. pp. 105-111.

Herron, D. P. "Inventory Management for Minimum Cost". Management Science, Vol. 14, No. 4, December 1967. pp. B-219- B-235.

Herron, D. P. "Service Levels Versus Stockout Penalties - A Suggested Synthesis". Production and Inventory Management, Third Quarter, 1969. pp. 43-53.

Holt, C. C., Modigliani, F., Muth, J. F. and Simon, H.A. Planning Production, Inventories and Work Force. Englewood Cliffs, New Jersey: Prentice-Hall, 1960.

Hooke, R. and Jeeves, T. A. "Direct Search Solution of Numerical and Statistical Problems". The Journal of the Association of Computing Machinery, April 1961. pp. 212-229.

Iglehart, D. L. "Optimality of (S,s) Policies in the Infinite Horizon Dynamic Inventory Problem". Management Science, Vol. 9, 1963. pp. 259-267.

Jones, C. H. "Parametric Production Planning". Management Science. Vol. 15, No. 11, July, 1967.

Kaimann, R. A. "A Fallacy of 'E.O.Q.ing'". Production and Inventory Management, First Quarter, 1968.

Kaimann, R. A. "E.O.Q. vs. Dynamic Programming-Which One to Use for Inventory Ordering?". Production and Inventory Management, Fourth Quarter, 1969.

Karlin, S. and Fabens, A. J. "A Stationary Inventory Model with Markovian Demand". Chapter 11 in Arrow, K. S., Karlin, S. and Suppes, P. (eds.). Mathematical Models in the Social Sciences. Stanford, California: Stanford University Press, 1960.

Kochenberger, G. A. "Inventory Models: Optimization by Geometric Programming". Decision Sciences. Vol. 2, April 1971. pp. 193-205.

Love, S. F. "A Facilities in Series Inventory Model with Nested Schedules". Management Science, Vol. 18, No. 5, January 1972. pp. 327-338.

Magee, J. F. and Boodman, D. M. Production Planning and Inventory Control. New York: M-Graw Hill, 1967.

Manne, A. S. "Programming of Economic Lot Sizes". Management Science, Vol. 4, No. 2, January 1958. pp. 115-135.

Orlicky, J. P. "Requirements Planning Systems". APICS Conference Proceedings, 1970. pp. 228-239.

Peterson, R. Thomas, L. J. and Loiseau, A. J. "Operational Inventory Control with Stochastic Seasonal Demand". Paper presented at the 39th National ORSA Meetings, Dallas, Texas. May 1971.

Plossl, G. and Wight, O. "Designing and Implementing a Material Requirements Planning System". _APICS Conference Proceedings_, 1970. pp. 206-227.

Pritchard, J. W. and Eagle, R. H. _Modern Inventory Management_. New York: McGraw Hill, 1956.

The Production Information and Control System, IBM Data Processing Manual, 1968.

Scarf, H. "The Optimality of (s,S) Policies in the Dynamic Inventory Problem" in Arrow, K. J., Karlin, S. and Suppes, P. (eds.). _Mathematical Models in the Social Sciences_. Stanford California: Stanford University Press, 1960.

Scarf, H. "A Survey of Analytical Techniques in Inventory Theory". Chapter 7 in Scarf, H. Gilford, D. and Shelley, M. (eds.). _Multistage Inventory Models and Techniques_. Stanford, California: Stanford University Press, 1963.

Schussel, G. "Job-Shop Lot Release Sizes". _Management Science_, Vol. 14, No. 8, 1968. pp. B449-B472.

Silver, E. A. "Some Characteristics of a Special Joint-Order Inventory Model". _Operations Research_, Vol. 13, No. 2, March-April 1965. pp. 319-322.

Silver, E. A. and Meal, H. C. "A Simple Modification of the EOQ for the Case of Varying Demand Rate". _Production and Inventory Management_, Fourth Quarter, 1968.

Starr, M. K. Production Management: Systems and Synthesis. Englewood Cliffs, New Jersey: Prentice-Hall, 1963.

Starr, M. and Miller, D. Inventory Control: Theory and Practice. Englewood Cliffs, New Jersey: Prentice Hall, 1962.

Taha, A. H. and Skeith, R. W. "The Economic Lot Size in Multistage Production Systems". AIEE Transactions, June 1970.

Thomas, A. B. "Optimizing a Multi-Stage Production Process". Operational Research Quarterly, Vol. 14, No. 2, June 1963. pp. 201-213.

Thurston, P. H. "Requirements Planning for Inventory Control". Harvard Business Review, Vol. 50, No. 3, May-June 1972. pp. 69-71.

Veinott, A. F. Jr. "The Status of Mathematical Inventory Theory". Management Science, Vol. 12, No. 11, June 1966. pp. 745-777.

Veinott, A. F., Jr. "Minimum Concave-Cost Solution of Leontief Substitution Models of Multi-Facility Inventory Systems". Operations Research, Vol. 17, No. 2, March-April 1969. pp. 262-291.

Veinott, A. F., Jr. and Wagner, H. M. "Computing Optimal (s,S) Inventory Policies". Management Science, Vol. 11, No. 7, May 1965. pp. 525-552.

Verma, H. L. "Some Joint Inventory Models". A paper presented at the 12th Annual Meeting of the Institute of Management Science, Detroit, Michigan, 1971.

Wagner, H. M., O'Hagan, M. and Lundh, H. "An Emperical Study of Exactly and Approximately Optimal Inventory Policies". Management Science, Vol. 11, No. 7, May 1965. pp. 690-723.

Wagner, H. M. and Whitin, T. M. "Dynamic Version of the Economic Lot Size Model". Management Science, Vol. 5, No. 1, October 1958.

Wight, O. "To Order Point or Not to Order Point". Production and Inventory Management, Third Quarter, 1968.

Wilde, D. J. Optimum Seeking Methods, Englewood Cliffs, New Jersey: Prentice-Hall, Inc. 1964.

Zangwill, W. I. "A Deterministic Multi-Period Production Scheduling Model with Backlogging". Management Science, Vol. 13, 1966. pp. 105-119.

Zangwill, W. I. "Production Smoothing of Economic Lot Sizes with Non-Decreasing Requirements". Management Science, Vol. 13, No. 3, November 1966. pp. 191-209.

Zangwill, W. I. "A Deterministic Multiproduct Multi-facility Production and Inventory Model". Operations Research, Vol. 14, 1966. pp. 486-507.

Zangwill, W. I. "Minimum Concave Cost Flows in Certain Networks". Management Science, Vol. 14, 1968. pp. 429-450.

Zangwill, W. I. "A Backlogging Model and Multi-Echelon Model of a Dynamic Economic Lot Size Production System-A Network Approach". Management Science, Vol. 15, No. 9, May 1969. pp. 506-527.

APPENDIX I

The Wilson EOQ Model

The basic formula for the Wilson model is derived as follows:

Assume:

1. Demand is constant, at a rate of A units per year.
2. The fixed cost of replenishing an order (administrative, shipping or set-up costs) is a constant S dollars. The unit cost of an item is a constant C.
3. The annual cost of carrying inventory in stock is i, expressed as a percentage of the cost of the item, C.
4. An order is placed for the items to be available when the inventory on hand is depleted.

If a quantity Q is obtained on each order, $\frac{A}{Q}$ orders per year must be placed. The average stock on hand will be equal to one-half the order quantity, $\frac{Q}{2}$. The total annual cost of placing orders, purchasing or manufacturing materials and carrying inventory is given by:

$$(I.1) \quad TC = \frac{AS}{Q} + \frac{QCi}{2} + AC.$$

The objective is to minimize the total of these inventory related costs. Differentiating Equation (I.1) with respect to Q gives:

$$(I.2) \quad \frac{d(TC)}{dQ} = \frac{-AS}{Q^2} + \frac{Ci}{2}$$

Setting Equation (I.2) equal to zero and solving for Q gives:

$$(I.3) \quad Q = \sqrt{\frac{2AS}{Ci}}.$$

To verify that the order quantity Q does minimize the total cost equation, the second derivative of Equation (I.1) with respect to Q yields:

$$(I.4) \quad \frac{d^2(TC)}{dQ^2} = \frac{2AS}{Q^3}$$

Substituting the value of Q from Equation (I.3) into Equation (I.4) gives a positive value indicating that the order quantity Q does minimize the total cost equation.

APPENDIX II

The Buchan and Koenigsberg Single Stage Multiproduct Continuous Deterministic Demand Model

The first case discussed by Buchan and Koenigsberg (1963) is when a limitation is set by a capital restriction. They consider the problem of an inventory of J items subject to a capital restriction such that the total value of inventory should not exceed X dollars. They state that if the capital restriction were not in effect, the total inventory value would be:

$$\text{(II.1)} \quad I_T = k \sum_{j=1}^{J} Q_j C_j \ .$$

where the j subscript indicates item j, and k is a "normalizing" factor, C_j is the cost per unit of item j as previously defined in Appendix I without the subscript. Buchan and Koenigsberg (1963) describe the normalizing factor k as follows:

> This normalizing factor is inserted to account for the fact that stocks of individual items can and do arrive independently of one another. The factor k lies between zero and one. If all items are replenished at the same time, then the maximum investment occurs at that time; that is k = 1. If we set k = ½, we assume that replenishments are spread over time so that the inventory investment, on the average, is half the maximum investment.

They suggest that to be on the conservative side, k should be assumed to be between ½ and 1. As previously stated, if I_T is less than X, the lot sizes determined by Equation (I.3) can be used. However, if I_T is greater than X when the Q_j values as determined by Equation (I.3) are inserted in Equation (II.1), the lot sizes must be adjusted to meet the capital limitation. The prevailing condition at that point is

$$(II.2)\quad X \geq I_T = k \sum_{j=1}^{J} Q_j C_j \ .$$

or

$$(II.2a)\quad X - k \sum_{j=1}^{J} Q_j C_j \geq 0.$$

Buchan and Koenigsberg (1963) define a quantity z such that:

$$z < 0 \text{ when } X - k \sum_{j=1}^{J} Q_j C_j = 0.$$

or

$$(II.3)\quad z = 0 \text{ when } X - k \sum_{j=1}^{J} Q_j C_j > 0.$$

They then write the total variable cost equation as:

$$(II.4)\quad TC_T = \sum_{j=1}^{J} \left(\frac{S_j A_j}{Q_j} + \frac{Q_j C_j i}{2} \right) + z \left(X - k \sum_{j=1}^{J} Q_j C_j \right) \ .$$

This equation is of the same format as Equation (I.1) rewritten as a summation over J products. The z which was introduced in Equation (I.3) is a Lagrangian multiplier which has the effect of driving the value of the last term of Equation (II.4) to zero by forcing a solution such that the total inventory investment is less than X dollars.

To minimize the total cost, Equation (II.4) is differentiated with respect to Q_j and the derivative is set equal to zero. The solution is given by:

$$(II.5)\quad Q_j = \sqrt{\frac{2\ A_j S_j}{C_j (i-2kz)}} \ .$$

This is the result obtained in Equation (I.3) except that i has been replaced by (i-2kz). However from Equation (II.3), it is noted that z must be negative or zero. The effect of Lagrangian multiplier z is to increase the cost of carrying inventory until the capital restriction is satisfied.

The second case discussed by Buchan and Koenigsberg (1963) is the limitation set by production scheduling and capital resources. To guarantee a feasible schedule, they impose a restriction making the number of runs the same for all products. A run of the entire product line is called a cycle. The first step is to find the number of cycles per year which minimizes the total cost. The total variable cost for all products can be written as:

$$\text{(II.6)} \quad TC_T^p = \sum_{j=1}^{J} \left[S_j N + \frac{A_j}{2N} C_j \; i \left(1 - \frac{A_{dj}}{p_{dj}} \right) \right] .$$

where N is the number of cycles per year, A_{dj} is the daily demand rate of the j^{th} item and p_{dj} is the daily production rates of the j^{th} item. In this model the assumption of instantaneous delivery has been replaced by the assumption that delivery of the item is spread over a period of time equal to $\frac{Q_j}{p_{dj}}$. It should also be noted that the maximum inventory on hand for a product j is $Q_j \left(1 - \frac{S_{dj}}{p_{dj}} \right)$.

The number of cycles per year which minimize Equation (II.6) is determined by differentiating with respect to

N, setting the derivative equal to zero and solving for N. The result is:

$$(II.7)\quad N = \sqrt{\frac{\sum_{j=1}^{J} A_j C_j \cdot i \left(1 - \frac{A_{dj}}{P_{dj}}\right)}{2 \sum_{j=1}^{J} S_j}} \; .$$

Since $Q_j = \frac{A_j}{N}$, then:

$$(II.8)\quad Q_j = \sqrt{\frac{2A_j^2 \sum_{j=1}^{J} S_j}{\sum_{j=1}^{J} A_j C_j \cdot i \left(1 - \frac{A_{dj}}{P_{dj}}\right)}} \; .$$

To include the capital restriction with the production scheduling restriction, the Lagrange multiplier method is used. The total variable cost equation under these conditions is:

$$(II.9)\quad TC_T^1 = \sum_{j=1}^{J} \left[S_j N + \frac{A_j}{2N^2} \cdot i \left(1 - \frac{A_{dj}}{P_{dj}}\right) \right] + z \left[X - k \sum_{j=1}^{J} \frac{A_j C_j}{N} \left(1 - \frac{A_{dj}}{P_{dj}}\right) \right] .$$

Differentiating with respect to N and setting the derivative equal to zero gives:

$$(II.10)\quad N = \sqrt{\frac{\sum_{j=1}^{J} A_j C_j \left(1 - \frac{A_{dj}}{P_{dj}}\right) (i - 2\,k\,z)}{2 \sum_{j=1}^{J} S_j}} \; .$$

and

$$(\text{II}.11) \quad Q_j = \frac{A_j}{N} = \sqrt{\frac{2\,A_j^2 \sum_{j=1}^{J} S_j}{\sum_{j=1}^{J} A_j C_j \left(1 - \frac{A_{dj}}{p_{dj}}\right) \cdot (i - 2\,k\,z)}} \,.$$

As before, the effect of z is to increase the cost of carrying inventory so that the lot sizes are reduced sufficiently to accommodate the capital restriction.

APPENDIX III

THE PRODUCTION CONTROL PROJECT - DCIDE

Manufacturing Operations

Business 444/644

INSTRUCTION MANUAL
PRODUCTION CONTROL PROJECT -- DCIDE
(IBM 1130 Version)

by

Robert R. Britney

School of Business Administration
The University of Western Ontario

Revised July, 1972

This simulation is an adaptation of a game originally developed by Dr. Peter F. Winters. Mr. Christopher J. Piper contributed substantially to the development of this IBM 1130 version.

TABLE OF CONTENTS

LIST OF FIGURES, TABLES AND COMPUTER PRINTOUTS

I. Introduction

The major project for Manufacturing Operations (Business 444-644) is an investigation of planning and scheduling for a simple, simulated factory. The project requires decision procedures to be programmed and run on the IBM 1130 by groups of students. The factory (Figure 1) consists of 3 departments: Final Assembly, Subassembly, and Parts. There are four types of inventory: finished goods, subassemblies, parts, and raw materials. Raw materials are ordered from suppliers, and sales of finished products are made to customers, so that the factory is delimited by dashed lines. All details of the factory simulation program are given in this report.

The simulation is made on a monthly basis. (See the Master Flow Chart.) A forecast of sales is made for each of the next 6 months for each of the 15 final products, and for 15 months for the aggregate (man-hour) demand on the factory. The Linear Decision Rules of Holt, Modigliani, Muth and Simon are used to calculate suggested production (P* , in man-hours) and workforce (W* , in number of workers) for the coming month. The program then calls on the program DCIDE (written by the students) to obtain 48 decisions:

a) how much to produce (if any) of each of the 15 final products, in the Final Assembly department;

b) how much to produce of each of the 7 subassemblies and 15 parts, in those departments;

c) how much of each of 8 raw materials to order;

d) how many workers to have in each of 3 departments.

DCIDE has available to it all relevant information, such as P* , W* , current inventory levels, forecasts, demand generation parameters, costs, and performance for year-to-date. It is not necessary for DCIDE to read any data cards or print out any results, but it may.

When DCIDE has made the 48 required decisions, control is returned to The Program PROSM which simulates the operation of the factory. For each department, input material requirements and workforce requirements are checked, and desired production is reduced if availabilities are exceeded. Then, all inventories which are changed by the department's production (input and output) are updated. The simulation operates the Final Assembly department first, so that input material to this department must have been made available by other departments at least by the end of the previous month. Then the Subassembly and Parts departments are operated, in that order, and with the same restrictions as above. Next, the month's demand is filled from inventory; excess demand is backordered and is added to demand in the following month. Finally, raw material orders placed at the beginning of the month (through DCIDE) are received and updated.

PROSM prints out a variety of information about demand and backorders, inventory levels and values, time spent on set-up and running, and idle time, straight time, and overtime. Decisions and actual production (which may be different from desired production) are printed by the main program. Costs are calculated for hiring, firing, and transfer of workers, payroll costs, inventory holding and backorder costs. All of the above is done during a simulated month. Then the program increases the month number and returns to the start for new forecasts, new decisions and so forth. Summary information is kept during the year, and is printed in a yearly report. Student investigators have control over the number of months to be run.

A statistical reporting option is available for those wishing to chart their performance relative to others playing the game. Details are given in Appendix A.

This project is to be worked on by groups of students, with the size and makeup of each group indicated by the course syllabus. The group should decide on policies and strategies for making decisions, and should write and punch DCIDE to be run on the 1130. Essentially unlimited experimentation is possible. The objective of this experimentation is to find decision processes which reduce (or "minimize") the costs listed above. At the end of the course the group should turn in:

a) a written report on the group's investigation which must include at least
   - i) a description of their experimentation and results;
   - ii) for DCIDE, a flow chart, program listing, and variable dictionary;
   - iii) the computer output for at least one sample run.

b) a complete DCIDE deck ready to run with all necessary data cards and instructions.

An oral team report will also be required with details to be discussed later.

During the course of the term, a consultant will be available for programming assistance as required.

II. Product Explosion

Figure 2 is a matrix (XM1(i,j)) that shows the materials (raw materials, parts, subassemblies) required to make other materials (parts and subassemblies) and the finished product. For simplicity, all entries are 0 or 1. A 1 indicates that 1 unit of the material above is required to make 1 unit of the product at the left. For example, to make 1 unit of product i=5 (a finished product) requires 1 unit of j=16 (subassembly) and 1 unit of j=27 (a part). To make 1 unit of product 16 (subassembly) requires 1 unit each of 28 (parts) and 38 and 44 (raw materials).

XM1 would require 4,050 words of core if used directly as a matrix. To eliminate this prohibitive requirement, product explosion data are stored in compressed form in matrix M1. A function, called XM1, then computes the "value" of XM1 for any XM1(I,J) by interpreting M1. The user may thus treat XM1 as if it were a matrix as long as he remembers two things: (1) M1 must be in COMMON and (2) XM1 must not be dimensioned nor placed in COMMON.

In the event of a material shortage, desired production is reduced proportionately in an attempt to just use up the short material. For example, if DCIDE tries to produce 100 units of 6 and 100 units of 7, when only 150 units of 17 are available, then a reduction to 75 units of each of 6 and 7 will be made.

In order to operate all the departments with the same program loop, products made (output) by department d are indexed by BMR(d) and BN(d), where BMR(d) is the product number immediately preceding the first one for department d, and BN(d) is the number of products produced by that department. Thus a DO loop starting with BMR(d)+1, and going in steps of 1, to BMR(d)+BN(d) will index the products made by department d.

FGP(i) (finished goods productivity for product i) gives the total manhours in all departments needed to produce 1 unit of the finished product. These 15 numbers are re-calculated every month and include pro-rated set-ups for each department based upon the lot sizes contained in user array Q. Initially this array is set to the values shown in figure 6. It is possible to determine approximately the total factory man-hours required to produce a given set of finished product lots, considering the production not only of the finished products but also all of the materials that go into the finished product. This total can be compared with PSTAR, stated in total factory man-hour terms, except of course that PSTAR is for a particular month, whereas the production of subassemblies and parts may occur in different months from the finished product production.

Raw materials are ordered in the amount specified by DCIDE, at the beginning of the month. Raw materials 38, 39, 40, 41 have one month lead times and will arrive at the end of the month in which they are ordered. Raw materials 42, 43, 44, 45 have 2-month lead times, arriving at the end of the following month. All materials arriving at the end of any month will appear on the on-hand inventory record at the end of that month, and will be available for use at the beginning of the following month. On-order quantities of 42, 43, 44, 45 are stored in, and printed as, 00(42), 00(43), 00(44), 00(45). Each raw material has a purchase cost in $/unit. These are:

RM No.	RC, $/unit	RM No.	RC,$/unit
38	10	42	2
39	5	43	5
40	50	44	20
41	10	45	80

III. Decisions Required

Production and workforce are to be inserted into the vector DEC(i), i=1, ..., 48, as follows:

(a) Units of production:

DEC(1) = production of product 1 this month (units)

.

.

DEC(37) = production of product 37 this month (units)

(b) Units of raw materials:

DEC(38) = order for raw material 38 this month (units)

.

.

DEC(45) = order for raw material 45 this month (units)

(c) Number of workers:

DEC(46) = workforce for Final Assembly this month, number of workers

DEC(47) = workforce for Subassembly this month, number of workers

DEC(48) = workforce for Parts this month, number of workers

IV. Demand Generation

Sales demand of final products is generated by the main program according to the following formulas using base, trend, sine function and noise components as shown below where * denotes multiplication:

(4.1) $SFG(i) = \left[AS(i) + BS(i) * t + CS(i) * SIN\left(\frac{\pi t}{6}\right)\right] * \left[1 + E(i)\right]$ for i=1, ..., 12

(4.2) $SFG(i) = e^{E(i)}$, for i = 13, 14, 15

where t is the month number, SFG (i) is new demand of product i in units.

For products 1-12 a normal random number is used to alter the base series, which is a sine function imposed on a linear trend. That is E(i) is a normal random number with mean 0 and standard deviation given below with the other relevant parameters.

Product (i)	AS(i)	BS(i)	CS(i)	σ of E(i)
1	1000	-8	-400	.1
2	500	+10	-200	.25
3	200	+5	-100	.25
4	100	0	-20	.1
5	50	+1	0	.15
6	300	-2	-100	.15
7	500	0	-200	.15
8	1000	0	-350	.2
9	100	+3	0	.25
10	500	-2	-150	.1
11	200	+2	+75	.2
12	100	-1	+50	.15

For products 13, 14 and 15, the mean and standard deviation of E(13), E(14), E(15), have been chosen so that, after the exponential transformation, SFG(13), SFG(14), SFG(15) will have the following mean and standard deviation:

Product (i)	Mean Sales	Standard Deviation
13	500	75
14	300	50
15	800	200

Each month, before calling DCIDE, forecasts for all final products are calculated from the base series (without random element) for the next 6 months, including the current month. These forecasts are in terms of units, and are stored in FOR(i,j), where i=1, ..., 15 refers to the product number, and j=1, ..., 6 refers to the month ahead. Thus FOR(2,3) is a forecast for product 2 for 3 months ahead; FOR(11,1) is a forecast for product 11 for the current month. In addition, the 15 months' forecasts used in the linear decision rules are made month by month, aggregated in man-hour terms using FGP(i) for each product. These forecasts are available in SHAT(1)...SHAT(15) where SHAT(1) refers to the current month.

Sales generation constants are provided through COMMON so that DCIDE can compute more forecasts easily. Also, the linear decision rule coefficients are provided so that plans for more than the current month (available as P* and W*) can be determined by DCIDE.

V. Cost Calculations

The costs calculated by the main program include: (a) hiring, firing and transfer of workers; (b) payroll costs consisting of straight time and overtime; (c) inventory holding costs; and (d) backorder costs.

(a) Hiring, Firing and Transfer Costs

The total workforce is the sum of the number of workers in each of the 3 departments. It costs $800/man if this total is increased (over the previous month) and $500/man if it is decreased. These calculations are made independent of how the total workforce is assigned to departments. In addition, a transfer cost of $100/worker is made. The following examples will clarify our meaning about these calculations, where A is the final assembly, S the subassembly, and P the parts department.

Month	Number of Workers In			Total	Cost of H, F, T
	A	S	P		
1	50	200	100	350	$
2	52	198	100	350	200 (T)
3	52	200	100	352	1,600 (H)
4	50	202	101	353	1,000 (H, T)
5	51	200	100	351	1,100 (F, T)

(b) Payroll Costs

Straight time is charged for all workers on the payroll for 160 hours/month, regardless of whether the full quantity of straight time is used. The hourly wage rate is $3.50.

In addition, overtime will be paid as required, at time-and-a-half. There is a limit of 30% overtime in any department, and desired production is automatically reduced if it requires more overtime.

(c) Inventory Holding Costs and Inventory Accounting

A charge of 2% of the inventory value is made each month, on the month-end $ balance. This rate is about equivalent to 24% per year.

The average cost method is used for inventory accounting. This means that whenever units are withdrawn from any inventory, the value balance is reduced by the average value of these units. If the inventory consists of I units and V dollars, and n units are withdrawn, the new balance is (I-n) units and {V-n[V/I]} dollars. When units are added to inventory, that is, when a product is made by a department, the unit balance is increased by the number made, and the dollar balance is increased by the value of input materials and direct costs incurred by the department to produce the material. For example, when 100 units of 20 are made, the value (current) of 100 units of 34 and 40 are added to the dollar balance of 20, along with the cost of straight time and over-time used in the subassembly department to make 20. Idle time and hiring, firing, and transfer costs are considered charged to current overhead expense; overhead is not charged to the inventory account.

(d) Backorder Cost of a Finished Good

If demand of a finished good exceeds its inventory, the excess demand is backordered (to be added to demand in the next period) and a charge is made of $140 per unit backordered. This cost is meant to represent partly lost profit (if in fact the demand were lost and not backordered) and partly customer ill will and loss of future business, charged now.

VI. Linear Decision Rules

The PROSM program caculates PSTAR and WSTAR according to the linear decision rules of Holt, Modigliani, Muth and Simon[4]. These calculations are made using the following formulas:

$$\text{(5.1)} \quad \text{PSTAR} = \sum_{t=1}^{12} \text{SHAT(t)} * \text{ALPHA(t)} + \text{ALPHA(13)} + \text{ALPHA(14)} * \text{WLAST} + \text{ALPHA(15)} * \text{XILST}$$

$$\text{(5.2)} \quad \text{WSTAR} = \sum_{t=1}^{12} \text{SHAT(t)} * \text{BETA(t)} + \text{BETA(13)} + \text{BETA(14)} * \text{WLAST} + \text{BETA (15)} * \text{XILST}$$

The SHAT(t), t=1, ..., 12 are forecasts of aggregate sales, in man-hour terms, for the next 12 months, where t=1 represents the current month. WLAST is the total factory work force during the previous month. XILST is the <u>net</u> finished goods inventory, in man-hours, at the end of the previous month; by <u>net</u>, we mean on-hand inventory - backorders. Inventories and backorders are expressed in units, and are translated to man-hour terms by the FGP(i).

The DCIDE program may use PSTAR and WSTAR, may modify the advice, or may ignore it altogether. All of the ingredients used to calculate these are provided in COMMON, along with 15 months of aggregate forecasts, so that it is possible for DCIDE to generate PSTAR and WSTAR for three additional periods, each month.

The following cost coefficients were used to develop the α and β constants for the rules:

C_1 = 560 \$/man-month

C_2 = 18.9 \$/man^2

C_3 = .0003113 \$/(man-hour)^2

C_4 = 160 man-hour/man-month

C_5 = 1,8680 \$/m-h

C_6 = 298.88 \$/man

C_7 = .000005 \$/(man-hour)^2

C_8 = 106,000 man-hours

C_9 = 0

C_{10} = 0

C_{11} =-4 men

C_{12} = 0

Graphs of the hiring-firing, overtime, and inventory cost functions are given in Figures 3, 4 and 5 respectively. If a team wants to use different cost coefficients and recompute the α's and β's, a FORTRAN program will be available to make these calculations.

VII System Description

"CALL LINK" Used. Production Control Project -- DCIDE has been made available on the IBM 1130 by making use of the "CALL LINK" and disc file capabilities of this machine. These features allow a number of mainline (ML) programs to be "linked" together into a chain. The chain, so formed, is capable of performing tasks that could otherwise only be performed by a machine with a larger core.

"COMMON Carries Information. Each ML shares a COMMON area and this is how information is passed from one ML to another. Note that information not placed in COMMON is not necessarily retained when a LINK is effected. The user may insert up to 1,200 words into the common area for retention from month to month. Information on such things as trigger levels may thus be retained from one month's operation to the next. When placing real variables into common, the user should take care to start each variable on an odd word. Not all 1,200 words need be used, (for example, a typical user common might simply be:

"COMMON TRIGL(45)"

In this case, the 1,110 words not used as common variables can be used instead to extend the length of the DCIDE program.

ML's in DCI Format. All ML's have been stored in disc core image (DCI) in the fixed area (FX) of disc 4444. This means that the programs can be executed immediately since all system subroutines were assembled with the ML's when the ML's were stored.

DCIDE. The DCIDE ML must be prepared by the user. The last statement before END must be CALL LINK (PROSI). (See section VIII.)

Error Checks. Just before calling DCIDE, the main program writes values of all COMMON variables in a "SAFE", and just after DCIDE checks to see that none of the COMMON values (except DEC(1) ... DEC(48) LIMIT, Q, MMAX and STAR) have been altered, inadvertantly or otherwise. If any value is changed, the message "WORD N changed illegally. Execution terminated." results. Word N refers to the position of the variable in COMMON, noting that each real variable has two words.

If any decision (DEC(i)) is negative, the main program will print a warning, along with the decision number and negative value, and then will set the decision to zero. No decision should be negative.

Disc Files Retain Game Information. Each team playing the game will be assigned a file on which the current value of all variables is recorded. These files will be identified by code names that will be provided. A team may thus start at month one on one job, terminate the run after M months, and resume execution at month M+1 on a subsequent job. This feature will permit the user to play the game while observing the queue discipline existing at the time (e.g., 5 minute maximum run time). Data switches 2 and 3 as described below are relevant.

Console Input. Certain information may be entered through the 1130's console keyboard and data switches. The type of information and the interpretation of data switch positions can be obtained from the console printer at the start of each run. A copy of this output is reproduced below.

VII a) CONSOLE OUTPUT

```
              MANUFACTURING OPERATIONS 444/644
            PRODUCTION CONTROL PROJECT --DCIDE

DATA SWITCHES IN THE UP POSITION FUNCTION AS FOLLOWS

0 - SUPPRESSES THESE INSTRUCTIONS
1 - RESUMES EXECUTION OF THE PREVIOUS RUN
2 - SUSPENDS EXECUTION AFTER PROCESSING DECISIONS FOR THE CURRENT
      MONTH
3 - SUPPRESSES MONTHLY PRINT-OUT
4 - ALLOWS NEW RANDOM START TO BE SPECIFIED BY USER
5 - READS USER-PROVIDED DATA CARDS TO START YEAR 1

    NOTE THAT CONSOLE INPUT MUST BE RIGHT JUSTIFIED UNDER THE
      DASHES

          PRESS START
```

- 11 -

```
ENTER NEW RANDOM NUMBER SEED
----
0132
ENTER MAXIMUM MONTHS FOR THIS RUN
---
012
```

VIII. Deck Make-Up and Run Control

(a) Deck Make-Up

Code the desired instructions in IBM 1130 FORTRAN All cards required for system compatibility have been provided for the submission and execution of a DCIDE program consisting of one hypothetical subroutine (SUB), and the ML. A *LOCAL card has been included to illustrate its format, and its effect on column 30 of the preceding *STORECI card. (The 1 equals the number of *LOCAL cards to follow.) The use of LOCAL's helps to reduce core requirements, and is recommended. (See "IBM 1130 Disk Monitor System, Version 2, Programming and Operator's Guide", if unfamiliar with LOCAL's.) Your deck should be sequenced as follows:

VIII a) DECK MAKE-UP

```
// JOB     2222 4444
// FOR BROWN SMITH AND JONES
* LIST ALL
* ONE WORD INTEGERS
      SUBROUTINE SUB
      INTEGER STAR
      COMMON M1(37,4)
      COMMON S(45),L(45),XI(45),V(45),FGP(15),CC(45),RC(45),
     1 BMR(3),FN(3),XLW(3),STH(3),CTH(3),XIH(3),
     2 SFG(15),LS(15),FC(15),TC(15),TB(15),AS(15),BS(15),CS(15),
     3 XMU(15),SIG(15),XMUX(15),SIGX(15),FCR(15,6),SHAT(15),
     4 WR,XMCT,HC,FC,TC,XIC,BC,COSTY(4),COSTT,
     5 PSTAR,WSTAR,WLAST,XILST,ALPHA(15),BETA(15),TMON,NXBYZ
      COMMON LIMIT,Q(45),MMAX,STAR,DEC(48)
C
C     YOUR SUBROUTINE GOES HERE
C
      RETURN
      END
// DUP
*DELETE             SUB
*STORE       WS  UA  SUB         2222  4444
// FOR BROWN SMITH AND JONES
* IOCS (CARD,1403 PRINTER)
```

```
* LIST ALL
* ONE WORD INTEGERS
* NAME DCIDE
      INTEGER STAR
      COMMON M1(37,4)
      COMMON S(45),L(45),XI(45),V(45),FGP(15),CC(45),RC(45),
     1 BMR(3),DN(3),XLW(3),STF(3),CTF(3),XIF(3),
     2 SFG(15),LS(15),BC(15),TC(15),TB(15),AS(15),BS(15),CS(15),
     3 XML(15),SIG(15),XMLX(15),SIGX(15),FCR(15,6),SHAT(15),
     4 WR,XMCT,FC,FC,TC,XIC,RC,CCSTY(4),CCSTT,
     5 PSTAR,WSTAR,WLAST,XILST,ALPHA(15),BETA(15),TMON,NXBYZ
      COMMON LIMIT,G(45),MMAX,STAR,DEC(48)
      COMMON           TRICL(45)
C
C     YOUR DCIDE ML GOES HERE
C
      CALL LINK (PROSI)
      END
// DUP
*DELETE                DCIDE
*STORECI    WS  FX  DCIDE    12222  4444
*LOCAL,SUB
// XEQ PROSM
CODE
DATA CARDS IF REQUIRED
```

(b) Run Control

Secure and load disc 4444. The game can now be played by placing your deck in the card reader. Note that your version of the DCIDE program will be retained on the computer's files until another team DELETE's yours. Thus, to play the game on a subsequent JOB, you need only the following cards as long as your DCIDE is in file.

VIII b) RUN CONTROL

```
//JOB       2222       4444
//XEQ PROSM
CODE
(data cards as required by user)
```

With a fairly complex DCIDE ML, running time will approximate one minute per month.

IX. Variable Dictionary and Initial Conditions

Following is a list of all variables in COMMON used by the main program.

C,D OO(i), i=1-45 (but i=1-41 ignored); on-order unit amounts for last 4 raw materials.

C,P RC(i), i=1-45 (but i=1-37 ignored); raw material cost, $/unit, charged for raw material receipts.

(b) <u>Department Variables, d = 1-3, for each of the 3 departments</u>

C,P BMR(d); beginning material number, see Section II

C,P BN(d); number of materials made in department d, see Section II

C,D XLW(d); last month's work force, number of employees

C STH(d); straight time man-hours available in department d

C OTH(d); overtime man-hours used in department d

C XIH(d); idle man-hours used in department d

(c) <u>Sales Variables, i=1-15, for each of the 15 final products</u>

C SFG(i); sales of finished goods, new demand (units) generated each period

C US(i); units supplied toward new sales and backorders

C,D BO(i); backorders, units

C TD(i); total demand, units, new demand totalled for year

C TB(i); total backorders, units totalled for year

C,P AS(i); constant for basic demand and forecast generation, see Section IV

C,P BS(i); constant for basic demand and forecast generation, see Section IV

C,P CS(i); constant for basic demand for forecast generation, see Section IV

C,P XMU(i); mean of normal random error E(i) in demand generation (after transformation from XMUX and SIGX, i=13-15)

C,P SIG(i); standard deviation of normal random error E(i) in demand generation (after transformation from XMUX and SIGX, i=13-15)

Some variables are parameters whose values are read at the beginning and do not change during a run. Some variables whose values do change during a run are set initially by reading other data cards. All of these variables and their values are listed on a printing of data cards. Still other variables are set initially and modified by the program itself.

Figure 6 is a listing of input data cards initializing and setting the simulation parameters, with variable names indicated. Parameter and initial data values can be determined from this listing.

The variable dictionary is organized by:

a) material variables
b) department variables
c) sales and forecast variables
d) cost variables
e) linear decision rule variables
f) decision variables
g) miscellaneous
h) special control constants

Associated with the variables are symbols that indicate the variable is in COMMON, and whether it is a parameter or a variable set initially by a data read:

C   COMMON
P   Parameter set as per figure 6
D   Variable set as per figure 6

(a) Material Variables

XM1(i,j), i=45, j=1-45; material mix matrix, see Figure 2 and Section III. Note that M1(i,j) is in common.

C,P S(i), i=1-45 (but i=38-45 ignored); set-up man-hours incurred if any positive amount of product i is produced.

C,P U(i), i=1-45 (but i=38-45 ignored); running man-hours/unit for product i.

C,P Q(i), i=1,45 (but i=38, 45 ignored); manufacturing lot sizes for finished goods, subassemblies and parts. (Used in FGP(i) calculations.)

C,D XI(i), i=1-45; on-hand inventory (units) of product i.

C,D V(i), i=1-45; inventory total $ value for product i.

C,P FGP(i), i=1-15; finished good productivity, the man-hours per unit (of finished product) required to make this product, and to make the subassemblies and parts it requires.

C,P XMUX(i), (but i=1-12 ignored); desired mean of generated log-normal distribution for last 3 products

C,P SIGX(i), (but i=1-12 ignored); desired standard deviation of (generated) log-normal distribution for last 3 products

E(i); generated normal random number

P NRNDN; number of uniform random numbers to be summed in generated normal distribution

C FOR(i,t), i=1-15, t=1-6; forecasts in units for each product i for each of t months ahead, see Section IV

C SHAT(t), t=1-15; aggregate forecast in man-hours for each fo the next 15 months; see Section IV

(d) <u>Cost Variables (see Section V)</u>

C,P WR; wage rate, $/man-hour

C,P XMOT; maximum overtime fraction

C,P HC; hiring cost, $/man hired

C,P FC; firing cost, $ man fired

C,P TC; transfer cost, $/man transferred

C,P XIC; inventory carrying cost, $/$ per month

C,P BC; backorder cost, $/unit newly backordered each month

C COSTY(j), j=1-4; $ cost in the 4 categories, summed for yearly report

1 - hiring, firing, transfer cost
2 - payroll-straight time and overtime cost
3 - inventory holding cost
4 - backorder cost

C COSTT; total $ cost summed for yearly report

(e) <u>Linear Decision Rule Variables (see Section VI)</u>

C PSTAR; suggested total production for coming month in man-hours

C WSTAR; suggested total workforce for coming month in number of employees

C WLAST; last month's actual total workforce

C XILST; last month's net total finished goods inventory, in man-hours (on-hand inventory less backorders)

C,P ALPHA(j), j=1-15; coefficients in P* linear decision rule

C,P BETA(j), j=1-15; coefficients in W* linear decision rule

(f) Decision Variables (see Section III)

C DEC(i), i=1-48; monthly decisions filled in by DCIDE

(g) Miscellaneous Variables

C TMON; month number, floating point, available in COMMON

(h) Special Control Constants

C,D STAR; starting random number, must be even, and between 0 and 9999

C,D LIMIT; print control where 0 gives monthly printout and year summary, and 1 the latter only.

C,D MMAX; number of months in a run

C NXBYZ; not used in this version of simulation.

X. Master Flow Chart of the Main Program

A master flow chart for the main program is given in Figure 7.

BIBLIOGRAPHY: MANAGEMENT GAMES

1. Cohen, K.J., W.R. Dill, A.A. Kuehn, P.R. Winters, The Carnegie Tech Management Game, Homewood, Ill.: Richard D. Irwin, 1964

2. Cohen, K.J. and E. Rhenman, "The Role of Management Games in Education and Research," Management Science, Vol. 7, No. 2 (Jan. 1961).

3. Dill, William R., "What Management Games Do Best," Business Horizons, Vol. 4, No. 3 (Fall 1961).

4. Holt, C.C., F. Modigliani, J.F. Muth and H.A. Simon, Planning Production Inventories, and Work Force, Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1960.

5. Kibbee, J.M. and C.J. Craft, Management Games, New York: Reinhold Publishing Co., 1961.

6. McKenney, James L. and William R. Dill, "Influence on Learning in Simulation Games," American Behavioral Scientist, Vol. 10, No. 2 (October 1966).

7. Porter, Sasien, Marks & Ackoff, "The Use of Simulation as a Pedagogical Device," Management Science, Vol. 12, No. 6 (February 1966).

8. Symonds, Gifford H., "A Study of Management Behavior by Use of Competitive Business Games," Management Science, Vol. 11, No. 1 (September 1964).

9. Thorelli, Hans B. and R.L. Graves, International Operations Simulation, Chicago: The Free Press of Glencoe, 1964.

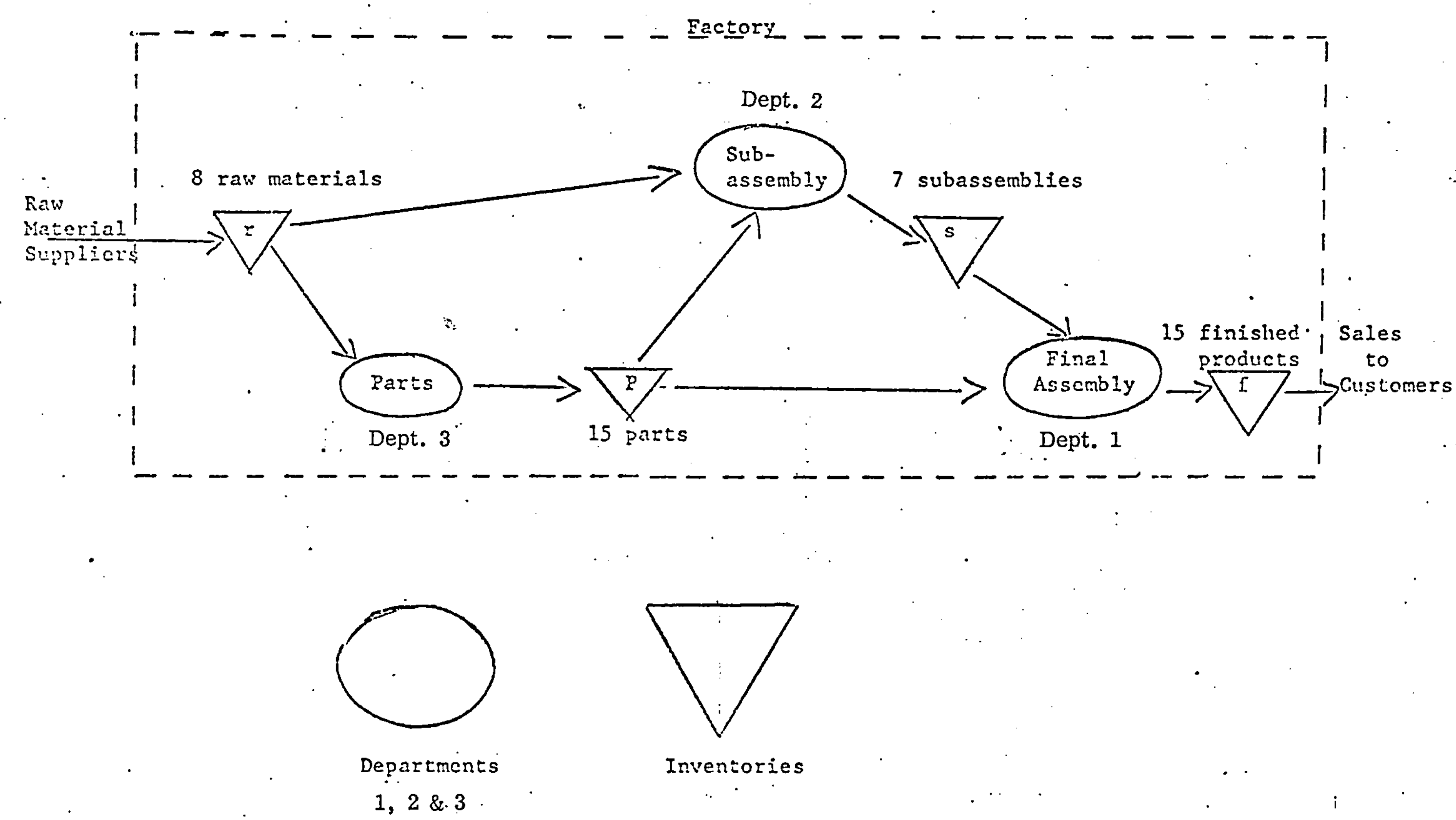


Fig. 1 Schematic of factory

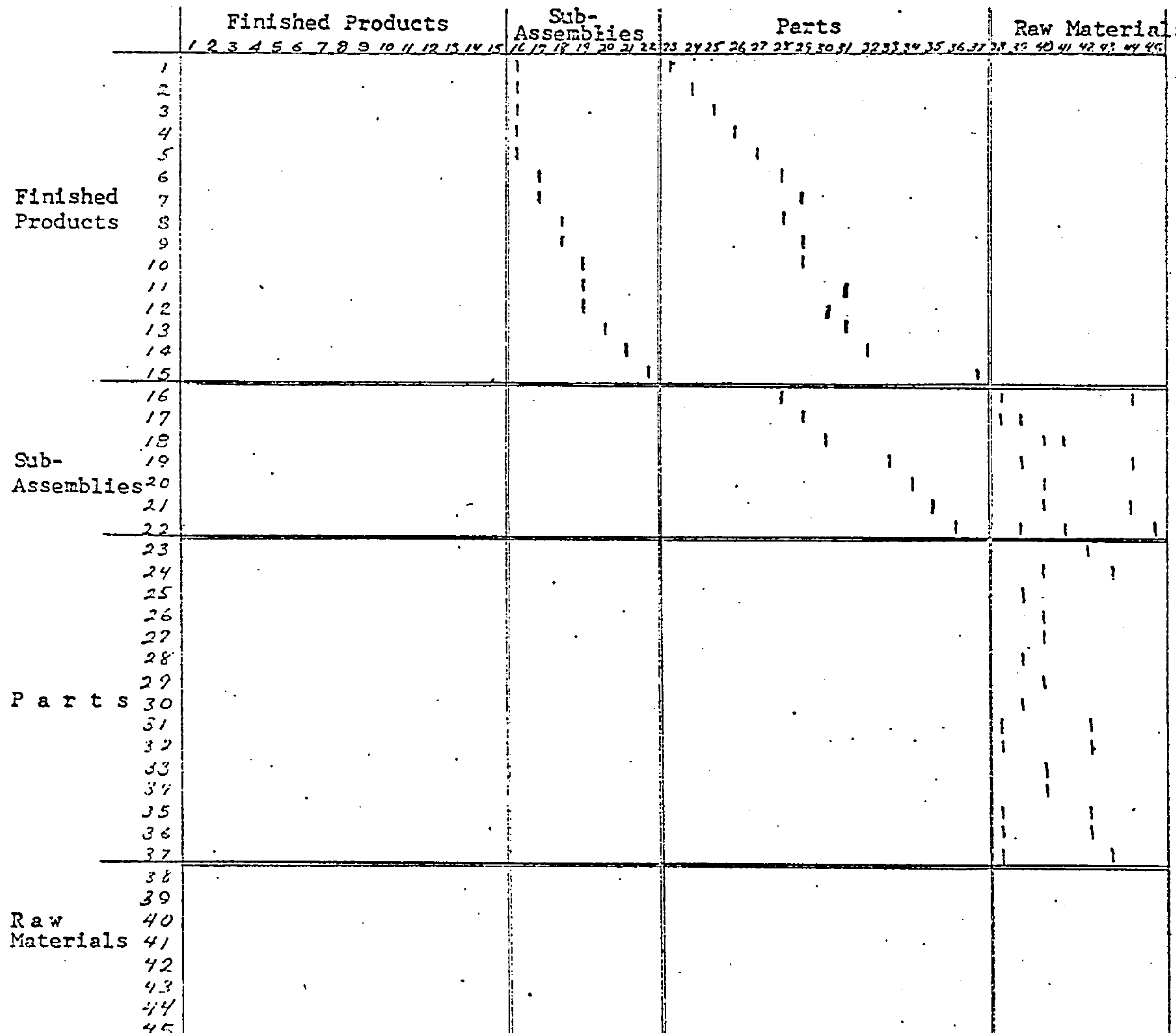


Fig. 2 Product explosion matrix

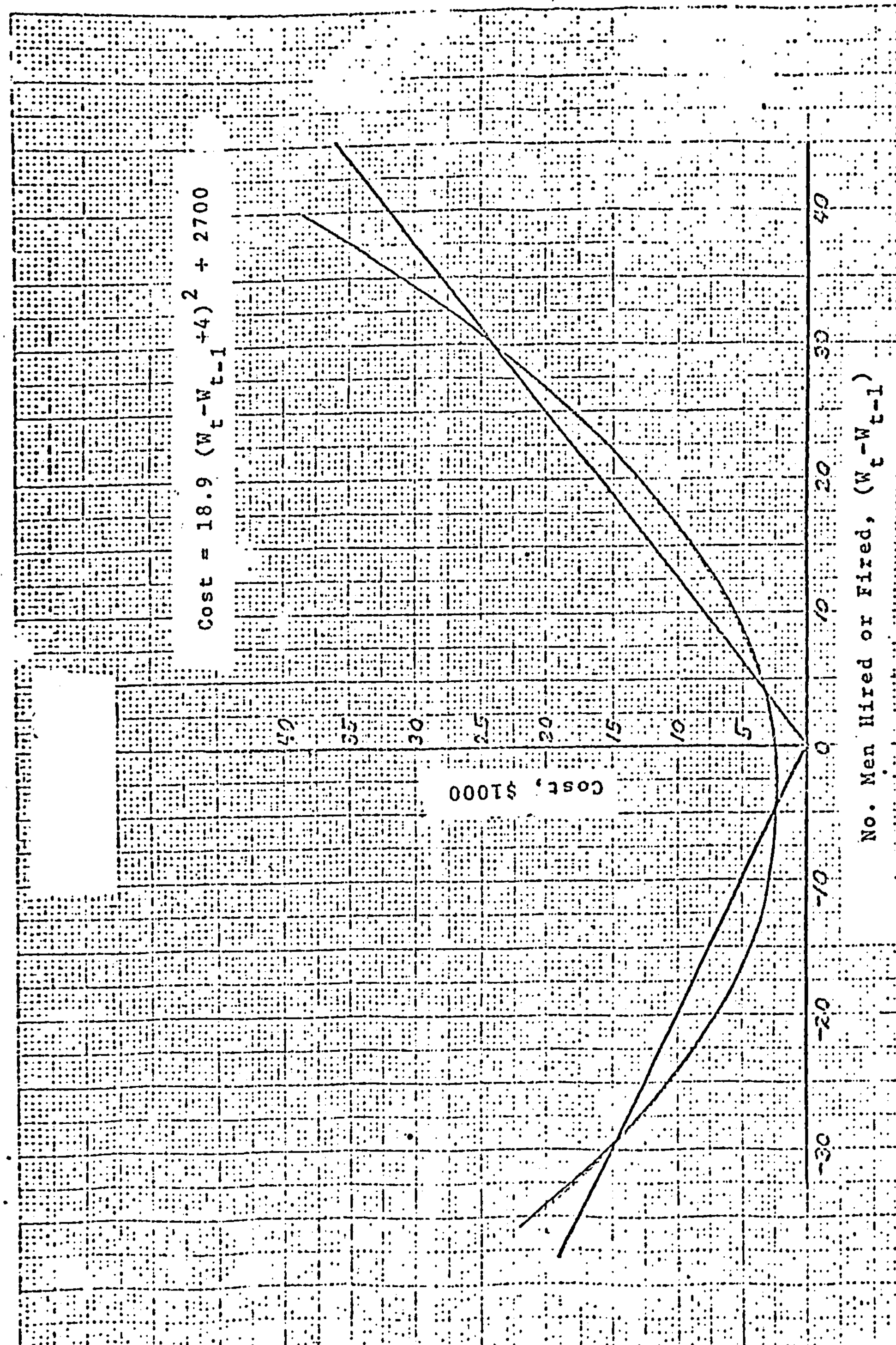


Fig. 3 Hiring and firing costs

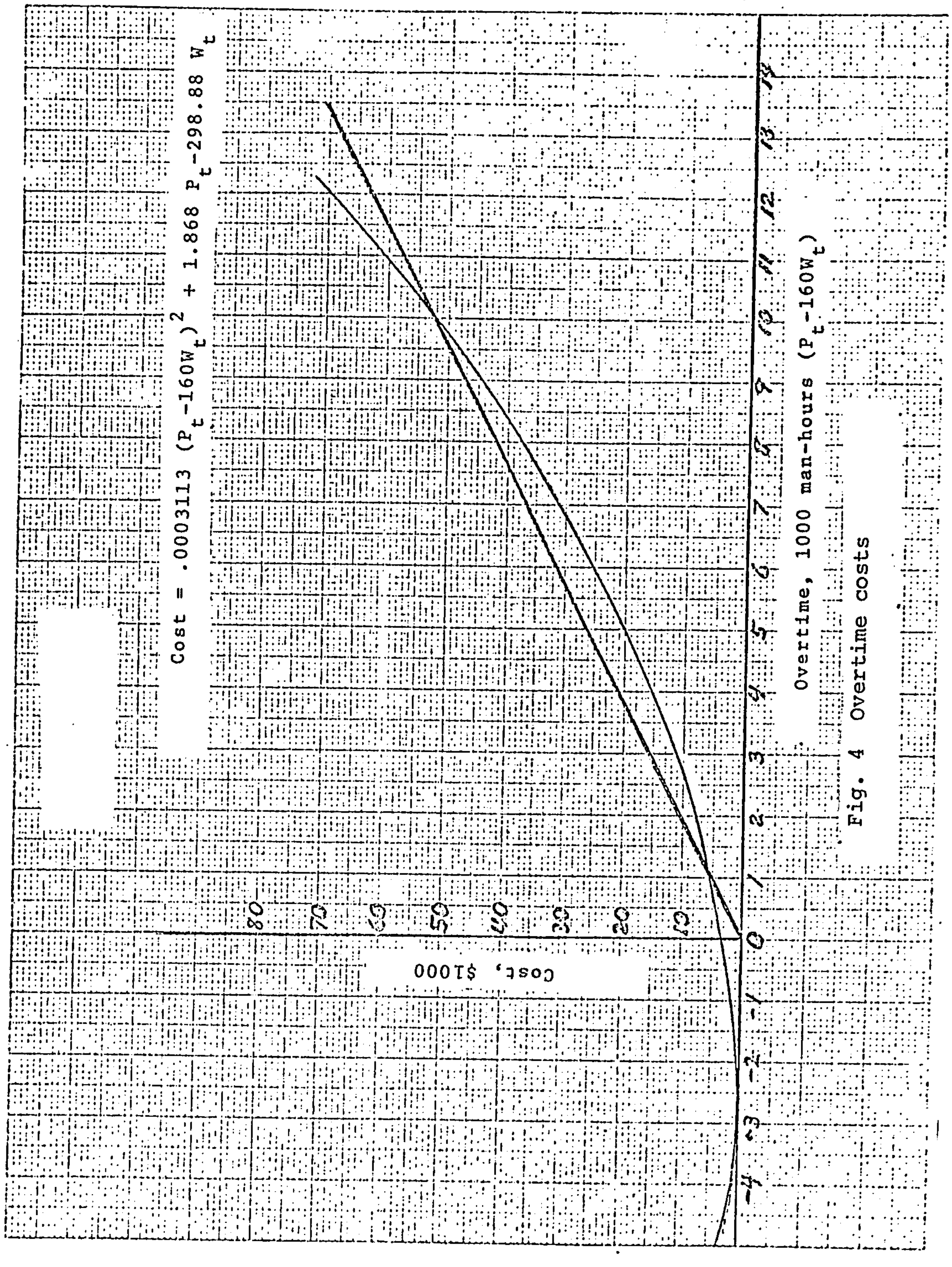


Fig. 4 Overtime costs

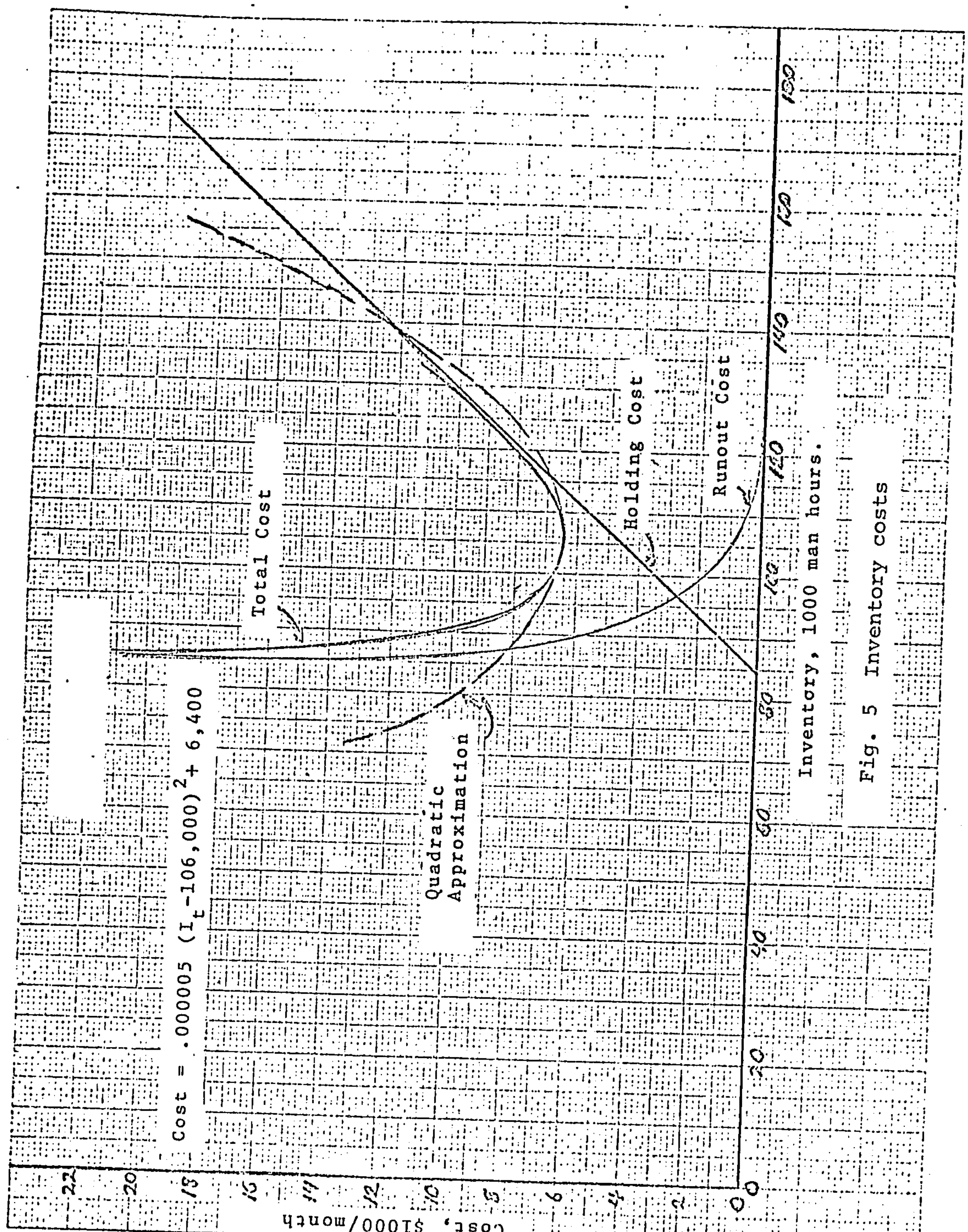
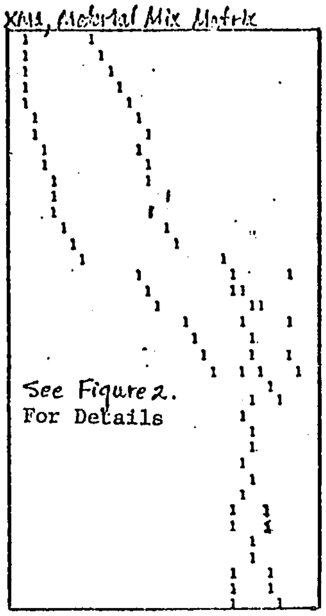


Fig. 5 Inventory costs

XMM, Material Mix Matrix



S(i)	U(i)	XI(i)	V(i)	Q(i)	Data Card No
645.	1.0	1000.	58030.00	2000	1
1500.	2.0	1500.	181350.00	1500	2
300.	5.0	300.	22950.00	600	3
560.	4.0	200.	24590.00	400	4
416.	3.0	100.	12280.00	250	5
378.	2.0	300.	37670.00	600	6
944.	3.0	500.	84350.00	1000	7
294.	3.0	1000.	104300.00	1000	8
686.	2.0	200.	31320.00	400	9
2180.	5.0	750.	130000.00	1500	10
860.	3.0	250.	31330.00	600	11
545.	1.0	600.	65700.00	500	12
2240.	5.0	1500.	270700.00	1500	13
1420.	5.0	600.	95600.00	1200	14
1430.	1.0	800.	128500.00	1600	15
248.	3.0	1850.	88500.00	1850	16
1910.	6.0	2400.	230000.00	2400	17
1060.	5.0	1100.	95000.00	2200	18
213.	2.0	800.	75000.00	800	19
3195.	9.0	1000.	142080.00	2000	20
2440.	7.0	1500.	182500.00	1500	21
1175.	5.0	800.	104900.00	1600	22
63.	1.0	2000.	11220.00	2000	23
89.	2.0	500.	3180.00	500	24
120.	1.0	1000.	8920.00	1000	25
67.	3.0	100.	6165.00	200	26
73.	2.0	150.	8900.00	150	27
60.	.5	3150.	21500.00	3150	28
310.	2.0	1900.	109500.00	1900	29
125.	1.0	2600.	22500.00	2600	30
106.	1.0	600.	9500.00	1200	31
133.	1.5	450.	8000.00	900	32
138.	3.0	800.	49000.00	800	33
690.	1.0	750.	41250.00	1500	34
410.	2.0	750.	14900.00	1500	35
310.	1.0	2400.	38400.00	2400	36
800.	2.0	1600.	36600.00	3200	37
		8175.	81750.00		38
		10575.	52875.00		39
		8625.	431250.00		40
		2850.	28500.00		41
		4500.	9000.00		42
		1950.	9750.00		43
		4425.	88500.00		44
		1200.	96000.00		45

									Data Card No	
BMR(d)	[0	1500	2200]	[1500	700	1500] BN(d)			46	d=1-3
XLW(d)	[137.00	120.00	193.00]						47	d=1-3
BO(i)	0.	0.	0.	0.	0.	0.	0.	0.	48	i=1-15
	0.	0.	0.	0.	0.	0.	0.		49	
AS(i)	1000.00	500.00	200.00	100.00	50.00	300.00	500.00	1000.00	50	i=1-15
	100.00	500.00	200.00	100.00	0.	0.	0.		51	
BS(i)	-8.	10.	5.	0.	1.	-2.	0.	0.	52	i=1-15
	3.	-2.	2.	-1.	0.	0.	0.		53	
CS(i)	-400.	-200.	-100.	-20.	0.	-100.	-200.	-350.	54	i=1-15
	0.	-150.	75.	50.	0.	0.	0.		55	
XMU(i)	0.	0.	0.	0.	0.	0.	0.	0.	56	i=1-15
	0.	0.	0.	0.	0.	0.	0.		57	
SIG(i)	.10	.25	.25	.10	.15	.15	.15	.20	58	i=1-15
	.25	.10	.20	.15	0.	0.			59	
XMUX(i)	[500.00	300.00	800.00]	[75.00	50.00	200.00] SIGX(i)			60	i=13-15
OO(i)	[0.	0.	0.	0.	3000.	1300.	2950.	800.]	61	i=35-45
RC(i)	[10.00	5.00	50.00	10.00	2.00	5.00	20.00	80.00]	62	
ALPHA(i)	0.1137735	0.0957193	0.0775646	0.0655515	0.0454154	0.0325145	0.0219232	0.0135513	63	i=1-15
	0.0071833	0.0025473	-.0006493	-.0026935	13537.798	70.245831	-.1137735		64	
BETA(i)	0.0003590	0.0003351	0.0002973	0.0002528	0.0002069	0.0001632	0.0001238	0.0000899	65	i=1-15
	0.0000617	0.0000392	0.0000219	0.0000092	33.146832	0.7002654	-.0003570		66	
	WR 3.50	XMOTI .30	HC 800.00	FC 500.00	TC 100.00	XTC .02	BC 140.	N 13	67	
	715 STAR	0 LIMIT	12 MMAX						68	

Fig. 6 Initial conditions of simulation

Figure 7.

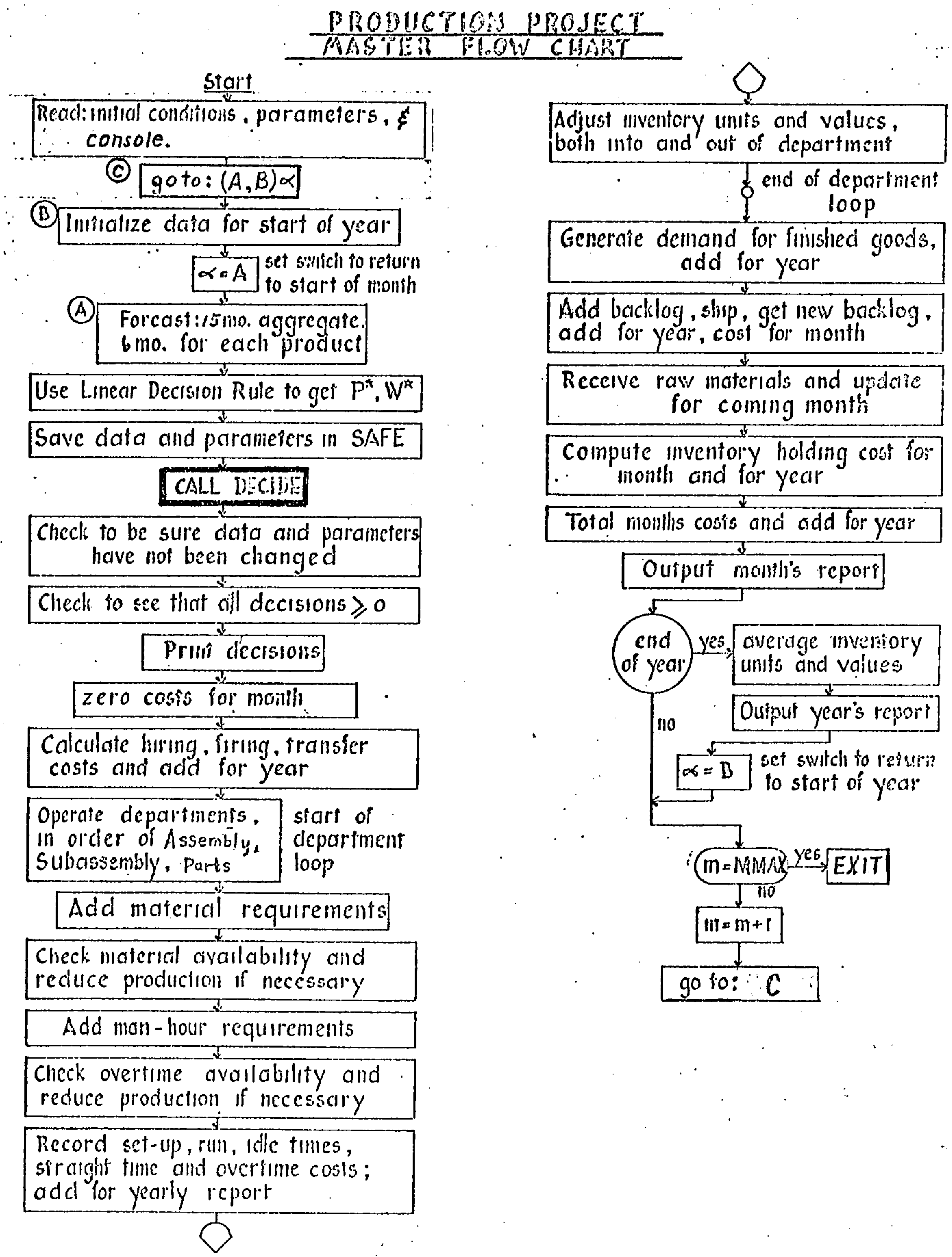

PRODUCTION PROJECT
MASTER FLOW CHART
Start
Read: initial conditions, parameters, & console.
© goto: (A,B)∝
Ⓑ Initialize data for start of year
∝=A
set switch to return to start of month
Ⓐ Forcast: 15mo. aggregate. 6mo. for each product
Use Linear Decision Rule to get P*, W*
Save data and parameters in SAFE
CALL DECIDE
Check to be sure data and parameters have not been changed
Check to see that all decisions ≥ 0
Print decisions
zero costs for month
Calculate hiring, firing, transfer costs and add for year
Operate departments, in order of Assembly, Subassembly, Parts
start of department loop
Add material requirements
Check material availability and reduce production if necessary
Add man-hour requirements
Check overtime availability and reduce production if necessary
Record set-up, run, idle times, straight time and overtime costs; add for yearly report
Adjust inventory units and values, both into and out of department
end of department loop
Generate demand for finished goods, add for year
Add backlog, ship, get new backlog, add for year, cost for month
Receive raw materials and update for coming month
Compute inventory holding cost for month and for year
Total months costs and add for year
Output month's report
end of year
yes
average inventory units and values
Output year's report
no
∝=B
set switch to return to start of year
m=MMAX
yes
EXIT
no
m=m+1
go to: C

Table II

CUMULATIVE LOGNORMAL DISTRIBUTION

If S is lognormal, with mean = m, standard deviation = s, then P(S ≤ L) gives the probability that S will be less than or equal to L.

	L		
P(S ≤ L)	Product 13 (m = 500, s = 75)	Product 14 (m = 300, s = 50)	Product 15 (m = 800, s = 200)
.50	493	296	772
.60	513	308	846
.70	534	324	880
.75	545	330	916
.80	560	340	954
.85	578	351	1002
.90	602	365	1065
.95	639	388	1177
.96	646	395	1200
.97	659	403	1249
.98	672	415	1300
.99	706	433	1394
.999	796	493	1686

XII Appendix: Statistical Reporting Option

The traditional computer output for this simulation is the monthly and yearly reports illustrated in the Computer Printout Exhibits A and B. Each month the decisions generated by DECIDE are recorded. The actual costs for the month are given, together with manpower usage data. Finally a finished goods, subassembly, part and raw material status report records actual production. The yearly report aggregates a selected few statistics, and prepares a simple report at twelve month intervals. (Section I discusses the information available through PROSM, and section VII discusses methods for supressing some of these reports.)

A second level of information is available for teams that desire more detailed information of their operations. By the use of print statements in DCIDE, any of the information contained in common is available. (See section VIII page 12 for a list of COMMON, and section IX for an explanation of the variables).

A third level of information is now available. Teams interested in their performance relative to the other teams engaged in the simulation exercise may employ the program PSTAT to obtain some additional historical records of performance. For the entire course, all DCIDE activities on the 1130 will be monitored. This information is held in an eight by eleven array; Table III displays the index to interpretation of this array. Briefly Block A contains some statistics on the overall simulation usage. Block B records some information on the latest simulation run. Block C contains detailed information on the lowest total cost (and presumably the most successful) run to date. Block D summarizes the relative success of all runs (via a frequency table) based upon the first twelve month total cost figures for all runs to date. Finally, Block E displays some statistics on extended runs. Cost statistics are recorded for the total cost (and etc.) for the year with the highest month simulated in any run to date. The information of cell 4 of Block A records the month (in multiples of 12 and hence the year) for which the information of Block E is relevant. Note that all costs are in millions of dollars.

This kind of information may be valuable in charting your performance relative to your colleague teams. The statistics of PSTAT have been further accumulated at three distinct time levels. The simulation will be divided into several calendar periods to be discussed in class. PSTAT will offer historical statistics as per Table III for the current as well as past calendar periods as follows:

1) First for your team only for the current calendar period, and then anonymously for all teams for the current calendar period ranked according to total costs (cell # 12).

2) Next, information averaged across across all teams for the current calendar period.

3) Finally, the same averaged across all teams, accumulated and averaged since the beginning of the course (all calendar periods to date including the current one).

XII a) EXECUTION OF PSTAT

PSTAT is executed as follows:

```
// JOB   2222 4444
// XEQ  PSTAT
CODE
```

Please note that in using PSTAT you will be required to insert the date on the console in the following format;

```
 --/--/--
(Month)(Date)(Year)
 0-12   1-31   72
```

For example, the 3rd of Feb., 1972 would be logged as 02/03/72

BE CAREFUL.

TABLE III

INDEX TO PSTAT STATISTICS

Location in Array		Information (Millions $)
A. 1-6		General Usage
	1	Number of starts at month 1
	2	Number of starts not at 1
	3	Number of months simulated
	4	Highest month simulated (in multiples of 12)
	5-6	Unused
B. 7-11		Latest Total Cost to Date (For 12 Mo. Run)
	7	Total cost *.001
	8	Hire/Fire/Transfer *.001
	9	Payroll *.001
	10	Inventory *.001
	11	Backorder *.001
C. 12-64		Lowest Total Cost to Date (For 12 Mo. Run)
	12	Total cost *.001
	13	Hire/Fire/Transfer *.001
	14	Payroll *.001
	15	Inventory *.001
	16	Backorder *.001
	17	Unused
	18-32	Demand by product i=1,...,15
	33-47	Backorders by product i=1,...,15
	48-50	Total straight time man-hours for year *.01 by department d=1,2,3.
	51-53	Total overtime man-hours for year *.01 by department d=1,2,3.
	54-56	Total set-up man-hours for year *.01 by department d=1,2,3.
	57-59	Total running man-hours for year *.01 by department d=1,2,3.
	60-62	Total idle man-hours for year *.01 by department d=1,2,3.
	63-64	Unused
D. 65-78		Relative Performance (For 12 Mo. Run)
	65	Frequence total cost < 3600
	66	Frequency total cost ≥ 3600
	67	Frequency total cost ≥ 3800
	68	Frequency total cost ≥ 4000
	69	Frequency total cost ≥ 4200
	70	Frequency total cost ≥ 4400
	71	Frequency total cost ≥ 4600
	72	Frequency total cost ≥ 4800
	73	Frequency total cost ≥ 5000
	74	Frequency total cost ≥ 5200
	75	Frequency total cost ≥ 5400
	76	Frequency total cost ≥ 5600

PSTAT STATISTICS (Continued)

D. 65-78		
	77	Frequency total cost ≥ 5800
	78	Frequency total cost ≥ 6000
E. 79-88		Extended Runs
	79	Lowest total cost
	80	Hire/Fire/Transfer (.001)
	81	Payroll *.001
	82	Inventory *.001
	83	Backorder*.001
	84-88	Unused

Note: For the example contained in Computer Printout Exhibit C, cell 14 contains the value $3.620 million, representing the payroll costs for the Brangate Co. for their lowest total cost run to date in the current calendar period. This does not compare favourably with the two teams ranked ahead of them exhibiting payroll costs of $3.187 and $3.147 millions respectively.

COMPUTER PRINTOUT EXHIBITS

A Monthly Report

```
DECISION FOR MONTH      1

                         FINAL PRODUCTS-PART NO.1 THRU 15
     295.243      C.CCC      C.CCC      C.CCC      0.0CO      113.C72    157.C44    372.981
       0.CCC      C.CCC    171.572      0.CCC      0.CCC        C.CCC    628.18C

                          SUB ASSY.-PART NO.16 THRU 22
    1972.725      C.CCC   1424.755   1791.086    392.612        C.CC0   1884.542

                          PARTS-NO.23 THRU 37
       0.CCC    576.334      C.CCC    167.565      C.CC0     3142.C32   2092.197      C.CCC
    2076.733    471.135   1774.609    785.225      0.0C0        0.CC0    628.180

                          RAW MATERIAL-PART NO.38 THRU 45
    5335.6C6   3962.852   2576.528   1645.C48   3904.7C2     1518.6C8   2209.674   1256.361

                          WORK FORCE
                        DEPT. NO.   NO. OF MEN

                          1          64.C81
                          2         239.243
                          3         131.334


PRODUCTION OF    19 REDUCED BECAUSE MATERIAL   33 WAS SHORT
PRODUCTION OF    18 REDUCED BECAUSE MATERIAL   41 WAS SHORT
PRODUCTION OF    22 REDUCED BECAUSE MATERIAL   41 WAS SHORT
PRODUCTION OF    22 REDUCED BECAUSE MATERIAL   45 WAS SHORT

MONTHLY REPORT FOR MONTH    1
           FFT         PAYROLL            INV       BACKORDER          PSTAR           WSTAR
                                                     COSTS
      15594.92       24340P.87       64498.17          0.C0         70552.29          434.65

TOTAL COST-       3275C1.93


DEPARTMENT OPERATIONS--MAN HOURS

             DEPT. 1          DEPT. 2          DEPT. 3

 STH        10253.C4         38278.88         21C13.48
 OTH            C.CC             C.CC             0.CC
 SUH         4551.CC          5891.CC          2393.00
 RH          3254.36         23186.75         17559.62
 XIH         2447.6C          92C1.12          1C6C.86
```

COMPUTER PRINTOUT EXHIBITS

A Monthly Report (Continued)

FINISHED GOOD STATUS

PRODUCT	AP-UNITS	SFG	US	BQ	FCR	XI-UNITS	V-VALUE
1	295.24	659.70	659.70	0.00	791.99	635.53	37831.21
2	0.00	507.92	507.92	0.00	409.99	592.07	119941.73
3	0.00	194.71	194.71	0.00	154.99	105.28	8054.19
4	0.00	105.83	105.83	0.00	89.99	94.76	11524.86
5	0.00	61.03	61.03	0.00	51.00	38.96	4784.46
6	113.07	270.38	270.38	0.00	247.99	142.68	16370.56
7	157.64	414.81	414.81	0.00	399.99	242.23	41808.31
8	372.98	607.40	607.40	0.00	824.99	765.58	80296.82
9	0.00	132.03	132.03	0.00	103.00	67.96	10643.21
10	0.00	532.80	532.80	0.00	422.99	217.19	37879.63
11	171.57	233.21	233.21	0.00	239.50	108.35	24548.40
12	0.00	146.20	146.20	0.00	124.00	453.79	49690.67
13	0.00	441.77	441.77	0.00	500.00	1058.22	190974.56
14	0.00	260.26	260.26	0.00	300.00	339.73	54131.46
15	628.10	890.99	890.99	0.00	800.00	537.18	87430.12

SUBASSEMBLY,PART,AND RAW MATERIAL STATUS

NUMBER	AP-UNITS	XI-UNITS	V-VALUE	ON ORDER
16	1972.72	3527.48	168604.15	0.00
17	0.00	2129.68	204113.75	0.00
18	1227.01	1994.03	172209.87	0.00
19	800.00	1428.42	134260.59	0.00
20	392.61	1392.61	206654.15	0.00
21	0.00	1500.00	182500.03	0.00
22	1200.00	1371.81	180842.25	0.00
23	0.00	1704.75	9563.68	0.00
24	576.33	1076.33	39224.25	0.00
25	0.00	1000.00	8520.00	0.00
26	167.56	267.56	16537.17	0.00
27	0.00	150.00	8900.00	0.00
28	3142.03	3833.25	26136.57	0.00
29	2092.19	3835.15	220789.43	0.00
30	0.00	1372.98	11801.61	0.00
31	2076.73	2565.16	39343.79	0.00
32	471.13	921.13	16592.58	0.00
33	1774.60	1774.60	107846.78	0.00
34	785.22	1142.61	64080.83	0.00
35	0.00	750.00	14960.00	0.00
36	0.00	1200.00	19200.00	0.00
37	628.18	1600.00	38850.34	0.00
38	0.00	8361.83	83618.28	0.00
39	0.00	9395.82	46979.10	0.00
40	0.00	4565.96	229298.34	0.00
41	0.00	2668.03	20680.35	0.00
42	0.00	4952.13	9904.26	3904.70
43	0.00	2645.48	10227.42	1516.00
44	0.00	4602.27	92045.48	2209.67
45	0.00	800.00	64000.02	1256.36

EXECUTION SUSPENDED AFTER 1 MONTHS.

COMPUTER PRINTOUT EXHIBITS

B Yearly Report

YEARLY REPORT,END OF MONTH 12

TOTAL COST IS- 4612672.01

HFT	PAYROLL	INV	BACKORDER COSTS
239615.96	3568578.00	749328.00	55151.03

	TOTAL DEMAND-UNITS	TOTAL BACKORDERS-UNITS
1	11157.77	0.00
2	7778.14	389.96
3	3088.29	3.97
4	1235.66	0.00
5	649.64	0.00
6	3553.73	0.00
7	5934.28	0.00
8	11411.98	0.00
9	1471.15	0.00
10	6057.50	0.00
11	2773.39	0.00
12	1164.14	0.00
13	5710.20	0.00
14	3462.71	0.00
15	9961.84	0.00

DEPARTMENT OPERATIONS

	DEPT.1	DEPT.2	DEPT.3
TSTH	359864.18	357544.31	202496.59
TOTH	12049.26	18640.40	9102.83
TSUH	147245.03	93813.01	31402.00
TRH	188139.18	295836.68	175758.43
TXIH	36529.40	26735.05	4439.02

EXECUTION SUSPENDED AFTER 12 MONTHS.
// DUP

*DUMP FX WS DCIDC
CART ID 4444 DB ADDR 14FC DB CNT 0060

COMPUTER PRINTOUT EXHIBITS

C PSTAT Statistics

```
FACE 1

// JCE    2222 4444

LCG CRIVE   CART SPEC   CART AVAIL  PHY CRIVE
  CCCC        2222        2222        CCCC
  CCC1        4444        4444        CCC1

V2 FCE  ACTLAL  EK  CCNFIG  EK

// XEC PSTAT

        PRCSIM INSFECTICN CNLY, PLEASE CATE
```

CELL NUMBER

FILE CCNTENTS AS CF 12//1/7C
INCIVICLAL TEAM STATISTICS

2E	C	325	24	2354	25	511C	112	3554	EE1	521
4659	54	362C	877	67	2354-	1212E	7269	3319	1112	EE2
3434	5717	11522	1538	555C	2413	11C3	5659	3645	55CC	C
C	C	23	C	39	E9	C	0	57	C	2E
C	33	212	2724	45C3	2325	154	242	13C	E54	122E
419	1571	34CE	1965	54	11C	7C	C	C	C	C
C	C	C	C	5	4	7	0	1	C	C
2	4823	65	3777	856	89	2354	C	—	—	—

ERANGATE

RANKEC PERFCRMANCES CF ALL TEAMS

3	E	171	48	197C	2	4351	2	3258	1C4E	45
4173	33	3187	5CC	52	C	11157	7778	3CEE	1235	449
3553	5534	11411	1471	4C57	2773	1164	531C	3462	5561	31
C	C	EE	42	C	C	C	C	21E	C	C
C	C	C	277C	4C34	2256	C	1	2	EEC	E43
24C	15CE	3392	2C57	1	C	C	C	C	C	C
C	1	7	C	C	C	C	C	C	C	C
C	4422	1C	3259	1C62	1CC	197C	C			

42	19	2CE	24	C	C	4259	127	3147	1C24	C
4259	127	3147	1C24-	C	C	11157	7778	3CEE	1235	E49
3553	5534	11411	1471	6C57	2773	1164	571C	3462	5561	C
C	C	C	C	C	C	C	C	C	C	C
C	C	C	2651	3858	2252	26	53	2C	E58	5C1
15C	2CC5	3367	2132	13	42	29	0	C	C	C
C	C	2	3	C	2	C	C	C	C	C
1	4523	214	3169	1C7C	6E	C	C.			

28	C	325	24	2354	25	511C	112	3554	EE1	521
4659	54	362C	877	67	2354	1212E	7269	3319	1112	662
3434	5717	11522	1538	555C	2413	1C3	5659	3645	55CC	C
C	C	22	C	35	E9	C	C	57	C	2E
C	33	212	2724	45C3	2325	154	242	13C	E54	122E
419	1571	34CE	1565	54	11C	7C	C	C	C	C
C	C	C	C	5	4	7	C	1	C	C
2	4833	69	3777	E5E	E9	2354	C			

AVERAGEC STATISTICS ACRCSS ALL TEAMS THIS PERICC

32	5	271	31	7EE	3	4E3E	153	3459	555	2EE
44E4	1C2	2467	EE4	2E	54C	1127E	7714	211E	1219	E5C
3538	55CE	11424	1475	6C4E	272E	115E	57C8	3424	5553	19
1E	17	14	5	12	34	E	C	3E	C	3
C	4	2E	25C7	3E75	2221	15E	27E	129	11CC	ECE
287	1547	3256	15E2	5E	65	E1	C	C	C	C
C	C	1	3	3	2	1	C	C	C	C
1	482C	55	3669	527	123	54C	141			

AVERAGEC STATISTICS ACRCSS ALL TEAMS TC CATE CVER ALL PERICCS

15	2	15E	17	E21	C	4EEE	119	3553	544	241
4565	1CE	3531	E5C	22	1C41	111E5	7E21	2C74	1219	E5E
3517	5543	115C2	14E2	6C4E	271E	114E	5749	34E4	5541	E
E7	21	11	4	3	1E	1C	C	49	C	C
C	C	E	2547	2E54	231E	224	2E4	129	112C	771
272	15EE	32EC	2C2E	E3	1C5	143	C	C	C	C
C	C	C	C	1	C	C	C	C	C	C
C	3CEE	57	2322	577	121	359	32			

ENC CF PSTAT

```
,T5000,CM100000,MT1.
ACCOUNT(310040635).
FTN.
SETCORE.
LGO.

      PROGRAM PRSIM(INPUT,OUTPUT,TAPE2,TAPE5=INPUT,TAPE6=OUTPUT)
      INTEGER      WSTAR,SA(45,120)
      DIMENSION TTSTH(3),TTCTH(3),TTSUH(3),TTRH(3),TTIH(3),TCOSTY(5)
      DIMENSION ITITC(15),ITITB(15),COSTMY(100)
      COMMON IXM1(37,30),S(37),U(37),IXI(45),V(45),STH(3),CTH(3),SUH(3),
     1XIH(3),IBMR(3),IBN(3),IXLW(3),ISFG(15),IUS(15),IBC(15),ISL(15),
     2ITC(15),ITB(15),ITSL(15),AS(15),BS(15),CS(15),IFCR(15,6),COSTY(5),
     3COST(5),COSTT,SHAT(12),ALPHA(15),BETA(15),XMU(15),SIG(15),FGF(15),
     4XMUX(3),SIGX(3),ICG(8),RC(8),E(15),A(15),B(15),IXM(45),DEC(48),
     5AP(45),ITI(45),TV(45),TSTH(3),TCTH(3),TSUH(3),TRH(3),RH(3),IC(45)
      COMMON STAR,WSTAR,PSTAR,WR,XMOT,IHC,ITC,IFC,XIC,IBC,ISC,N,MMAX,M
      COMMON COSTM,TIH(3),JFCR(45,18),JSE(45,18),ZZ,ZY,ZX,ZW
      COMMON IXYZ(15,30,4)
      COMMON IDEC(15,6),ICQ(15,6)
      DO 8901 I=1,15
8901  IBO(I)=0
C        READ INITIAL CONDITIONS
      DO 10 I=1,37
10    READ(5,8000)(IXM1(I,J),J=1,30),S(I),U(I),IXI(I),V(I),IC(I)
      DO 2113 I=1,37
      DO 2113 J=1,30
2113  IXM1(I,J)=IABS(IXM1(I,J))
8000  FORMAT(30I1,2F10.2,I5,F10.2,I5)
      DO 830 I=38,45
830   READ(5,8010)IXI(I),V(I)
8010  FORMAT(I5,F10.2)
      READ(5,8001)IBMR,IBN,IXLW
8001  FORMAT(9I5)
      READ(5,8002) AS,BS,CS
8002  FORMAT(8F10.2,/,7F10.2)
      READ(5,8002)XMU,SIG
      READ(5,8004)ICG,RC
8004  FORMAT(8I5/8F10.2)
      READ(5,8005) ALPHA,BETA
8005  FORMAT(8F10.7/7F10.7)
      READ(5,8006)WR,XMOT,IHC,IFC,ITC,XIC,IBC,ISC,N
8006  FORMAT(2F10.2,3I5,F5.2,3I5)
      READ(5,8007)STAR,MMAX
8007  FORMAT(F5.0,I5)
C     WRITE INITIAL CONDITIONS ON TAPE 2
      REWIND 2
      DO 90030 I=1,45
      IF(I-37)90040,90040,90050
90040 WRITE(2,9017)(IXM1(I,J),J=1,30),S(I),U(I),IXI(I),V(I),IC(I)
9017  FORMAT(1X,30I1,2F10.2,I10,F10.2,I5)
      GO TO 90030
90050 WRITE(2,9025)IXI(I),V(I)
9025  FORMAT(1X,I10,F10.2)
90030 CONTINUE
      WRITE(2,9018)IBMR,IBN,IXLW
9018  FORMAT(1X,9I10)
      WRITE(2,9019)AS,BS,CS,XMU,SIG
9019  FORMAT(1X,8F10.2/1X,7F10.2)
      WRITE(2,9021)ICG,RC
```

```
9021  FORMAT(1X,8I1C/1X,8F10.2)
      WRITE(2,9022)ALPHA,BETA
9022  FCRMAT(1X,8F15.7/1X,7F15.7)
      WRITE(2,9023)WR,XMOT,IHC,IFC,ITC,XIC,IBC,ISC,N
9023  FORMAT(1X,2F1C.2,3I10,F10.2,3I10)
      WRITE(2,9024)STAR,MMAX
9024  FORMAT(1X,F1C.2,I10)
      DO 1200 I=1,15
      DO 1200 J=16,45
      DO 1200 K=1,4
1200  IXYZ(I,J-15,K)=0
      DO 2000 I=16,45
      IF(I-22)2010,201C,2020
2010  DO 2030 J=1,15
      IF(IXM1(J,I-15))203C,2030,2C40
2040  IXYZ(J,I-15,1)=IXYZ(J,I-15,1)+1
2030  CCNTINUE
      GO TC 2000
2020  IF(I-37)2050,2050,2060
2050  DO 2070 J=1,15
      IF(IXM1(J,I-15))207C,207C,2080
2080  IXYZ(J,I-15,1)=IXYZ(J,I-15,1)+1
2070  CONTINUE
      DO 2090 J=16,22
      DO 2090 K=1,15
      ITEMP=IXM1(K,J-15)*IXM1(J,I-15)
      IF(ITEMP)2090,2090,2100
2100  IXYZ(K,I-15,2)=IXYZ(K,I-15,2)+1
2090  CONTINUE
      GO TC 2000
2060  IF(I-41)2110,2110,2120
2110  DO 2130 J=16,37
      DO 2130 K=1,15
      ITEMP=IXM1(K,J-15)*IXM1(J,I-15)
      IF(ITEMP)2130,213C,2140
2140  IXYZ(K,I-15,2)=IXYZ(K,I-15,2)+1
2130  CCNTINUE
      DO 2150 J=23,37
      DO 2150 K=16,22
      DO 2150 L=1,15
      ITEMP=IXM1(L,K-15)*IXM1(K,J-15)*IXM1(J,I-15)
      IF(ITEMP)215C,2150,216C
2160  IXYZ(L,I-15,3)=IXYZ(L,I-15,3)+1
2150  CONTINUE
      GO TC 2000
2120  DO 2170 J=16,37
      DO 2170 K=1,15
      ITEMP=IXM1(K,J-15)*IXM1(J,I-15)
      IF(ITEMP)2170,2170,2180
2180  IXYZ(K,I-15,3)=IXYZ(K,I-15,3)+1
2170  CONTINUE
      DO 2190 J=23,37
      DO 2190 K=16,22
      DO 2190 L=1,15
      ITEMP=IXM1(L,K-15)*IXM1(K,J-15)*IXM1(J,I-15)
      IF(ITEMP)219C,2190,220C
2200  IXYZ(L,I-15,4)=IXYZ(L,I-15,4)+1
2190  CONTINUE
2000  CCNTINUE
      DO 730 IZZ=25,25,5
```

```
      ZZ=IZZ/10.
      DO 730 IZY=15,15,5
      ZY=IZY/10.
      DO 730 IZX=25,25,5
      ZX=IZX/10.
      DO 730 IZW=40,40,5
      ZW=IZW/10.
      M=0
C     READ INITIAL CONDITIONS FROM TAPE 2
      REWIND 2
      DO 90010 I=1,37
90010 READ(2,9017)(IXM1(I,J),J=1,30),S(I),U(I),IXI(I),V(I),IC(I)
      DO 90020 I=38,45
90020 READ(2,9025)IXI(I),V(I)
      READ(2,9018)IBMR,IBN,IXLW
      READ(2,9019) AS,BS,CS,XMU,SIG
      READ(2,9021)ICC,RC
      READ(2,9022) ALPHA,BETA
      READ(2,9023)WR,XMCT,IHC,IFC,ITC,XIC,IBC,ISC,N
      READ(2,9024)STAR,MMAX
800   M=M+1
      IF(M-13)1337,1338,1337
1338  DO 1339 I=1,5
1339  TCOSTY(I)=0.
      TCOSTT=0.
      DO 1336 IC=1,3
      TTSTH(IC)=0.
      TTOTH(IC)=0.
      TTSUH(IC)=0.
      TTRH(IC)=0.
1336  TTIH(IC)=0.
      DO 2938 I=1,15
      ITITC(I)=0
2938  ITITB(I)=0
1337  CONTINUE
      IF(M-1)100,100,110
C        CONSTANTS FOR RANDOM EFFECT GENERATION
100   DO 40 I=1,15
      A(I)=2.*SIG(I)*SQRT(3./FLOAT(N))
40    B(I)=FLOAT(N)/2.
110   I=M/12
      I=M-12*I-1
      IF(I)750,760,750
C        INITIALIZE TO ZERO FOR START OF YEAR
760   DO 8902 I=1,45
      ITI(I)=0
8902  TV(I)=0.
      DO 8903 I=1,3
      TSTH(I)=0.
      TOTH(I)=0.
      TSUH(I)=0.
      TRH(I)=0.
8903  TIH(I)=0.
      DO 8904 I=1,15
      ITO(I)=0
8904  ITB(I)=0
      DO 8905 I=1,5
8905  COSTY(I)=0.
      COSTT=0.
C        START OF MONTHLY LOOP
```

```
750     CONTINUE
C          18 MONTHS OF FORECASTS FOR EACH ITEM IN UNITS
        DO 50 J=1,18
        XM=M-1+J
        T=SIN(3.14159*(XM/6.))
        DO 60 I=1,15
        JFCR(I,J)=AS(I)+BS(I)*XM+CS(I)*T+0.5
        IF(JFCR(I,J))70,60,60
70      JFCR(I,J)=0
60      CONTINUE
50      CONTINUE
C          AVERAGE MAN-HOUR CALCULATIONS FOR FGP'S
        DO 200 I=1,15
        FGP(I)=U(I)+S(I)/IQ(I)
        DO 210  J=16,37
        IF(IXM1(I,J-15))210,210,220
220     FGP(I)=FGP(I)+U(J)+S(J)/IQ(J)
        IF(J-23)230,210,210
230     DO 240 K=23,37
        IF(IXM1(J,K-15))240,240,250
250     FGP(I)=FGP(I)+U(K)+S(K)/IQ(K)
240     CONTINUE
210     CONTINUE
200     CONTINUE
C          12 MONTHS FORECAST IN AGGREGATE MANHOURS
        DO 90 IT=1,12
        SHAT(IT)=0.
        T=M+IT-1
        DO 111 I=1,15
        SHAT(IT)=SHAT(IT)+FGP(I)*(AS(I)+BS(I)*T+CS(I)*SIN(3.14159*T/6.))
        IF(SHAT(IT))120,111,111
120     SHAT(IT)=0.
111     CONTINUE
90      CONTINUE
C          GET WORKFORCE AND INVENTORY AND APPLY LINEAR DECISION RULES
        XILST=0.
        IWLST=0
        DO 125 I=1,3
125     IWLST=IWLST+IXLW(I)
        DO 130 I=1,15
130     XILST=XILST+FGP(I)*(IXI(I)-IBO(I))
        PSTAR=ALPHA(13)+ALPHA(14)*IWLST+ALPHA(15)*XILST
        WSTAR=BETA(13)+BETA(14)*IWLST+BETA(15)*XILST+0.5
        DO 140 IT=1,12
        PSTAR=PSTAR+ALPHA(IT)*SHAT(IT)
140     WSTAR=WSTAR+BETA(IT)*SHAT(IT)+0.5
C          CHECK FOR NEGATIVE PSTAR AND WSTAR - SET TO ZERO IF THEY ARE
        IF(PSTAR)150,160,160
150     WRITE(6,9000)PSTAR
9000    FORMAT(1H0,10X,17HPSTAR IS NEGATIVE,F15.4)
        PSTAR=0.
160     IF(WSTAR)170,180,180
170     WRITE(6,9001)WSTAR
9001    FORMAT(1H0,10X,17HWSTAR IS NEGATIVE,I10)
        WSTAR=0
180     CONTINUE
        TMON=M
103     DO 122 I=1,15
        DO 122 MA=1,18
122     JSE(I,MA)=SIG(I)*JFCR(I,MA)+0.5
```

```
C         CALL DECIDE TO MAKE PERIOD DECISIONS                                  PRSIM240
      CALL DECIDE                                                               PRSIM241
C         CHECK DECISIONS AND SET TO ZERO IF NEGATIVE                           PRSIM242
      DO 190 I=1,48                                                             PRSIM243
      IF(DEC(I))201,190,190                                                     PRSIM244
  201 DEC(I)=0                                                                  PRSIM245
  190 CONTINUE                                                                  PRSIM246
      DO 8906 I=1,4                                                             PRSIM247
 8906 COST(I)=0.                                                                PRSIM248
C         CALCULATE HIRING AND FIRING COSTS                                     PRSIM249
      T1=DEC(46)+DEC(47)+DEC(48)                                                PRSIM250
      T2=IXLW(1)+IXLW(2)+IXLW(3)                                                PRSIM251
      T=T1-T2                                                                   PRSIM252
      IF(T)221,231,211                                                          PRSIM253
  211 COST(1)=T*IHC                                                             PRSIM254
      GO TO 231                                                                 PRSIM255
  221 COST(1)=-T*IFC                                                            PRSIM256
  231 CONTINUE                                                                  PRSIM257
C         CALCULATE TRANSFER COSTS                                              PRSIM258
      DO 241 I=1,3                                                              PRSIM259
      T1=DEC(I+45)-IXLW(I)                                                      PRSIM260
      IF(T1)241,241,251                                                         PRSIM261
  251 IF(T)261,261,271                                                          PRSIM262
  271 IF(T-T1)281,281,291                                                       PRSIM263
  281 T1=T1-T                                                                   PRSIM264
      T=0.                                                                      PRSIM265
  261 COST(1)=COST(1)+T1*ITC                                                    PRSIM266
      GO TO 241                                                                 PRSIM267
  291 T=T-T1                                                                    PRSIM268
  241 IXLW(I)=DEC(I+45)                                                         PRSIM269
C         START DEPARTMENT LOOP                                                 PRSIM270
      DO 300 ID=1,3                                                             PRSIM271
C         OPERATE DEPARTMENTS                                                   PRSIM272
C         ZERO MATERIAL USEAGE                                                  PRSIM273
      DO 1 I=1,45                                                               PRSIM274
    1 IXM(I)=0                                                                  PRSIM275
C     GET MATERIAL LOWER AND UPPER LIMITS FOR THIS DEPARTMENT                   PRSIM276
      IL=IBMR(ID)+1                                                             PRSIM277
      IU=IBMR(ID)+IBN(ID)                                                       PRSIM278
C         ADD MATERIAL REQUIREMENTS                                             PRSIM279
      DO 310 IP=IL,IU                                                           PRSIM280
      JZ=IU+1                                                                   PRSIM281
      DO 320 IZ=JZ,45                                                           PRSIM282
  320 IXM(IZ)=IXM(IZ)+IXM1(IP,IZ-15)*DEC(IP)                                    PRSIM283
  310 CONTINUE                                                                  PRSIM284
C         CHECK MATERIAL AVAILABILITY                                           PRSIM285
      DO 330 J=16,45                                                            PRSIM286
      IF(IXI(J))340,350,350                                                     PRSIM287
  340 IXI(J)=0                                                                  PRSIM288
  350 IF(IXM(J)-IXI(J))330,330,360                                              PRSIM289
  360 FA=FLOAT(IXI(J))/FLOAT(IXM(J))                                            PRSIM290
C         REDUCE DESIRED PRODUCTION                                             PRSIM291
      DO 370 IP=IL,IU                                                           PRSIM292
      IF(IXM1(IP,J-15)-1)370,380,380                                            PRSIM293
  380 IF(DEC(IP))390,390,400                                                    PRSIM294
  400 WRITE(6,9003)IP,J                                                         PRSIM295
 9003 FORMAT(1H0,13HPRODUCTION OF,I5,25H REDUCED BECAUSE MATERIAL,I5,10HPRSIM296
     1 WAS SHORT)                                                               PRSIM297
  390 DO 410 IZ=16,45                                                           PRSIM298
  410 IXM(IZ)=IXM(IZ)+IXM1(IP,IZ-15)*DEC(IP)*(FA-1.)+0.5                        PRSIM299
```

```
      DEC(IP)=IFIX(FA*DEC(IP)+0.5)
370   CONTINUE
330   CONTINUE
C        END MATERIAL AVAILABILITY CHECK
C        ZERO MANHOUR REQUIREMENTS
      T=0.
C        GET MANHOUR REQUIREMENTS
      DO 420 IP=IL,IU
      IF(DEC(IP))420,420,430
430   T=T+S(IP)+DEC(IP)*U(IP)
420   CONTINUE
C        NOW HAVE MANHOUR REQUIREMENTS
C        STRAIGHT TIME MAN HOURS
      ST=160.*DEC(ID+45)
C        CHECK OVERTIME AND CALCULATE OVERTIME AND IDLETIME
      IF(T-ST)440,440,450
440   OTH(ID)=0.
      XIH(ID)=ST-T
      FB=1.
      GO TO 460
450   IF(ST)470,470,480
470   FB=0.
      T=0.
      GO TO 490
480   FB=T/ST
      IF(FB-XMOT)485,485,489
485   FB=1.
      GO TO 490
C        GET FACTOR TO REDUCE PRODUCTION
489   FB=XMOT/FB
      T=XMOT*ST
      WRITE(6,9004)ID
9004  FORMAT(1H0,24HPRODUCTION IN DEPARTMENT,I3,43H WAS REDUCED BECAUSE
     1OF OVERTIME LIMITATION)
490   OTH(ID)=T-ST
      XIH(ID)=0.
460   TR=0.
      TS=0.
C        OVERTIME FUDGE FACTOR
      IF(ST+OTH(ID))461,461,462
461   G=0.
      GO TO 463
462   CONTINUE
      G=((ST+1.5*OTH(ID))/(ST+OTH(ID)))*WR
463   CONTINUE
C        REDUCE PRODUCTION IF NECESSARY AND ADD SETUP AND RUN TIMES
      DO 500 IP=IL,IU
      DEC(IP)=IFIX(FB*DEC(IP)+0.5)
      AP(IP)=DEC(IP)
      IF(DEC(IP))500,500,510
510   TS=TS+S(IP)
      TR=TR+DEC(IP)*U(IP)
C        ADD IN LABOUR COSTS
      T=(S(IP)+DEC(IP)*U(IP))*G
      DO 520 IZ=1,30
      IF(IXM1(IP,IZ)-1)520,530,530
530   Z=V(IZ+15)/IXI(IZ+15)
C        VALUE ADDED FROM INVNETORY
      T=T+DEC(IP)*IXM1(IP,IZ)*Z
      IXI(IZ+15)=IXI(IZ+15)-DEC(IP)*IXM1(IP,IZ)
```

```
      V(IZ+15)=IXI(IZ+15)*Z
520   CONTINUE
C        ACC CUTPUT INVNETORY UNITS AND VALUE
      IXI(IP)=IXI(IP)+CEC(IP)
      V(IP)=V(IP)+T
500   CONTINUE
C        ENC CF PRCCUCTION
C        RECCRC STRAIGHT TIME HOURS
      STH(IC)=ST
      SUH(IC)=TS
      RH(IC)=TR
C        ACC FCR YEARLY REPORT
      TSTH(IC)=ST+TSTH(IC)
      TCTH(IC)=CTH(IC)+TOTH(ID)
      TSUH(IC)=TS+TSUH(IC)
      TRH(IC)=TR+TRH(IC)
      TIH(IC)=XIH(IC)+TIH(ID)
C        PAYRCLL CCSTS
      COST(2)=COST(2)+(ST+1.5*OTH(IO))*WR
300   CONTINUE
C        ENC CF CEPARTMENT LOOP
C        START CEMANC ANC SALES CALCULATIONS
      DO 540 I=1,15
C        RANDCM NUMBER
      CALL RANSET(STAR)
      T=0.
      CO 550 J=1,N
550   T=T+RANF(-1)
540   E(I)=A(I)*(T-B(I))+XMU(I)
      CALL RANGET(STAR)
C        CEMANC BY ITEM
      DO 560 I=1,15
      ISFG(I)=JFCR(I,1)*(E(I)+1.)+0.5
560   SA(I,M)=ISFG(I)
C        FINISHEC GCCCS PRCCUCT ITERATION
      DO 910 I=1,15
C        CEMANC INCLUCES BACKLOG
      JC=ISFG(I)+IBC(I)
C        CHECK FCR ENCUGH INVENTORY
      IF(IXI(I)-JC)580,590,590
580   IUS(I)=IXI(I)
      IBO(I)=JC-IXI(I)
      GO TC 600
590   IUS(I)=JC
      IBO(I)=0
600   IF(IXI(I))601,601,602
601   Z=0.
      GO TC 603
602   Z=V(I)/IXI(I)
603   CONTINUE
C        RECUCE UNITS ANC VALUE ACCED BY AMOUNT SHIPPED
      V(I)=V(I)-IUS(I)*Z
      IXI(I)=IXI(I)-IUS(I)
C     TOTALS FOR YEAR
      ITC(I)=ITC(I)+ISFG(I)
      ITB(I)=ITB(I)+IBC(I)
C        BACKCRCER CCST
      COST(4)=COST(4)+IBC*FLCAT(IBO(I))
910   CONTINUE
C        ENC CF FINISHEC GCCCS LOOP
```

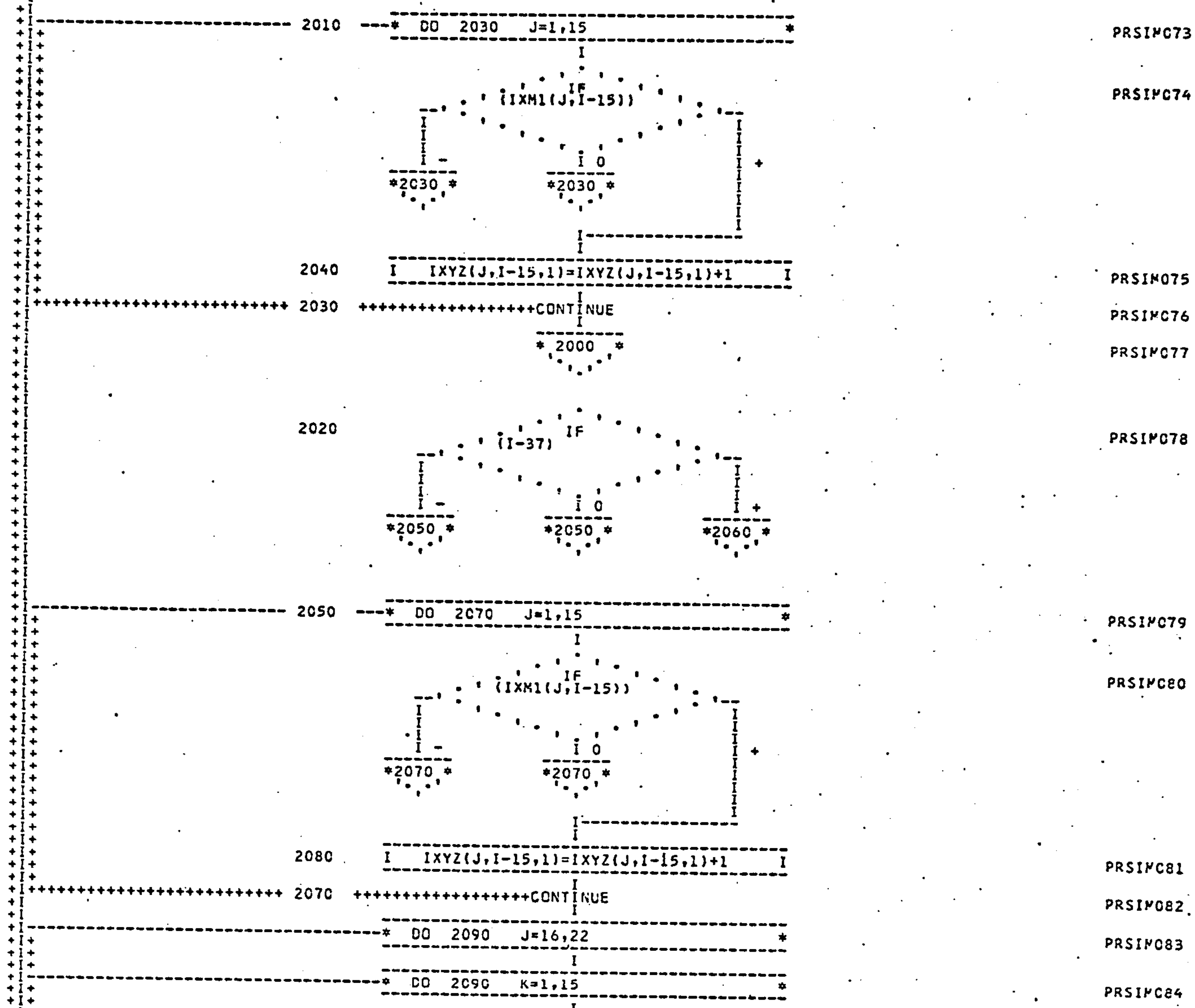
```
C        START RECEIVING RAW MATERIALS
      DO 610 IP=38,45
      IF(42-IP)620,620,630
620   R=ICC(IP-37)
      ICC(IP-37)=DEC(IP)
      GO TO 640
630   R=DEC(IP)
640   IXI(IP)=IXI(IP)+R
610   V(IP)=V(IP)+R*RC(IP-37)
C        MATERIALS HAVE BEEN SETUP FOR NEXT PERIOD
      T=0.
C        INVENTORY TOTALS FOR YEAR AND HOLDING COSTS
      DO 650 IP=1,45
      ITI(IP)=ITI(IP)+IXI(IP)
      TV(IP)=TV(IP)+V(IP)
650   T=T+V(IP)
C        RECORD HOLDING COSTS
      COST(3)=T*XIC
C          CALCULATE ORDERING COSTS
      COST(5)=0.
      DO 651 II=38,45
      IF(DEC(II))651,651,652
652   COST(5)=COST(5)+ISC
651   CONTINUE
C     ADD COSTS FOR YEAR
      T=0.
      DO 660 I=1,5
      COSTY(I)=COSTY(I) + COST(I)
660   T=T+COST(I)
      COSTMY(M)=T
      COSTT=COSTT+T
      IF(M-14)800,920,920
920   JM=M-12
      KM=JM-1
      TCOSTM=0.
      AVARM=0.
      DO 930 JJM=13,M
930   TCOSTM=TCOSTM+COSTMY(JJM)
      ACOSTM=TCOSTM/JM
      DO 940 JJM=13,M
940   AVARM=AVARM+(COSTMY(JJM)-ACOSTM)**2
      IF(KM)950,950,960
950   ASDM=0.
      GO TO 970
960   ASDM=SQRT(AVARM/FLOAT(KM))
970   CONTINUE
C        GET MONTH IN YEAR AND CHECK IF YEAR END
      I=M/12
      I=M-12*I
      IF(I)700,710,700
710   DO 720 IP=1,45
      ITI(IP)=ITI(IP)/12
720   TV(IP)=TV(IP)/12.
C        OUTPUT YEARLY REPORT
      DO 2933 I=1,5
2933  TCOSTY(I)=TCOSTY(I)+COSTY(I)
      TCOSTT=TCOSTT+COSTT
      DO 2935 ID=1,3
      TTSTH(ID)=TTSTH(ID)+TSTH(ID)
      TTOTH(ID)=TTOTH(ID)+TOTH(ID)
```

```
      TTSUH(ID)=TTSUH(ID)+TSUH(ID)
      TTRH(ID)=TTRH(ID)+TRH(ID)
2935  TTIH(ID)=TTIH(ID)+TIH(ID)
      DO 2939 I=1,15
      ITITD(I)=ITITD(I)+ITD(I)
2939  ITITB(I)=ITITB(I)+ITB(I)
700   IF(MMAX-M)731,731,800
731   CONTINUE
      WRITE(6,9040)TCOSTY,TCOSTT
9040  FORMAT(27H1FINAL REPORT FOR 60 MONTHS/1H0,11X,4H HFT,7X,8H PAYROLL
     1,11X,4H INV,5X,10H BACKORDER,6X,9H ORDERING,/,1H ,51X,6H COSTS,/,
     21H 5F15.2,/,11H0TOTAL COST,F16.2//)
      WRITE(6,9013)ITITD
9013  FORMAT(/20H TOTAL DEMAND, UNITS/(1H 8I11))
      WRITE(6,9014)ITITB
9014  FORMAT(/22H TOTAL BACKORDER UNITS/(1H 8I11))
      WRITE(6,9999)ZZ,ZY,ZX,ZW
9999  FORMAT(1H0,4F15.2)
      WRITE(6,9006)TTSTH,TTOTH,TTSUH,TTRH,TTIH
9006  FORMAT(33H0DEPARTMENT OPERATIONS--MAN HOURS/1H010X,8H DEPT. 1,8X,8
     1H DEPT. 2,8X,8H DEPT. 3/1H04H STH,3F15.2/1H 4H OTH,3F15.2/1H 4H SUP
     2H,3F15.2/1H 4H RH ,3F15.2/1H 4H XIH,3F15.2//)
      WRITE(6,9799)ACOSTM,ASOM
9799  FORMAT(/////,2F15.2)
730   CONTINUE
7301  WRITE(6,9016)
9016  FORMAT(///11H END OF RUN)
      STOP
      END
```

```
            DO 8901  I=1,15                                   PRSIM020
8901        IBO(I)=0                                          PRSIM021
      ***   READ INITIAL CONDITIONS                        ***PRSIMC22
            DO 10  I=1,37                                     PRSIMC23
10          READ (5,8000)
            (IXM1(I,J),J=1,30),
            S(I),U(I),IXI(I),V(I),I,
            C(I)                                              PRSIMC24
            DO 2113  I=1,37                                   PRSIMC25
            DO 2113  J=1,30                                   PRSIM026
2113        IXM1(I,J)=IABS(IXM1(I,J))                         PRSIMC27
8000        FORMAT(30I1,2F10.2,I5,F10.2,I5)                ** PRSIMC28
            DO 830  I=38,45                                   PRSIMC29
830         READ (5,8010)
            IXI(I),V(I)                                       PRSIMC30
8010        FORMAT(I5,F10.2)                               ** PRSIMC31
            READ (5,8001)
            IBMR,IBN,IXLW                                     PRSIMC32
8001        FORMAT(9I5)                                    ** PRSIMC33
            READ (5,8002)
            AS,BS,CS                                          PRSIMC34
8002        FORMAT(8F10.2,/,7F10.2)                        ** PRSIMC35
            READ (5,8002)
            XMU,SIG                                           PRSIMC36
            READ (5,8004)
            IDD,RC                                            PRSIM037
8004        FORMAT(8I5/8F10.2)                             ** PRSIMC38
```

```
                                                          PRSIMC39
              READ
             (5,8005)
             ALPHA,BETA
8005    FORMAT(8F10.7/7F10.7)                          ** PRSIMC40
              READ                                        PRSIMC41
             (5,8006)
             WR,XMOT,IHC,IFC,ITC,
             XIC,IBC,ISC,N
8006    FORMAT(2F10.2,3I5,F5.2,3I5)                    ** PRSIMC42
              READ                                        PRSIMC43
             (5,8007)
             STAR,MMAX
8007    FORMAT(F5.0,I5)                                ** PRSIMC44
***       WRITE INITIAL CONDITICNS CN TAPE 2            ***PRSIMC45
              REWIND                                      PRSIMC46
              2
        * DO 90030 I=1,45 *                               PRSIMC47
              IF                                          PRSIMC48
             (I-37)
        -          0          +
                           *90050*
90040         WRITE                                       PRSIMC49
             (2,9017)
             (IXM1(I,J),J=1,30),
             S(I),U(I),IXI(I),V(I),I,
             Q(I)
9017    FORMAT(1X,30I1,2F10.2,I10,F10.2,I5)            ** PRSIMO5C
             * 90030 *                                    PRSIMO51
90050         WRITE                                       PRSIMC52
             (2,9025)
             IXI(I),V(I)
9025    FORMAT(1X,I10,F10.2)                           ** PRSIMC53
90030         CONTINUE                                    PRSIMC54
```

```
              WRITE (2,9018)                                        PRSIMC55
              IBMR,IBN,IXLW
9018   FORMAT(1X,9I10)                                       **     PRSIMC56
              WRITE (2,9019)                                        PRSIMC57
              AS,BS,CS,XMU,SIG
9019   FORMAT(1X,8F10.2/1X,7F10.2)                           **     PRSIMC58
              WRITE (2,9021)                                        PRSIMC59
              IOO,RC
9021   FORMAT(1X,8I10/1X,8F10.2)                             **     PRSIMC60
              WRITE (2,9022)                                        PRSIMC61
              ALPHA,BETA
9022   FORMAT(1X,8F15.7/1X,7F15.7)                           **     PRSIMC62
              WRITE (2,9023)                                        PRSIMC63
              WR,XMOT,IHC,IFC,ITC
              XIC,IBC,ISC,N
9023   FORMAT(1X,2F10.2,3I10,F10.2,3I10)                     **     PRSIMC64
              WRITE (2,9024)                                        PRSIMC65
              STAR,MMAX
9024   FORMAT(1X,F10.2,I10)                                  **     PRSIMC66
       DO 1200   I=1,15                                             PRSIMC67
       DO 1200   J=16,45                                            PRSIMC68
       DO 1200   K=1,4                                              PRSIMC69
1200   IXYZ(I,J-15,K)=0                                             PRSIMC70
       DO 2000   I=16,45                                            PRSIMC71
       IF (I-22)                                                    PRSIMC72
          -  2010      0  2010      +  2020
```


2010
DO 2030 J=1,15
IF (IXM1(J,I-15))
2030
2030
2040
IXYZ(J,I-15,1)=IXYZ(J,I-15,1)+1
2030
CONTINUE
2000
2020
IF (I-37)
2050
2050
2060
2050
DO 2070 J=1,15
IF (IXM1(J,I-15))
2070
2070
2080
IXYZ(J,I-15,1)=IXYZ(J,I-15,1)+1
2070
CONTINUE
DO 2090 J=16,22
DO 2090 K=1,15
PRSIMC73
PRSIMC74
PRSIM075
PRSIMC76
PRSIMC77
PRSIM078
PRSIMC79
PRSIMC80
PRSIMC81
PRSIM082
PRSIM083
PRSIMC84

```
                                     I
                ---------------------------------------
                I   ITEMP=IXM1(K,J-15)*IXM1(J,I-15)   I          PRSIMC85
                ---------------------------------------
                                     I
                                     IF
                              (ITEMP)                            PRSIMC86
                I -                  I 0                 I +
             *2090 *              *2090 *
                                     I-------------------
      2100      ---------------------------------------
                I   IXYZ(K,I-15,2)=IXYZ(K,I-15,2)+1   I          PRSIMC87
                ---------------------------------------
      2090  ++++++++++++++++CONTINUE                              PRSIMC88
                                  * 2000 *                        PRSIMC89

      2060                           IF
                              (I-41)                              PRSIMC9C
                I -                  I 0                 I +
             *2110 *              *2110 *              *2120 *

      2110  ---* DO  2130    J=16,37                   *          PRSIMC91
                                     I
             --* DO  2130    K=1,15                    *          PRSIMC92
                                     I
                I   ITEMP=IXM1(K,J-15)*IXM1(J,I-15)   I          PRSIMC93
                                     I
                                     IF
                              (ITEMP)                            PRSIMC94
                I -                  I 0                 I +
             *2130 *              *2130 *
                                     I-------------------
      2140      I   IXYZ(K,I-15,2)=IXYZ(K,I-15,2)+1   I          PRSIMC95
      2130  ++++++++++++++++CONTINUE                              PRSIMC96
```

```
                              * DO 2150   J=23,37                     *   PRSIM097
                              * DO 2150   K=16,22                     *   PRSIM098
                              * DO 2150   L=1,15                      *   PRSIM099
                              I ITEMP=IXM1(L,K-15)*IXM1(K,J-15)*IX  I   PRSIM100
                              I    M1(J,I-15)                         I
                                        IF
                                     (ITEMP)                              PRSIM101
                              I -          I 0              +
                              *2150 *      *2150 *
                   2160       I  IXYZ(L,I-15,3)=IXYZ(L,I-15,3)+1      I   PRSIM102
+++++++++++++++++ 2150  +++++++++++++++++CONTINUE                         PRSIM103
                                       * 2000 *                           PRSIM104
------------------ 2120 ----* DO 2170   J=16,37                       *   PRSIM105
                              * DO 2170   K=1,15                      *   PRSIM106
                              I  ITEMP=IXM1(K,J-15)*IXM1(J,I-15)      I   PRSIM107
                                        IF
                                     (ITEMP)                              PRSIM108
                              I -          I 0              +
                              *2170 *      *2170 *
                   2180       I  IXYZ(K,I-15,3)=IXYZ(K,I-15,3)+1      I   PRSIM109
+++++++++++++++++ 2170  +++++++++++++++++CONTINUE                         PRSIM110
                              * DO 2170   J=23,37                     *   PRSIM111
```

```
                                            I
                              * DO  2190    K=16,22              *   PRSIM112
                                            I
                              * DO  2190    L=1,15               *   PRSIM113
                                            I
                              I  ITEMP=IXM1(L,K-15)*IXM1(K,J-15)*IX  I   PRSIM114
                              I     M1(J,I-15)                        I
                                            I
                                       IF (ITEMP)                        PRSIM115
                                -               0              +
                              *2190 *        *2190 *
                                            I
                 2200         I  IXYZ(L,I-15,4)=IXYZ(L,I-15,4)+1  I   PRSIM116
                 2190         +++++++++++++++++CONTINUE               PRSIM117
                 2000         +++++++++++++++++CONTINUE               PRSIM118
                              * DO  730    IZZ=25,25,5           *   PRSIM119
                                            I
                              I  ZZ=IZZ/10.                      I   PRSIM120
                                            I
                              * DO  730    IZY=15,15,5           *   PRSIM121
                                            I
                              I  ZY=IZY/10.                      I   PRSIM122
                                            I
                              * DO  730    IZX=25,25,5           *   PRSIM123
                                            I
                              I  ZX=IZX/10.                      I   PRSIM124
                                            I
                              * DO  730    IZW=40,40,5           *   PRSIM125
                                            I
                              I  ZW=IZW/10.                      I   PRSIM126
                              I  M=0                             I   PRSIM127
                                            I
              ***   READ INITIAL CONDITIONS FROM TAPE 2            ***PRSIM128
                                            I
                                     2  REWIND                       PRSIM129
                                            I
                              * DO  90010  I=1,37                *   PRSIM130
                                            I
```

```
90010      READ (2,9017) ((XM1(I,J),J=1,30),        PRSIM131
           S(I),U(I),IXI(I),V(I),I=
           Q(I)
           DO 90020 I=38,45                         PRSIM132
90020      READ (2,9025) IXI(I),V(I)                PRSIM133
           READ (2,9018) IBMR,IBN,IXLW              PRSIM134
           READ (2,9019) AS,BS,CS,XMU,SIG           PRSIM135
           READ (2,9021) IOO,RC                     PRSIM136
           READ (2,9022) ALPHA,BETA                 PRSIM137
           READ (2,9023) NR,XMOT,IHC,IFC,ITC,       PRSIM138
           XIC,IBC,ISC,N
           READ (2,9024) STAR,MMAX                  PRSIM139
800        M=M+1                                    PRSIM140
           IF (M-13)                                PRSIM141
             -  *1337*
             0  *1338*
             +  *1337*
1338       DO 1339 I=1,5                            PRSIM142
1339       TCOSTY(I)=0.                             PRSIM143
           TCOSTT=0.                                PRSIM144
```

```
                                        I
        -------*  DO  1336   ID=1,3                     *        PRSIM145
                                        I
                 I  TTSTM(ID)=0.                        I        PRSIM146
                 I  TTOTM(ID)=0.                        I        PRSIM147
                 I  TTSUM(ID)=0.                        I        PRSIM148
                 I  TTRM(ID)=0.                         I        PRSIM149
                                        I
        ++++++ 1336   +++I  TTIM(ID)=0.                 I        PRSIM150
                                        I
        -------*  DO  2938   I=1,15                     *        PRSIM151
                                        I
                 I  ITITD(I)=0                          I        PRSIM152
                                        I
        ++++++ 2938   +++I  ITITB(I)=0                  I        PRSIM153
                                        I
               1337               CONTINUE                       PRSIM154
                                        I
                          (M-1)        IF                        PRSIM155
                    -                   0                  +
                 *100 *              *100 *             *110 *

        ***      CONSTANTS FOR RANDOM EFFECT GENERATION       ***PRSIM156

        ------ 100    ---*  DO  40      I=1,15                  *        PRSIM157
                                        I
                 I  A(I)=2.*SIG(I)*SQRT(3./FLOAT(N))    I        PRSIM158
                                        I
        ++++++ 40     +++I  B(I)=FLOAT(N)/2.            I        PRSIM159
                                        I
               110       I  I=M/12                      I        PRSIM160
                         I  I=M-12*I-1                  I        PRSIM161
                                        I
                          (I)          IF                        PRSIM162
                    -                   0                  +
                 *750 *              *760 *             *750 *

        ***      INITIALIZE TO ZERO FOR START OF YEAR         ***PRSIM163
```

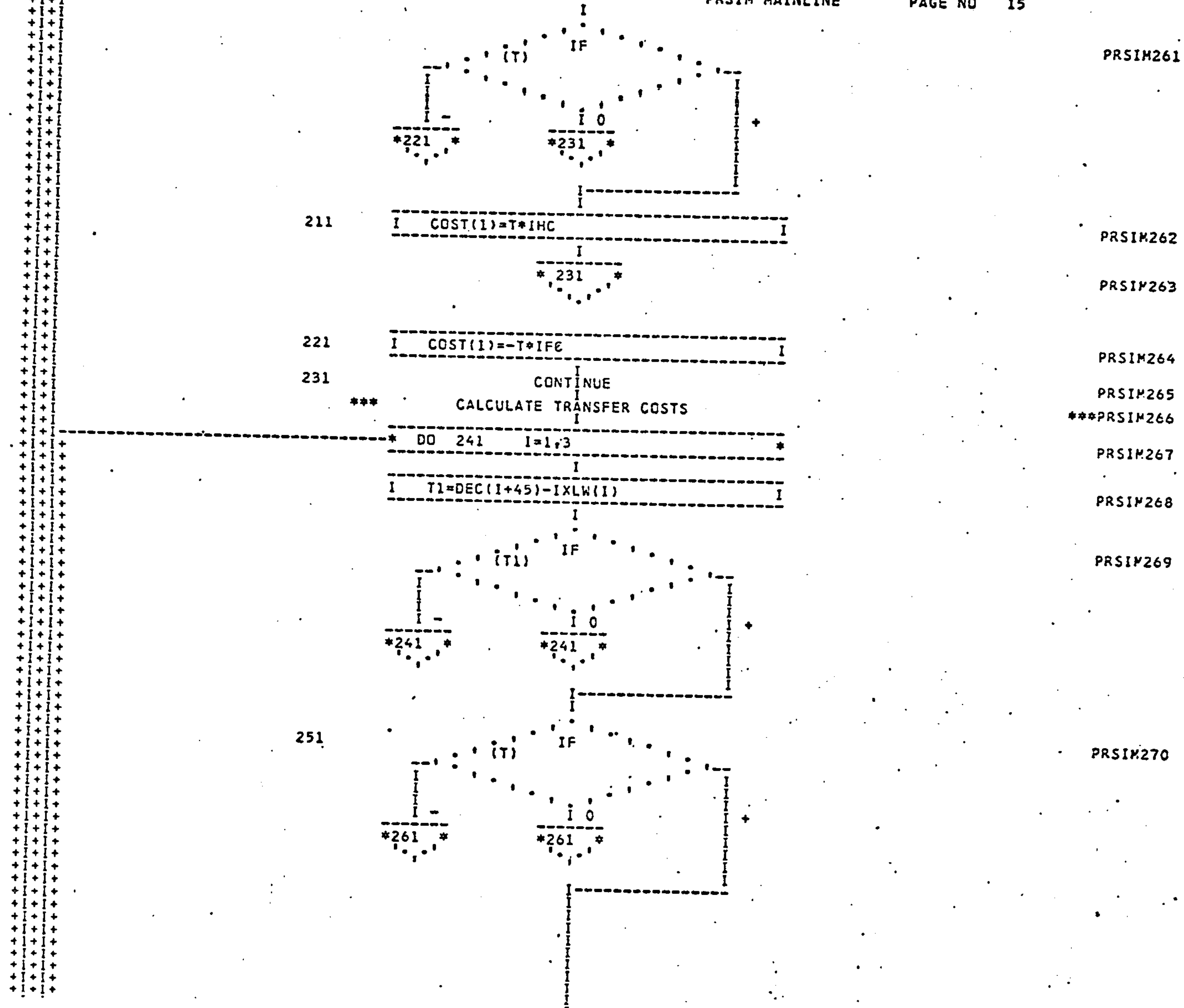
```
 760    DO 8902  I=1,45                                    PRSIM164
          ITI(I)=0                                         PRSIM165
8902      TV(I)=0.                                         PRSIM166
        DO 8903  I=1,3                                     PRSIM167
          TSTH(I)=0.                                       PRSIM168
          TOTH(I)=0.                                       PRSIM169
          TSCH(I)=0.                                       PRSIM170
          TRH(I)=0.                                        PRSIM171
8903      TIH(I)=0.                                        PRSIM172
        DO 8904  I=1,15                                    PRSIM173
          ITD(I)=0                                         PRSIM174
8904      ITB(I)=0                                         PRSIM175
        DO 8905  I=1,5                                     PRSIM176
8905      COSTY(I)=C.                                      PRSIM177
          COSTT=C.                                         PRSIM178
***     START OF MONTHLY LOOP                           ***PRSIM179
 750    CONTINUE                                           PRSIM180
***     18 MONTHS OF FORECASTS FOR EACH ITEM IN UNITS   ***PRSIM181
        DO 50  J=1,18                                      PRSIM182
          XM=M-1+J                                         PRSIM183
          T=SIN(3.14159*(XM/6.))                           PRSIM184
        DO 60  I=1,15                                      PRSIM185
          JFOR(I,J)=AS(I)+BS(I)*XM+CS(I)*T+0               PRSIM186
           .5
```

```
                IF (JFOR(I,J))                                PRSIM187
        -               0               +
       *70             *60             *60

   70   I  JFOR(I,J)=0                                  I     PRSIM188
   60   +++++++++++++++++CONTINUE                             PRSIM189
   50   +++++++++++++++++CONTINUE                             PRSIM190
 ***        AVERAGE MAN-HOUR CALCULATIONS FOR FGP'S        ***PRSIM191
        * DO  200    I=1,15                             *     PRSIM192
        I  FGP(I)=U(I)+S(I)/IC(I)                       I     PRSIM193
        * DO  210    J=16,37                            *     PRSIM194
                IF (IXM1(I,J-15))                             PRSIM195
        -               0               +
      *210            *210
  220   I  FGP(I)=FGP(I)+U(J)+S(J)/IC(J)                I     PRSIM196
                IF (J-23)                                     PRSIM197
        -               0               +
      *230            *210            *210
  230 --* DO  240    K=23,37                            *     PRSIM198
```

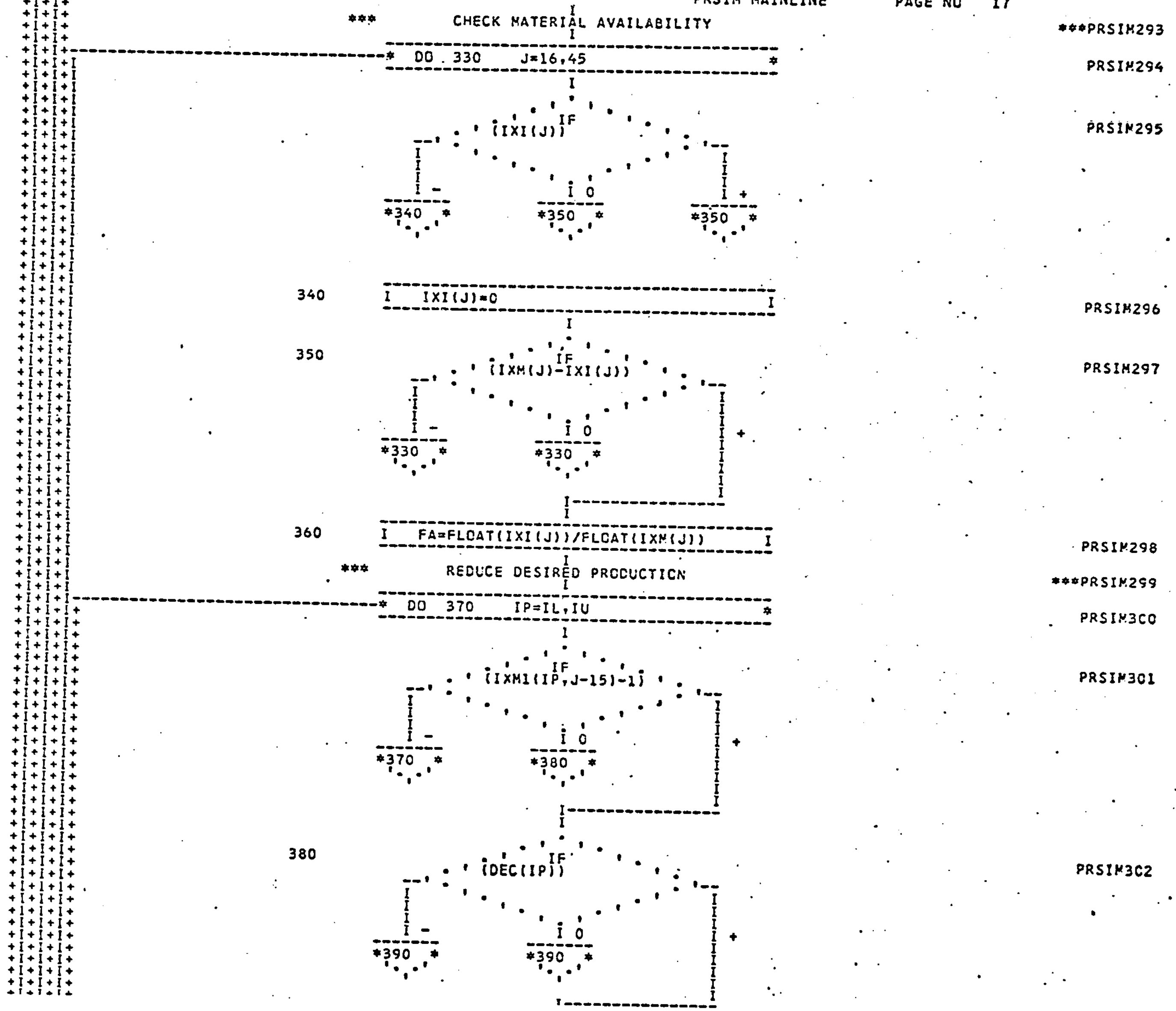
```
                 IF
           (IXM1(J,K-15))                                      PRSIM199
      -          0          +
    *240*      *240*

250     I  FGP(I)=FGP(I)+U(K)+S(K)/IC(K)  I                    PRSIM200
240     +++++++++++++++++CONTINUE                              PRSIM201
210     +++++++++++++++++CONTINUE                              PRSIM202
200     +++++++++++++++++CONTINUE                              PRSIM203
***          12 MONTHS FORECAST IN AGGREGATE MANHOURS       ***PRSIM204
        *  DO  90     IT=1,12                    *             PRSIM205
        I  SHAT(IT)=0.                           I             PRSIM206
        I  T=M+IT-1                              I             PRSIM207
        *  DO  111    I=1,15                     *             PRSIM208
        I  SHAT(IT)=SHAT(IT)+FGP(I)*(AS(I)+BS    I             PRSIM209
        I      (I)*T+CS(I)*SIN(3.14159*T/6.)     I
        I      )                                 I
                 IF
              (SHAT(IT))                                       PRSIM210
      -          0          +
    *120*      *111*      *111*

120     I  SHAT(IT)=0.                           I             PRSIM211
111     +++++++++++++++++CONTINUE                              PRSIM212
90      +++++++++++++++++CONTINUE                              PRSIM213
***     GET WORKFORCE AND INVENTORY AND APPLY LINEAR DECISION RULES.  ***PRSIM214
        I  XILST=0.                              I             PRSIM215
        I  IWLST=0                               I             PRSIM216
        *  DO  125    I=1,3                      *             PRSIM217
```

```
125   IWLST=IWLST+IXLW(I)                                        PRSIM218
      DO 130   I=1,15                                            PRSIM219
130   XILST=XILST+FGP(I)*(IXI(I)-IBG(I))                         PRSIM220
      PSTAR=ALPHA(13)+ALPHA(14)*IWLST+AL                         PRSIM221
         PHA(15)*XILST
      WSTAR=BETA(13)+BETA(14)*IWLST+BETA                         PRSIM222
         (15)*XILST+0.5
      DO 140   IT=1,12                                           PRSIM223
      PSTAR=PSTAR+ALPHA(IT)*SHAT(IT)                             PRSIM224
140   WSTAR=WSTAR+BETA(IT)*SHAT(IT)+0.5                          PRSIM225
***   CHECK FOR NEGATIVE PSTAR AND WSTAR - SET TO ZERO IF THEY ARE ***PRSIM226
      IF (PSTAR)   - 150   0 160   + 160                         PRSIM227
150   WRITE (6,9000) PSTAR                                       PRSIM228
9000  FORMAT(1H0,10X,17HPSTAR IS NEGATIVE,F15.4)              ** PRSIM229
      PSTAR=0.                                                   PRSIM230
160   IF (WSTAR)   - 170   0 180   + 180                         PRSIM231
170   WRITE (6,9001) WSTAR                                       PRSIM232
9001  FORMAT(1H0,10X,17HWSTAR IS NEGATIVE,I10)                ** PRSIM233
```

```
                                   I
                 -------------------------------------------
                 I   WSTAR=0                               I       PRSIM234
                 -------------------------------------------
                                   I
      180                     CONTINUE                             PRSIM235
                                   I
                 -------------------------------------------
                 I   TMON=M                                I       PRSIM236
                 -------------------------------------------
                                   I
      103     ---* DO  122     I=1,15                      *       PRSIM237
                 -------------------------------------------
                                   I
                 * DO  122     MA=1,18                     *       PRSIM238
                 -------------------------------------------
                                   I
      122     +++I  JSE(I,MA)=SIG(I)*JFGR(I,MA)+0.5        I       PRSIM239
                 -------------------------------------------
             ***      CALL DECIDE TO MAKE PERICD DECISIONS      ***PRSIM240
                                   I
                      CALL DECIDE                                  PRSIM241
                                   I
             ***      CHECK DECISIONS AND SET TO ZERG IF NEGATIVE ***PRSIM242
                                   I
                 * DO  190     I=1,48                      *       PRSIM243
                 -------------------------------------------
                                   I
                               IF (DEC(I))                         PRSIM244
                 I -               I 0                 I +
               *201 *            *190 *              *190 *

      201        I   DEC(I)=0                              I       PRSIM247
                 -------------------------------------------
      190     ++++++++++++++++++CONTINUE                           PRSIM248
                                   I
                 * DO  8906    I=1,4                       *       PRSIM255
                 -------------------------------------------
                                   I
      8906    +++I  COST(I)=0.                             I       PRSIM256
                 -------------------------------------------
             ***      CALCULATE HIRING AND FIRING COSTS         ***PRSIM257
                                   I
                 I   T1=DEC(46)+DEC(47)+DEC(48)            I       PRSIM258
                 I   T2=IXLW(1)+IXLW(2)+IXLW(3)            I       PRSIM259
                 I   T=T1-T2                               I       PRSIM260
                 -------------------------------------------
                                   I
```


IF
(T)
*221
*231
PRSIM261
211
COST(1)=T*IHC
PRSIM262
*231
PRSIM263
221
COST(1)=-T*IFE
PRSIM264
231
CONTINUE
PRSIM265
***
CALCULATE TRANSFER COSTS
***PRSIM266
DO 241 I=1,3
PRSIM267
T1=DEC(I+45)-IXLW(I)
PRSIM268
IF
(T1)
*241
*241
PRSIM269
251
IF
(T)
*261
*261
PRSIM270

```
271        IF (T-T1)                                   PRSIM271
             -  -> 281
             0  -> 281
             +  -> 291

281        T1=T1-T                                     PRSIM272
           T=0.                                        PRSIM273

261        COST(1)=COST(1)+T1*ITC                      PRSIM274

           -> 241                                      PRSIM275

291        T=T-T1                                      PRSIM276

241        IXLW(I)=DEC(I+45)                           PRSIM277

***        START DEPARTMENT LOOP                    ***PRSIM278

           DO 300   ID=1,3                             PRSIM279

***        OPERATE DEPARTMENTS                      ***PRSIM280
***        ZERO MATERIAL USEAGE                     ***PRSIM281

           DO 1    I=1,45                              PRSIM282

1          IXM(I)=0                                    PRSIM283

***        GET MATERIAL LOWER AND UPPER LIMITS FOR THIS DEPARTMENT   ***PRSIM284

           IL=IBMR(ID)+1                               PRSIM285
           IU=IBMR(ID)+IBN(ID)                         PRSIM286

***        ADD MATERIAL REQUIREMENTS                ***PRSIM287

           DO 310   IP=IL,IU                           PRSIM288

           JZ=IU+1                                     PRSIM289

           DO 320   IZ=JZ,45                           PRSIM290

320        IXM(IZ)=IXM(IZ)+IXM1(IP,IZ-15)*DEC          PRSIM291
           (IP)

310        CONTINUE                                    PRSIM292
```


*** CHECK MATERIAL AVAILABILITY ***PRSIM293
DO 330 J=16,45 PRSIM294
IF (IXI(J)) PRSIM295
*340
*350
*350
340 IXI(J)=0 PRSIM296
350 IF (IXM(J)-IXI(J)) PRSIM297
*330
*330
360 FA=FLOAT(IXI(J))/FLOAT(IXM(J)) PRSIM298
*** REDUCE DESIRED PRODUCTION ***PRSIM299
DO 370 IP=IL,IU PRSIM300
IF (IXM1(IP,J-15)-1) PRSIM301
*370
*380
380 IF (DEC(IP)) PRSIM302
*390
*390

```
400        WRITE (6,9003) IP,J                                                PRSIM303
9003       FORMAT(1HO,13HPRODUCTICN OF,I5,25H REDUCED BECAUSE MATERIAL,I5,10H**   PRSIM304
              ** WAS SHORT)                                                   **
390        DO  410     IZ=16,45                                               PRSIM306
410        IXM(IZ)=IXM(IZ)+IXM1(IP,IZ-15)*DEC                                 PRSIM307
             (IP)*(FA-1.)+0.5
           DEC(IP)=IFIX(FA*DEC(IP)+0.5)                                       PRSIM308
370        CONTINUE                                                           PRSIM309
330        CONTINUE                                                           PRSIM310
***        END MATERIAL AVAILABILITY CHECK                                 ***PRSIM311
***        ZERO MANHOUR REQUIREMENTS                                       ***PRSIM312
           T=0.                                                               PRSIM313
***        GET MANHOUR REQUIREMENTS                                        ***PRSIM314
           DO  420     IP=IL,IU                                               PRSIM315
           IF (DEC(IP))                                                       PRSIM316
             -  *420
             0  *420
             +
430        T=T+S(IP)+DEC(IP)*U(IP)                                            PRSIM317
420        CONTINUE                                                           PRSIM318
***        NOW HAVE MANHOUR REQUIREMENTS                                   ***PRSIM319
***        STRAIGHT TIME MAN HOURS                                         ***PRSIM320
           ST=160.*DEC(ID+45)                                                 PRSIM321
***        CHECK OVERTIME AND CALCULATE OVERTIME AND IDLETIME              ***PRSIM322
           IF (T-ST)                                                          PRSIM323
             -  *440
             0  *440
             +  *450
```

```
 440   ---------------------------------------------         PRSIM324
       I   OTH(ID)=0.                              I         PRSIM325
       I   XIH(ID)=ST-T                            I         PRSIM326
       I   FB=1.                                   I
       ---------------------------------------------
                          I
                     ----------
                     * 460    *                              PRSIM327
                      *.....*

 450                     IF                                  PRSIM328
                   (ST)
          I                I                 I
          I -              I 0               I +
       *470  *         *470  *           *480  *

 470   ---------------------------------------------         PRSIM329
       I   FB=0.                                   I         PRSIM330
       I   T=0.                                    I
       ---------------------------------------------
                          I
                     * 490    *                              PRSIM331

 480   ---------------------------------------------         PRSIM332
       I   FB=T/ST                                 I
       ---------------------------------------------
                          I
                         IF                                  PRSIM333
                   (FB-XMOT)
          I                I                 I
          I -              I 0               I +
       *485  *         *485  *           *489  *

 485   ---------------------------------------------         PRSIM334
       I   FB=1.                                   I
       ---------------------------------------------
                          I
                     * 490    *                              PRSIM335

 ***          GET FACTOR TO REDUCE PRODUCTION             ***PRSIM336

 489   ---------------------------------------------         PRSIM337
       I   FB=XMOT/FB                              I         PRSIM338
       I   T=XMOT*ST                               I
       ---------------------------------------------
                          I
                     +++++++++++
                      WRITE                                  PRSIM339
                     (6,9004)
                   ID
                   +++++++++++++++++++++
                          I
 9004  FORMAT(1HC,24HPRODUCTION IN DEPARTMENT,I3,43H WAS REDUCED BECAUSE **   PRSIM340
```

```
490      I  OTH(ID)=T-ST                         I        PRSIM342
         I  XIH(ID)=0.                           I        PRSIM343

460      I  TR=0.                                I        PRSIM344
         I  TS=0.                                I        PRSIM345

     ***      OVERTIME FUDGE FACTCR                    ***PRSIM346

                    IF                                    PRSIM347
              (ST+OTH(ID))
         -            0            +
       *461 *      *461 *       *462 *

461      I  G=0.                                 I        PRSIM348

                  * 463 *                                 PRSIM349

462                CONTINUE                               PRSIM350

         I  G=((ST+1.5*OTH(ID))/(ST+OTH(ID)))*   I        PRSIM351
         I      WR                               I

463                CONTINUE                               PRSIM352
     ***   REDUCE PRODUCTION IF NECESSARY AND ADD SETUP AND RUN TIMES   ***PRSIM353

         *  DO  500     IP=IL,IU                 *        PRSIM354

         I  DEC(IP)=IFIX(FB*DEC(IP)+0.5)         I        PRSIM355
         I  AP(IP)=DEC(IP)                       I        PRSIM356

                    IF                                    PRSIM357
                (DEC(IP))
         -            0            +
       *500 *      *500 *

510      I  TS=TS+S(IP)                          I        PRSIM358
         I  TR=TR+DEC(IP)*U(IP)                  I        PRSIM359

     ***      ADD IN LABCUR COSTS                      ***PRSIM360
```

```
          T=(S(IP)+DEC(IP)*U(IP))*G                          PRSIM361
          DO 520     IZ=1,30                                 PRSIM362
          IF (IXM1(IP,IZ)-1)                                 PRSIM363
             -  *520*      0  *530*      +
530       Z=V(IZ+15)/IXI(IZ+15)                              PRSIM364
***       VALUE ADDED FROM INVNETORY                      ***PRSIM365
          T=T+DEC(IP)*IXM1(IP,IZ)*Z                          PRSIM366
          IXI(IZ+15)=IXI(IZ+15)-DEC(IP)*IXM1                 PRSIM367
             (IP,IZ)
          V(IZ+15)=IXI(IZ+15)*Z                              PRSIM368
520       CONTINUE                                           PRSIM369
***       ADD OUTPUT INVNETORY UNITS AND VALUE            ***PRSIM370
          IXI(IP)=IXI(IP)+DEC(IP)                            PRSIM371
          V(IP)=V(IP)+T                                      PRSIM372
500       CONTINUE                                           PRSIM373
***       END OF PRODUCTION                               ***PRSIM374
***       RECORD STRAIGHT TIME HOURS                      ***PRSIM375
          STH(ID)=ST                                         PRSIM376
          SUH(ID)=TS                                         PRSIM377
          RH(ID)=TR                                          PRSIM378
***       ADD FOR YEARLY REPORT                           ***PRSIM379
          TSTH(ID)=ST+TSTH(ID)                               PRSIM380
          TOTH(ID)=OTH(ID)+TOTH(ID)                          PRSIM381
          TSUH(ID)=TS+TSUH(ID)                               PRSIM382
          TRH(ID)=TR+TRH(ID)                                 PRSIM383
          TIH(ID)=XIH(ID)+TIH(ID)                            PRSIM384
***       PAYROLL COSTS                                   ***PRSIM385
          COST(2)=COST(2)+(ST+1.5*OTH(ID))*W                 PRSIM386
                                            R
300       CONTINUE                                           PRSIM387
***       END OF DEPARTMENT LOOP                          ***PRSIM388
***       START DEMAND AND SALES CALCULATIONS             ***PRSIM389
          DO 540     I=1,15                                  PRSIM390
```

```
                                                  I
                                   .' CALL RANSET(STA  '.                 PRSIM392
                                      R)
                                                  I
                              I  T=0.                              I      PRSIM393
                                                  I
 ---------------------------* DO 550    J=1,N                      *      PRSIM394
                                                  I
 ++++++++++++++++++++ 550  +++I  T=T+RANF(-1)                      I      PRSIM395
                                                  I
 ++++++++++++++++++++ 540  +++I  E(I)=A(I)*(T-B(I))+XMU(I)         I      PRSIM396
                                                  I
                                   .' CALL RANGET(STA  '.                 PRSIM397
                                      R)
                                                  I
                          ***      DEMAND BY ITEM                         ***PRSIM398
                                                  I
 ---------------------------* DO 560    I=1,15                     *      PRSIM399
                                                  I
                              I  ISFG(I)=JFOR(I,1)*(E(I)+1.)+0.5   I      PRSIM4CC
                                                  I
 ++++++++++++++++++++ 560  +++I  SA(I,M)=ISFG(I)                   I      PRSIM4C1
                                                  I
                          ***      FINISHED GOODS PRODUCT ITERATION       ***PRSIM402
                                                  I
 ---------------------------* DO 910    I=1,15                     *      PRSIM403
                                                  I
                          ***      DEMAND INCLUDES BACKLOG                ***PRSIM404
                                                  I
                              I  JD=ISFG(I)+IBO(I)                 I      PRSIM405
                                                  I
                          ***      CHECK FOR ENOUGH INVENTORY             ***PRSIM4C6
                                                  I
                                                 IF                       PRSIM407
                                            (IXI(I)-JD)
                              I                                    I
                              I -                 I 0              I +
                            *580 *              *590 *           *590 *

                       580    I  IUS(I)=IXI(I)                     I      PRSIM4C8
                              I  IBO(I)=JD-IXI(I)                  I      PRSIM4C9
                                                  I
                                               * 600 *                    PRSIM410
```

```
 590   I  IUS(I)=JD                                   I        PRSIM411
       I  IBO(I)=0                                    I        PRSIM412

 600          IF (IXI(I))                                      PRSIM413
        -  *601 *        0  *601 *        +  *602 *

 601   I  Z=0.                                        I        PRSIM414
              * 603 *                                          PRSIM415

 602   I  Z=V(I)/IXI(I)                               I        PRSIM416
 603              CONTINUE                                     PRSIM417
  ***      REDUCE UNITS AND VALUE ADDED BY AMOUNT SHIPPED   ***PRSIM418
       I  V(I)=V(I)-IUS(I)*Z                          I        PRSIM419
       I  IXI(I)=IXI(I)-IUS(I)                        I        PRSIM420
  ***     TOTALS FOR YEAR                                   ***PRSIM421
       I  ITD(I)=ITD(I)+ISFG(I)                       I        PRSIM422
       I  ITB(I)=ITB(I)+IBO(I)                        I        PRSIM423
  ***      BACKORDER COST                                   ***PRSIM424
       I  COST(4)=COST(4)+IBC*FLOAT(IBO(I))           I        PRSIM425
 910  +++++++++++++++++CONTINUE                                PRSIM426
  ***      END OF FINISHED GOODS LOOP                       ***PRSIM427
  ***      START RECEIVING RAW MATERIALS                    ***PRSIM428
      * DO  610    IP=38,45                           *        PRSIM429

              IF (42-IP)                                       PRSIM430
        -  *620 *        0  *620 *        +  *630 *
```

```
620       R=IOO(IP-37)                                        PRSIM431
          IOO(IP-37)=DEC(IP)                                  PRSIM432
               * 640 *                                        PRSIM433
630       R=DEC(IP)                                           PRSIM434
640       IXI(IP)=IXI(IP)+R                                   PRSIM435
610   +++ V(IP)=V(IP)+R*RC(IP-37)                             PRSIM436
***       MATERIALS HAVE BEEN SETUP FOR NEXT PERIOD        ***PRSIM437
          T=C.                                                PRSIM438
***       INVENTORY TOTALS FOR YEAR AND HOLDING COSTS      ***PRSIM439
          DO 650    IP=1,45                                   PRSIM440
          ITI(IP)=ITI(IP)+IXI(IP)                             PRSIM441
          TV(IP)=TV(IP)+V(IP)                                 PRSIM442
650   +++ T=T+V(IP)                                           PRSIM443
***       RECORD HOLDING COSTS                             ***PRSIM444
          COST(3)=T*XIC                                       PRSIM445
***       CALCULATE ORDERING COSTS                         ***PRSIM446
          COST(5)=0.                                          PRSIM447
          DO 651    II=38,45                                  PRSIM448
               IF (DEC(II))                                   PRSIM449
          -              0              +
        *651 *         *651 *
652       COST(5)=COST(5)+ISC                                 PRSIM45C
651       ++++++++++++++++CONTINUE                            PRSIM451
***       ADD COSTS FOR YEAR                               ***PRSIM452
```

```
                                   T=0.                                        PRSIM453
        ---------------------* DO 660    I=1,5                    *            PRSIM454
                               COSTY(I)=COSTY(I)+COST(I)                       PRSIM455
        ++++++++++++++ 660 +++ T=T+COST(I)                                     PRSIM456
                               COSTMY(M)=T                                     PRSIM457
                               COSTT=COSTT+T                                   PRSIM458

                               IF (M-14)                                       PRSIM485
                        -            0             +
                      *800*        *920*

                   920         JM=M-12                                         PRSIM486
                               KM=JM-1                                         PRSIM487
                               TCCSTM=0.                                       PRSIM488
                               AVARM=0.                                        PRSIM489
        ---------------------* DO 930    JJM=13,M                 *            PRSIM490
        ++++++++++++++ 930 +++ TCOSTM=TCOSTM+COSTMY(JJM)                       PRSIM491
                               ACCSTM=TCOSTM/JM                                PRSIM492
        ---------------------* DO 940    JJM=13,M                 *            PRSIM493
        ++++++++++++++ 940 +++ AVARM=AVARM+(COSTMY(JJM)-ACOSTM)**2             PRSIM494

                               IF (KM)                                         PRSIM495
                        -            0             +
                      *950*        *950*         *960*

                   950         ASDM=C.                                         PRSIM496
                                   * 970 *                                     PRSIM497
```

```
        960          I  ASDM=SQRT(AVARM/FLOAT(KM))          I      PRSIM498
                                     I
        970                      CONTINUE
                                     I
            ***      GET MONTH IN YEAR AND CHECK IF YEAR END       ***PRSIM501
                                     I
                     I  I=M/12                               I      PRSIM502
                     I  I=M-12*I                             I      PRSIM503
                                     I
                                    IF                              PRSIM504
                               (I)
                     I -             I 0              I +
                   *700 *          *710 *           *700 *

---------- 710   ---*  DO  720    IP=1,45                    *      PRSIM505
                                     I
                     I  ITI(IP)=ITI(IP)/12                   I      PRSIM506
                                     I
++++++++++ 720   +++I  TV(IP)=TV(IP)/12.                     I      PRSIM507
                                     I
            ***        OUTPUT YEARLY REPORT                       ***PRSIM508
                                     I
-----------------*  DO  2933    I=1,5                        *      PRSIM522
                                     I
++++++++++ 2933  +++I  TCOSTY(I)=TCOSTY(I)+COSTY(I)          I      PRSIM523
                    I  TCOSTT=TCOSTT+CCSTT                   I      PRSIM524
                                     I
-----------------*  DO  2935    ID=1,3                       *      PRSIM525
                                     I
                    I  TTSTH(ID)=TTSTH(ID)+TSTH(ID)          I      PRSIM526
                    I  TTCTH(ID)=TTCTH(ID)+TCTH(ID)          I      PRSIM527
                    I  TTSUH(ID)=TTSUH(ID)+TSUH(ID)          I      PRSIM528
                    I  TTRH(ID)=TTRH(ID)+TRH(ID)             I      PRSIM529
                                     I
++++++++++ 2935  +++I  TTIH(ID)=TTIH(ID)+TIH(ID)             I      PRSIM530
                                     I
-----------------*  DO  2939    I=1,15                       *      PRSIM531
                                     I
                    I  ITITD(I)=ITITD(I)+ITD(I)              I      PRSIM532
                                     I
++++++++++ 2939  +++I  ITITB(I)=ITITB(I)+ITB(I)              I      PRSIM533
                                     I
```

```
700          IF (MMAX-M)                                                  PRSIM534
             -  *731      0  *731      +  *800

731          CONTINUE                                                     PRSIM535

             WRITE (6,9040) TCOSTY,TCOSTT                                 PRSIM536

9040         FORMAT(27H1FINAL REPORT FOR 60 MONTHS/1H0,11X,4H HFT,7X,8H PAYROLL**  PRSIM537
            **,11X,4H INV,5X,10H BACKORDER,6X,9H ORDERING,/,1H ,51X,6H COSTS,/,   **
            **1H 5F15.2,/,11HOTOTAL COST,F16.2//)                                 **

             WRITE (6,9013) ITITD                                         PRSIM540

9013         FORMAT(/20H TOTAL DEMAND, UNITS/(1H 8I11))          **       PRSIM514

             WRITE (6,9014) ITITB                                         PRSIM541

9014         FORMAT(/22H TOTAL BACKORDER UNITS/(1H 8I11))        **       PRSIM516

             WRITE (6,9999) ZZ,ZY,ZX,ZW                                   PRSIM542

9999         FORMAT(1H0,4F15.2)                                  **       PRSIM543

             WRITE (6,9006) TTSTH,TTCTH,TTSUH,TTRH,TTIH                   PRSIM544

9006         FORMAT(33HODEPARTMENT OPERATIONS--MAN HOURS/1H010X,8H DEPT. 1,8X,8**  PRSIM518
            **H DEPT. 2,8X,8H DEPT. 3/1H04H STH,3F15.2/1H 4H OTH,3F15.2/1H 4H SU**
            **H,3F15.2/1H 4H RH ,3F15.2/1H 4H XIH,3F15.2//)                       **

             WRITE (6,9799) ACOSTM,ASDM                                   PRSIM499

9799         FORMAT(/////,2F15.2)                                **       PRSIM500
730          CONTINUE                                                     PRSIM545

7301         WRITE (6,9016)                                               PRSIM546

9016         FORMAT(///11H END OF RUN)                           **       PRSIM547

             STOP                                                         PRSIM548
```

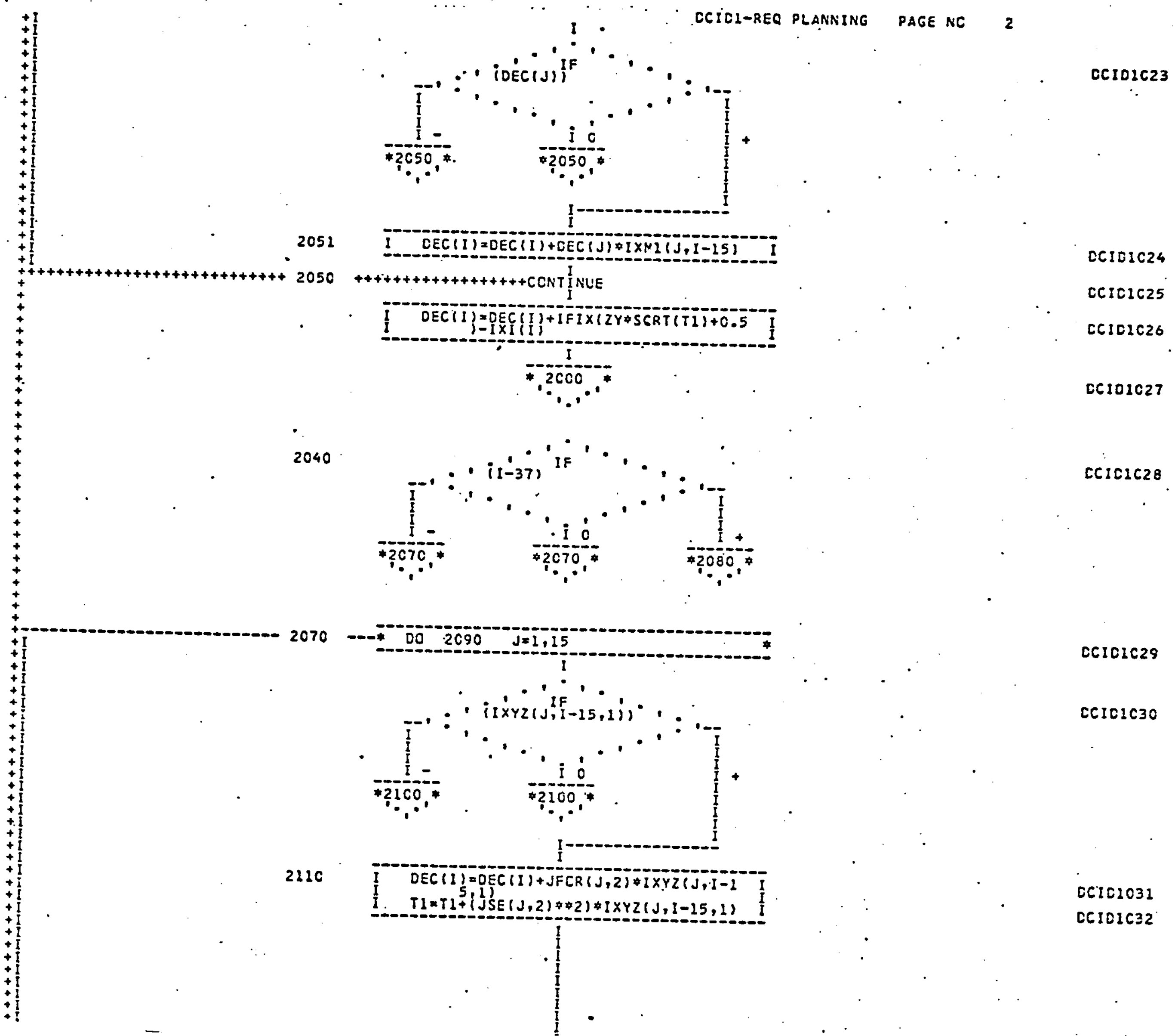
APPENDIX IV

THE REQUIREMENTS PLANNING LOT-FOR-LOT MODEL

```
      SUBROUTINE DECIDE
      INTEGER WSTAR
      COMMON IXM1(37,30),S(37),U(37),IXI(45),V(45),STH(3),DTH(3),SUH(3),
     1XIH(3),IBMR(3),IEN(3),IXLW(3),ISFG(15),IUS(15),IBO(15),ISL(15),
     2ITD(15),ITB(15),ITSL(15),AS(15),BS(15),CS(15),IFOR(15,6),CCSTY(5),
     3COST(5),COSTT,SPAT(12),ALPHA(15),BETA(15),XMU(15),SIG(15),FGP(15),
     4XMUX(3),SIGX(3),ICO(8),KC(8),E(15),A(15),B(15),IXM(45),DEC(48),
     5AP(45),ITI(45),TV(45),TSTH(3),TOTH(3),TSUH(3),TRH(3),RH(3),IC(45)
      COMMON STAR,WSTAR,PSTAR,WR,XMOT,IHC,ITC,IFC,XIC,IBC,ISC,N,MMAX,M
      COMMON COSTM,TIH(3),JFOR(45,18),JSE(45,18),ZZ,ZY,ZX,ZW
      COMMON IXYZ(15,30,4)
      COMMON IDEC(15,6),ICQ(15,6)
C        DECIDE 1 - REQUIREMENTS PLANNING
      DO 2000 I=1,45
      DEC(I)=0.
      T1=0.
      IF(I-15)2010,2010,2020
 2010 DEC(I)=JFOR(I,1)+IFIX(ZZ*JSE(I,1)+0.5)+IBO(I)-IXI(I)
      GO TO 2000
 2020 IF(I-22)2030,2030,2040
 2030 DO 2050 J=1,15
      IF(IXYZ(J,I-15,1))2050,2050,2060
 2060 DEC(I)=DEC(I)+JFOR(J,2)*IXYZ(J,I-15,1)
      T1=T1+(JSE(J,2)**2)*IXYZ(J,I-15,1)
      IF(DEC(J))2050,2050,2051
 2051 DEC(I)=DEC(I)+DEC(J)*IXM1(J,I-15)
 2050 CONTINUE
      DEC(I)=DEC(I)+IFIX(ZY*SQRT(T1)+0.5)-IXI(I)
      GO TO 2000
 2040 IF(I-37)2070,2070,2080
 2070 DO 2090 J=1,15
      IF(IXYZ(J,I-15,1))2100,2100,2110
 2110 DEC(I)=DEC(I)+JFOR(J,2)*IXYZ(J,I-15,1)
      T1=T1+(JSE(J,2)**2)*IXYZ(J,I-15,1)
 2100 IF(IXYZ(J,I-15,2))2090,2090,2120
 2120 DEC(I)=DEC(I)+JFOR(J,3)*IXYZ(J,I-15,2)
      T1=T1+(JSE(J,3)**2)*IXYZ(J,I-15,2)
 2090 CONTINUE
      DO 2130 K=1,22
      IF(DEC(K))2130,2130,2131
 2131 DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)
 2130 CONTINUE
      DEC(I)=DEC(I)+IFIX(ZX*SQRT(T1)+0.5)-IXI(I)
      GO TO 2000
 2080 IF(I-41)2140,2140,2150
 2140 DO 2160 J=1,15
      IF(IXYZ(J,I-15,2))2170,2170,2180
 2180 DEC(I)=DEC(I)+JFOR(J,3)*IXYZ(J,I-15,2)
      T1=T1+(JSE(J,3)**2)*IXYZ(J,I-15,2)
 2170 IF(IXYZ(J,I-15,3))2160,2160,2200
 2200 DEC(I)=DEC(I)+JFOR(J,4)*IXYZ(J,I-15,3)
      T1=T1+(JSE(J,4)**2)*IXYZ(J,I-15,3)
 2160 CONTINUE
      DO 2210 K=16,37
      IF(DEC(K))2210,2210,2211
 2211 DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)
 2210 CONTINUE
      DEC(I)=DEC(I)+IFIX(ZW*SQRT(T1)+0.5)-IXI(I)
      GO TO 2000
 2150 DO 2220 J=1,15
```

```
      IF(IXYZ(J,I-15,3))2230,2230,2240                                  DCID1060
 2240 DEC(I)=DEC(I)+(JFOR(J,4)+JFOR(J,3))*IXYZ(J,I-15,3)                DCID1061
      T1=T1+(JSE(J,3)**2+JSE(J,4)**2)*IXYZ(J,I-15,3)                    DCID1062
 2230 IF(IXYZ(J,I-15,4))2220,2220,2250                                  DCID1063
 2250 DEC(I)=DEC(I)+(JFOR(J,4)+JFOR(J,5))*IXYZ(J,I-15,4)                DCID1064
      T1=T1+(JSE(J,4)**2+JSE(J,5)**2)*IXYZ(J,I-15,4)                    DCID1065
 2220 CONTINUE                                                          DCID1066
      DO 2260 K=16,37                                                   DCID1067
      IF(DEC(K))2260,2260,2261                                          DCID1068
 2261 DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)                                 DCID1069
 2260 CONTINUE                                                          DCID1070
      DEC(I)=DEC(I)+IFIX(ZW*SQRT(T1)+0.5)-IXI(I)-IOO(I-37)              DCID1071
 2000 CONTINUE                                                          DCID1072
      DO 20 ID=1,3                                                      DCID1073
      TOT=0.                                                            DCID1074
      IL=IBMR(ID)+1                                                     DCID1075
      IU=IBMR(ID)+IBN(ID)                                               DCID1076
      DO 61 IP=IL,IU                                                    DCID1077
      IF(DEC(IP))61,61,62                                               DCID1078
   62 TOT=TOT+S(IP)+DEC(IP)*U(IP)                                       DCID1079
   61 CONTINUE                                                          DCID1080
      DEC(ID+45)=IFIX(TOT/160.+0.5)                                     DCID1081
   20 CONTINUE                                                          DCID1082
      RETURN                                                            DCID1083
      END                                                               DCID1084
```

```
***      DECIDE 1 - REQUIREMENTS PLANNING                             ***

              * DO 2000   I=1,45                  *         DCID1C12
                              I
              I  DEC(I)=0.                        I         DCID1C13
              I  TI=0.                            I         DCID1C14
                              I
                         IF (I-15)                          DCID1C15
                  I -         I 0           I +
               *2010 *     *2010 *       *2020 *

 2010         I  DEC(I)=JFCR(I,1)+IFIX(ZZ*JSE(I,1)+  I      DCID1C16
              I      C.5)+IBC(I)-IXI(I)              I
                              I
                          * 2000 *                          DCID1C17

 2020                    IF (I-22)                          DCID1C18
                  I -         I 0           I +
               *2030 *     *2030 *       *2040 *

 2030     ---* DO 2050   J=1,15                   *         DCID1019
                              I
                     IF (IXYZ(J,I-15,1))                    DCID1C20
                  I -         I 0           I +
               *2050 *     *2050 *
                              I------------------

 2060         I  DEC(I)=DEC(I)+JFOR(J,2)*IXYZ(J,I-1  I      DCID1021
              I      5,1)                            I
              I  TI=TI+(JSE(J,2)**2)*IXYZ(J,I-15,1)  I      DCID1C22
                              I
```


IF (DEC(J))
- 2050
0 2050
+
2051 DEC(I)=DEC(I)+DEC(J)*IXM1(J,I-15)
2050 CONTINUE
DEC(I)=DEC(I)+IFIX(ZY*SCRT(T1)+0.5)-IXI(I)
2000
2040 IF (I-37)
- 2070
0 2070
+ 2080
2070 DO 2090 J=1,15
IF (IXYZ(J,I-15,1))
- 2100
0 2100
+
2110 DEC(I)=DEC(I)+JFCR(J,2)*IXYZ(J,I-15,1)
T1=T1+(JSE(J,2)**2)*IXYZ(J,I-15,1)
DCIDIC23
DCIDIC24
DCIDIC25
DCIDIC26
DCIDIC27
DCIDIC28
DCIDIC29
DCIDIC30
DCIDIC31
DCIDIC32

```
 2100        IF (IXYZ(J,I-15,2))                                   DCIDIC33
             -: *2090    0: *2090    +

 2120        DEC(I)=DEC(I)+JFCR(J,3)*IXYZ(J,I-15,2)                DCIDIC34
             T1=T1+(JSE(J,3)**2)*IXYZ(J,I-15,2)                    DCIDIC35
 2090        CONTINUE                                              DCIDIC36
             DO 2130 , K=1,22                                      DCIDIC37
             IF (DEC(K))                                           DCIDIC38
             -: *2130    0: *2130    +

 2131        DEC(I)=DEC(I)+DEC(K)*IXP1(K,I-15)                     DCIDIC39
 2130        CONTINUE                                              DCIDIC40
             DEC(I)=DEC(I)+IFIX(ZX*SQRT(T1)+0.5)-IXI(I)            DCIDIC41
             * 2000 *                                              DCIDIC42
 2080        IF (I-41)                                             DCIDIC43
             -: *2140    0: *2140    +: *2150
 2140        DO 2160  J=1,15                                       DCIDIC44
```

```
                                     IF
                               (IXYZ(J,I-15,2))                           CCID1C45
                      I -            I 0
                    *2170 *        *2170 *

2180   I  DEC(I)=DEC(I)+JFCR(J,3)*IXYZ(J,I-1     I                        CCID1C46
       I      5,2)                               I
       I  T1=T1+(JSE(J,3)**2)*IXYZ(J,I-15,2)     I                        CCID1C47

2170                                 IF                                   CCID1C48
                               (IXYZ(J,I-15,3))
                      I -            I 0
                    *2160 *        *2160 *

2200   I  DEC(I)=DEC(I)+JFCR(J,4)*IXYZ(J,I-1     I                        CCID1C49
       I      5,3)                               I
       I  T1=T1+(JSE(J,4)**2)*IXYZ(J,I-15,3)     I                        CCID1C50
2160   +++++++++++++++++CONTINUE                                          CCID1C51
       *  DO  2210   K=16,37                      *                       CCID1C52
                                     IF
                               (DEC(K))                                   CCID1C53
                      I -            I 0
                    *2210 *        *2210 *

2211   I  DEC(I)=DEC(I)+DEC(K)*IXP1(K,I-15)      I                        CCID1C54
2210   +++++++++++++++++CONTINUE                                          CCIC1C55
       I  DEC(I)=DEC(I)+IFIX(ZW*SCRT(T1)+0.5     I                        CCID1056
       I      )-IXI(I)                           I
                          * 2000 *                                        CCID1C57
```

```
2150   * DO 2220  J=1,15 *                                          DCID1C58

       IF (IXYZ(J,I-15,3))                                          DCID1C59
       -  *2230*     0  *2230*     +

2240   DEC(I)=DEC(I)+(JFCR(J,4)+JFCR(J,3)                           DCID1C60
         )*IXYZ(J,I-15,3)
       T1=T1+(JSE(J,3)**2+JSE(J,4)**2)*IX                           DCID1C61
         YZ(J,I-15,3)

2230   IF (IXYZ(J,I-15,4))                                          DCID1C62
       -  *2220*     0  *2220*     +

2250   DEC(I)=DEC(I)+(JFCR(J,4)+JFCR(J,5)                           DCID1C63
         )*IXYZ(J,I-15,4)
       T1=T1+(JSE(J,4)**2+JSE(J,5)**2)*IX                           DCID1C64
         YZ(J,I-15,4)

2220   CONTINUE                                                     DCID1C65

       * DO 2260  K=16,37 *                                         DCID1C66

       IF (DEC(K))                                                  DCID1C67
       -  *2260*     0  *2260*     +

2261   DEC(I)=DEC(I)+DEC(K)*IXP1(K,I-15)                            DCID1C68

2260   CONTINUE                                                     DCID1C69

       DEC(I)=DEC(I)+IFIX(ZN*SQRT(T1)+0.5                           DCID1C70
         )-IXI(I)-ICO(I-37)
```

```
                                                                          CCID1-REQ PLANNING   PAGE NO .  6
                                                       I
                                       --------------------------------
 ---------------------------------*  DO  20      ID=1,3              *                            CCID1072
 +                                     --------------------------------
 +                                                     I
 +                                     --------------------------------
 +                                     I   TOT=0.                      I                            CCID1C73
 +                                     I   IL=IBMR(ID)+1               I                            CCID1C74
 +                                     I   IU=IBMR(ID)+IBN(ID)         I                            CCID1C75
 +                                     --------------------------------
 +                                                     I
 +                                     --------------------------------
 +---------------------------------*  DO  61      IP=IL,IU            *                            CCID1C76
 +I                                    --------------------------------
 +I                                                    I
 +I                                              . ' . ' .
 +I                                         . '    IF      ' .                                     CCID1C77
 +I                                     --' . (DEC(IP))        ' . '--
 +I                                     I   ' .              . '      I
 +I                                     I       ' .      . '          I
 +I                                     I -         I 0               I +
 +I                                   ------      ------              I
 +I                                   *61  *      *61  *              I
 +I                                    ' . '       ' . '              I
 +I                                                    I---------------I
 +I                                                    I
 +I                                    --------------------------------
 +I                       62           I   TOT=TOT+S(IP)+DEC(IP)*U(IP)  I                           CCID1C78
 +I                                    --------------------------------
 ++++++++++++++++++++++++++ 61   ++++++++++++++++++CONTINUE                                         CCID1C79
 +                                     --------------------------------
 +                                     I   DEC(ID+45)=IFIX(TOT/160.+0.5)  I                         CCID1C80
 +                                     --------------------------------
 ++++++++++++++++++++++++++ 20   ++++++++++++++++++CONTINUE                                         CCID1C81
                                            -----------------------
                                            I RETURN                 ' .                            CCID1C82
                                            I                          :
                                            I                        . '
                                            -----------------------
***END*******END*******END*******END*******END*******END*******END*******END*******END*******END*******END*******END***   CCID1083
```

APPENDIX V

AN ANALYTICAL SOLUTION TO THE
PRODUCTION CONTROL PROJECT - DECIDE
USING A LOT-FOR-LOT, LOT SIZING MODEL

Decisions

Finished Products (I=1,15)

$$DEC(I,T) = FOR(I,T) + ZZ(I)*SE(I,T) + BO(I) - IXI(I)$$

Subassemblies (I=16,22)

$$DEC(I,T)= \sum_{J=1}^{15} (FOR(J,T+1)*XYZ(J,I,1)+DEC(J)*XM(J,I))$$

$$+ ZZ(I)\sqrt{\sum_{J=1}^{15} SE(J,T+1)^2*XYZ(J,I,1)} -IXI(I).$$

Parts (I=23,37)

$$DEC(I,T)= \sum_{J=1}^{15} (FOR(J,T+1)*XYZ(J,I,1)+FOR(J,T+2)*XYZ(J,I,2))$$

$$+ \sum_{K=1}^{22} (DEC(K)*XM(K,I)) - IXI(I)$$

$$+ZZ(I)\sqrt{\sum_{J=1}^{15} (SE(J,T+1)^2*XYZ(J,I,1)+SE(J,T+2)^2*XYZ(J,I,2))}.$$

Raw Materials (I=38,41)

$$DEC(I,T)= \sum_{J=1}^{15} (FOR(J,T+2)*XYZ(J,I,2)+FOR(J,T+3)*XYZ(J,I,3))$$

$$+ \sum_{K=16}^{37} (DEC(K)*XM(K,I)) - IXI(I)$$

$$+ZZ(I)\sqrt{\sum_{J=1}^{15} SE(J,T+2)^2*XYZ(J,I,2)+SE(J,T+3)^2*XYZ(J,I,3)} .$$

Raw Materials (I=42,45)

$$DEC(I,T)= \sum_{J=1}^{15} (FOR(J,T+3)*XYZ(J,I,3)+FOR(J,T+4)*XYZ(J,I,4))$$

$$+\sum_{K=16}^{37} (DEC(K,T)*XM(K,I)) - IXI(I) - IOO(I)$$

$$+ZZ(I)\sqrt{\sum_{J=1}^{15} SE(J,T+3)^2*XYZ(J,I,3)+SE(J,T+4)^2*XYZ(J,I,4)} \; .$$

Workforce (ID=1,3)

$$DEC(ID=1,T)=INTEGER\left[\frac{\sum_{I=1}^{15}(S(I)+DEC(I)*U(I))}{160.} + 0.5\right] \quad (IF\ DEC(I)>0)$$

$$DEC(ID=2,T)=INTEGER\left[\frac{\sum_{I=16}^{22}(S(I)+DEC(I)*U(I))}{160.} +0.5\right] \quad (IF\ DEC(I)>0)$$

$$DEC(ID=3,T)=INTEGER\left[\frac{\sum_{I=23}^{37}(S(I)+DEC(I)*U(I))}{160.} +0.5\right] \quad (IF\ DEC(I)>0)$$

Labour Adjustment Costs

Hiring Costs

$$HC(T) = \left(\sum_{ID=1}^{3} (DEC(ID,T)-DEC(ID,T-1))\right)^{+}*H$$

Superscript $^{+}$ indicates that only positive values are considered.

Firing Costs

$$FC(T) = \left(\sum_{ID=1}^{3} (DEC(ID,T)-DEC(ID,T-1))\right)^{-}*F$$

Superscript $^{-}$ indicates that only negative values are considered.

Transfer Costs

$$TC(T) = MINIMUM\left[\begin{array}{l}\left|\sum_{ID=1}^{3} (DEC(ID,T)-DEC(ID,T-1))^{+}\right| \\ \left|\sum_{ID=1}^{3} (DEC(ID,T)-DEC(ID,T-1))^{-}\right|\end{array}\right]*T$$

Operate Departments (ID=1,3)

$$IL = IBMR(ID)+1$$
$$IU = IBMR(ID)+IBN(ID)$$

Material Requirements (IZ=IU+1,45)

$$IXM(IZ,T) = \sum_{IP=IL}^{IU} XM(IP,IZ)*DEC(IP,T)$$

$$FA(IZ) = \begin{cases} \frac{IXI(IZ)}{IXM(IZ,T)} & \text{IF } IXM(IZ,T) > IXI(I) \\ 1 & \text{IF } IXM(IZ,T) \leq IXI(I) \end{cases}$$

$$IXM(IZ,T) = IXM(IZ,T)+ \sum_{IP=IL}^{IU} XM(IP,IZ)*DEC(IP,T)*(FA(IZ)-1)$$

$$DEC(IP,T) = DEC(IP,T)*FA(IZ)*XM(IP,IZ)$$

Manpower Requirements (ID=1,3)

$$RT(ID) = \sum_{IP=IL}^{IU} (S(IP) + DEC(IP,T)*U(IP))^{+}$$

$$ST(ID) = 160. *DEC(ID,T)$$

$$OT(ID) = \begin{cases} 0 & \text{IF } ST(ID) \geq RT(ID) \\ RT(ID)-ST(ID) & \text{IF } ST(ID) < RT(ID) \end{cases}$$

$$G(ID) = \frac{ST(ID) + 1.5 *OT(ID)}{ST(ID) + OT(ID)} *WR$$

$$TL(IP) = (S(IP)+DEC(IP,T)*U(IP))^{+}*G(ID)$$

$$Z(IZ) = \frac{V(IZ)}{IXI(IZ)}$$

$$TV(IP) = \sum_{IZ=IU+1}^{45} DEC(IP)*XM(IP,IZ)*Z(IZ)$$

$$IXI(IZ) = IXI(IZ)-DEC(IP,T)*XM(IP,IZ)$$

$$V(IZ) = IXI(IZ)*Z(IZ)$$

$$IXI(IP) = IXI(IP) + DEC(IP,T)$$

$$V(IP) = V(IP) + TL(IP) + TV(IP)$$

Sales of Finished Products (I=1,15)

$$SFG(I,T) = FOR(I,T) + E(I,T)$$

$$JD(I,T) = SFG(I,T) + BO(I)$$

$$US(I,T) = \begin{Bmatrix} IXI(I)+DEC(I,T) \\ JD(I,T) \end{Bmatrix} \begin{matrix} \text{IF } JD(I,T)>IXJ(I)+DEC(I,T) \\ \text{IF } JD(I,T)\leq IXI(I)+DEC(I,T) \end{matrix}$$

$$IXI(I) = (IXI(I)+DEC(I,T) - JD(I,T))^{+}$$

$$BO(I) = -(IXI(I)+DEC(I,T) - JD(I,T))^{-}$$

$$Z(I) = \frac{V(I)}{IXI(I)}$$

$$V(I) = V(I) - US(I)*Z(I)$$

Raw Materials

I = 38,41

$$IXI(I) = IXI(I) + DEC(I,T)$$

$$V(I) = V(I) + DEC(I,T)*RC(I)$$

I = 42,45

$$IXI(I) = IXI(I) + IOO(I)$$

$$V(I) = V(I) + IOO(I)*RC(I)$$

$$IOO(I) = DEC(I,T)$$

Ordering Costs (I=38,45)

$$\delta(I) = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix} \begin{matrix} \text{IF } DEC(I,T) \\ \text{IF } DEC(I,T) \end{matrix}$$

$$OC(T) = \sum_{I=38}^{45} \delta(I)*SC$$

Backorder Costs (I=1,15)

$$BOC(T) = \sum_{I=1}^{15} BO(I)*BC$$

Inventory Holding Costs (I=1,45)

$$IHC(T) = \sum_{I=1}^{45} V(I)*XIC$$

Total Costs

$$TC = \sum_{T=1}^{N} \Bigg[HC(T) + FC(T) + TC(T) + \sum_{ID=1}^{3} \left(\sum_{IP=IL}^{IU} (S(IP)+DEC(IP)*U(IP))^{+} \right)*G(ID) + OC(T) + \sum_{I=38}^{41} DEC(I,T)*RC(I) + \sum_{I=42}^{45} IOO(I)*RC(I) + BOC(T) + IHC(T) \Bigg].$$

The objective is to minimize the total cost over the planning horizon N. The decision variable is the safety stock factors ZZ(I). Therefore the total cost equation must be differentiated with respect to each of the 45 ZZ(I)'s. The resulting 45 equations are set equal to zero and solved simultaneously. An examination of the backorder charges will indicate to complexity of this solution procedure.

$$BOC(T) = \sum_{I=1}^{15} BO(I)*BC$$

$$= \sum_{I=1}^{15} (FOR(I,T)+E(I,T)+BO(I) -IXI(I) -DEC(I,T))^{+}*BC$$

$$= \sum_{I=1}^{15} (FOR(I,T)+E(I,T)+BO(I)-IXI(I)$$

$$- \sum_{J=16}^{37} \{FA(J)*DEC(I,T)*XM(I,J)\})^{+}*BC$$

$$= \sum_{I=1}^{15} \Bigg(FOR(I,T)+E(I,T)+BO(I)-IXI(I)$$

$$- \sum_{J=16}^{37} \left\{ \frac{IXI(J)*DEC(I,T)}{\sum_{K=1}^{15} XM(K,J)*DEC(K,T)} *XM(I,J) \right\} \Bigg)^{+}*BC$$

$$BOC(T) = \sum_{I=1}^{15} \Bigg(FOR(I,T)+E(I,T)+BO(I)-IXI(I)$$

$$- \sum_{J=16}^{13} \left\{ \frac{IXI(J)*\ FOR(I,T)+ZZ(I)*SE(I,T)+BO(I)-IXI(I)\quad *XM(I,J)}{\sum_{J=16}^{15} XM(K,J)*\{FOR(K,T)+ZZ(K)*SE(K,T)+BO(K)-IXI(K)\}} \right\} \Bigg)^{+}*BC$$

To differentiate the expression for the backorder costs, an expectation function for the stochastic random demand function must be included. Therefore:

$$E\left[BOC(T)\right] = \sum_{I=1}^{15} BC \int_{ZZ(I)*SE(I,T)}^{\infty} \Bigg(\Big[FOR(I,T)+BO(I)-IXI(I)+E(I,T) \Big]$$

$$- \sum_{J=16}^{37} \left\{ \frac{IXI(J)*\ FOR(I,T)+ZZ(I)*SE(I,T)+BO(I)-IXI(I)\quad *XM(I,J)}{\sum_{K=1}^{15} XM(K,J)*\{FOR(K,T)+ZZ(K)*SE(K,T)+BO(K)-IXI(K)\}} \right\} \Bigg)^{+}$$

$$*f\{E(I,T)\}d\{E(I,T)\}.$$

where: $f\{E(I,T)\}$ is the height of the density function of the probability distribution on E(I,T) where the random variate is measured from the mean.

To differentiate the expectation function, use Leibniz's rule:

$$\text{If } G(x) = \int_{h(x)}^{k(x)} g(x,y)\, dy ,$$

$$\text{then } \frac{d\,G(x)}{dx} = \int_{h(x)}^{k(x)} \frac{\partial g(x,y)}{\partial x}\, dy + g\left[k(x),x\right]\frac{dk(x)}{dx} - g\left[h(x),x\right]\frac{dh(x)}{dx}.$$

For the expectation function:

$k(x) = \infty$

$h(x) = ZZ(I)*SE(I,T).$

$g(x,y) = (\qquad) f\{E(I,T)\}.$

$$\frac{d\,k(x)}{dx} = 0$$

$$\frac{d\,h(x)}{dx} = SE(I,T)$$

$$\frac{\partial g(x,y)}{\partial x} = \frac{\partial(\qquad) f\{E(I,T)\}}{\partial\, ZZ(I)}$$

This partial derivative takes the form of

$$\frac{d}{dx}\left(\frac{u}{v}\right) \qquad \frac{v\frac{du}{dx} - u\frac{dv}{dx}}{v^2}$$

Let: $\alpha(I,T) = FOR(I,T)+BO(I) - IXI(I)$

Then:

$$E\left[BOC(T)\right] = \sum_{I=1}^{15} BC \int_{ZZ(I)*SE(I,T)}^{\infty} \Bigg(\alpha(I,T)+E(I,T)$$

$$- \sum_{J=16}^{37} \left\{ \frac{IXI(J)*\alpha(I,T)+ZZ(I)*SE(I,T) \quad *XM(I,J)}{\sum_{K=1}^{15} XM(K,J)*(\alpha(K,T)+ZZ(K)*SE(K,T)} \right\} \Bigg)^{+}$$

$$*f\{E(I,T)\}d\{E(I,T)\} .$$

Let: $U = IXI(I)*\left[\alpha(I,T)+ZZ(I)*SE(I,T)\right]*XM(I,J)$.

$V = \sum_{K=1}^{15} XM(K,J)*(\alpha(K,T)+ZZ(K)*SE(K,T)$.

Then:

$\frac{du}{dZZ(I)} = IXI(J)*SE(I,T)*XM(I,J)$.

$\frac{dv}{dZZ(I)} = XM(I,J)*SE(I,T)$.

Therefore:

$$\frac{\partial(\quad)f\{E(I,T)\}}{\partial(ZZ(I))} = \sum_{J=16}^{37}\left[\frac{\sum_{K=1}^{15} XM(K,J)*(\alpha(K,T)+ZZ(K)*SE(K,T)) * IXI(I)*SE(I,T)*XM(I,J) - IXI(I)*(\alpha(I,T)+ZZ(I)*SE(I,T)*XM(I,J) * XM(I,J)*SE(I,J)}{\sum_{K=1}^{15} XM(I,J)*\{\alpha(K,T)+ZZ(K)*JE(K,T)\}}\right]$$

$$*f\{E(I,T)\}$$

$$= \beta(I,T)f\{E(J,T)\}$$

$$g\left[h(x,x\right]\frac{dh(x)}{dx} = \Bigg(\{\alpha(I,T)+ZZ(I)*SE(I,T)\}$$

$$- \sum_{J=16}^{37}\left\{\frac{IXI(I)*\ \alpha(I,T)+ZZ*SE(I,T)\ *XM(J,J)}{\sum_{K=1}^{15} XM(K,J)*\ \alpha(K,T)+ZZ(K)*SE(K,T)}\right\}\Bigg)*SE(I,T)*f\{ZZ(I)*SE(I,T)\}.$$

$$= \Gamma\ (I,T)\ f\{ZZ(I)*SE(I,T)\}$$

$$g\left[k(x),x\right]\ \frac{d\ k(x)}{dx} = 0$$

The differentiated equation is therefore

$$\frac{\partial E[BOC(T)]}{\partial ZZ(I)} = \int_{ZZ(I)*SE(I,T)}^{\infty} \beta(I,T) f\{E(I,T)\}\, d\{(I,T)\} + \Gamma(I,T) f\{ZZ(I)*SE(I,T)\}$$

The objective is to minimize the total cost of operations over the planning horizon of N periods. Therefore equations of the form above will be included in each of the I5 equations differentiated with respect to the 15 finished products in each of the N periods of the planning horizon. The inclusion of the decision variable in the limits of the integration operator and as a parameter in the density function operator indicates that a search procedure is necessary to find the optimizing value of each ZZ(I). Simulation is a much more tractable solution procedure requiring much less time to reach a solution with little if any loss of accuracy.

Listing of Variables

DEC(I,T) = the production quantity decision for product I in time period T.

FOR(I,T) = the forecast demand of product I in time period T.

ZZ(I) = the multiple of the standard deviation used to determine the safety stock for product I.

SE(I,T) = the standard error of the forecast demand of product I in period T.

BO(I) = the current number of backorders for product I.

IXI(I) = the current number of units of product I in inventory.

XYZ(A,B,C) = the time-phase requirements of product A for component B in C time periods.

XM(J,I) = the number of units of component I used directly in the assembly of product J.

S(I) = the set-up time in man-hours to ready the production equipment to assemble product I.

U(I) = the assembly time in man-hours required to produce a unit of product I.

HC(T) = hiring cost in period T.

H = hiring cost per man added to the total workforce.

FC(T) = firing cost in period T.

F = firing cost per man discharged from the total workforce.

TC(T) = transfer cost in period T.

T = transfer cost per man transferred from one department to another department within the system.

IL = lowest product number for a department.

IU = highest product number for a department.

IBMR(ID) = lowest product number for department ID minus 1.

IBN(ID) = number of products assembled in department ID.

IXM(I,T) = number of units of component IZ required in period T to facilitate the assembly of the products of another department.

FA(IZ) = ratio of required to available inventory of of component IZ.

RT(ID) = total man-hours required in department ID.

ST(ID) = standard time man-hours available in department ID.

OT(ID) = overtime man-hours required in department ID.

G(ID) = average hourly wage rate in department ID adjusted for the bonus on overtime hours.

TL(IP) = labour cost for assembly of product IP.

V(IZ) = value of the inventory of item IZ.

Z(IZ) = unit value of item IZ.

TV(IP) = value added to inventory of item IZ.

SFG(I,T) = sales of finished product I in period T.

E(I,T) = normally distributed random variation of sales with mean zero and standard deviation SE(I,T).

JD(I,T) = total demand for finished product I in period t.

RC(I) = cost per unit of raw material item I.

δ(I) = indicator of the positiveness of DEC(I,T)

δC = cost of placing an order for a raw material.

OC(T) = total ordering cost in period T.

BC = cost per unit backordered.

BOC(T) = total cost of backorders in period T.

XIC = inventory holding cost expressed as a fraction of the value of period ending inventory value.

IHC(T) = total inventory holding cost in period T.

APPENDIX VI

THE STATISTICAL STANDARD DEVIATION MODEL

```
      SUBROUTINE DECIDE                                                 DCID4000
      INTEGER WSTAR                                                     DCID4001
      COMMON IXM1(37,30),S(37),U(37),IXI(45),V(45),STH(3),OTH(3),SUH(3),DCID4002
     1XIH(3),IBMR(3),IBN(3),IXLW(3),ISFG(15),IUS(15),IBO(15),ISL(15),   DCID4003
     2ITD(15),ITB(15),ITSL(15),AS(15),BS(15),CS(15),IFOR(15,6),COSTY(5),DCID4004
     3COST(5),COSTT,SHAT(12),ALPHA(15),BETA(15),XMU(15),SIG(15),FGP(15),DCID4005
     4XMUX(3),SIGX(3),ICO(8),KC(8),E(15),A(15),B(15),IXM(45),DEC(48),   DCID4006
     5AP(45),ITI(45),TV(45),TSTH(3),TOTH(3),TSUH(3),TRH(3),RH(3),IC(45) DCID4007
      COMMON STAR,WSTAR,PSTAR,WR,XMOT,IHC,ITC,IFC,XIC,IBC,ISC,N,MMAX,M  DCID4008
      COMMON COSTM,TIH(3),JFOR(45,18),JSE(45,18),ZZ,ZY,ZX,ZW            DCID4009
      COMMON IXYZ(15,30,4),ICQ(45),ISS(45),ISA(45,72)                   DCID4010
C        DCID4 - STATISTICAL WITH STANDARD DEVIATIONS                   DCID4011
      M1=M-1                                                            DCID4012
      DO 4000 I=1,45                                                    DCID4013
      T1=0.                                                             DCID4014
      T2=0.                                                             DCID4015
      T3=0.                                                             DCID4016
      T4=0.                                                             DCID4017
      DEC(I)=0.                                                         DCID4018
      IF(I-15)4010,4010,4020                                            DCID4019
 4010 T1=0.                                                             DCID4020
      DO 4030 K=1,12                                                    DCID4021
 4030 T1=T1+JFOR(I,K)                                                   DCID4022
      IF(IXI(I))4031,4031,4032                                          DCID4023
 4032 ICQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)/FLOAT(IXI(I)))*12.*XIC))+0.5   DCID4024
 4031 ISS(I)=ZZ*JSE(I,1)+0.5                                            DCID4025
      IF(ICQ(I)-JFOR(I,1))4033,4033,4034                                DCID4026
 4033 ICQ(I)=JFOR(I,1)                                                  DCID4027
 4034 CONTINUE                                                          DCID4028
      IF(IXI(I)-JFOR(I,1)-ISS(I))4040,4050,4050                         DCID4029
 4040 DEC(I)=ICQ(I)+IBO(I)+ISS(I)-IXI(I)                                DCID4030
      GO TO 4000                                                        DCID4031
 4050 DEC(I)=0.                                                         DCID4032
      GO TO 4000                                                        DCID4033
 4020 IF(I-22)4060,4060,4070                                            DCID4034
 4060 DO 4080 J=1,15                                                    DCID4035
      IF(IXYZ(J,I-15,1))4080,4080,4090                                  DCID4036
 4090 DO 4100 K=2,13                                                    DCID4037
 4100 T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)                                    DCID4038
      T2=T2+DEC(J)*IXM1(J,I-15)                                         DCID4039
 4080 CONTINUE                                                          DCID4040
      ISA(I,M)=T2                                                       DCID4041
      IF(M-3)4081,4081,4082                                             DCID4042
 4082 DO 4083 MM=1,M                                                    DCID4043
 4083 T4=T4+ISA(I,MM)                                                   DCID4044
      XM=T4/M                                                           DCID4045
      DO 4084 MM=1,M                                                    DCID4046
 4084 T3=T3+(XM-ISA(I,MM))**2                                           DCID4047
      T3=T3/M1                                                          DCID4048
 4081 CONTINUE                                                          DCID4049
      IF(IXI(I))4061,4061,4062                                          DCID4050
 4062 ICQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)/FLOAT(IXI(I)))*12.*XIC))+0.5   DCID4051
 4061 ISS(I)=ZY*SQRT(T3)+0.5                                            DCID4052
      DO 4065 J=1,15                                                    DCID4053
      IF(IXM1(J,I-15))4065,4065,4066                                    DCID4054
 4066 IF(ICQ(J)-ICQ(I))4065,4065,4067                                   DCID4055
 4067 ICQ(I)=ICQ(J)                                                     DCID4056
 4065 CONTINUE                                                          DCID4057
      IF(IXI(I)-T2-ISS(I))4110,4110,4120                                DCID4058
 4110 DEC(I)=ICQ(I)+ISS(I)-IXI(I)+T2                                    DCID4059
```

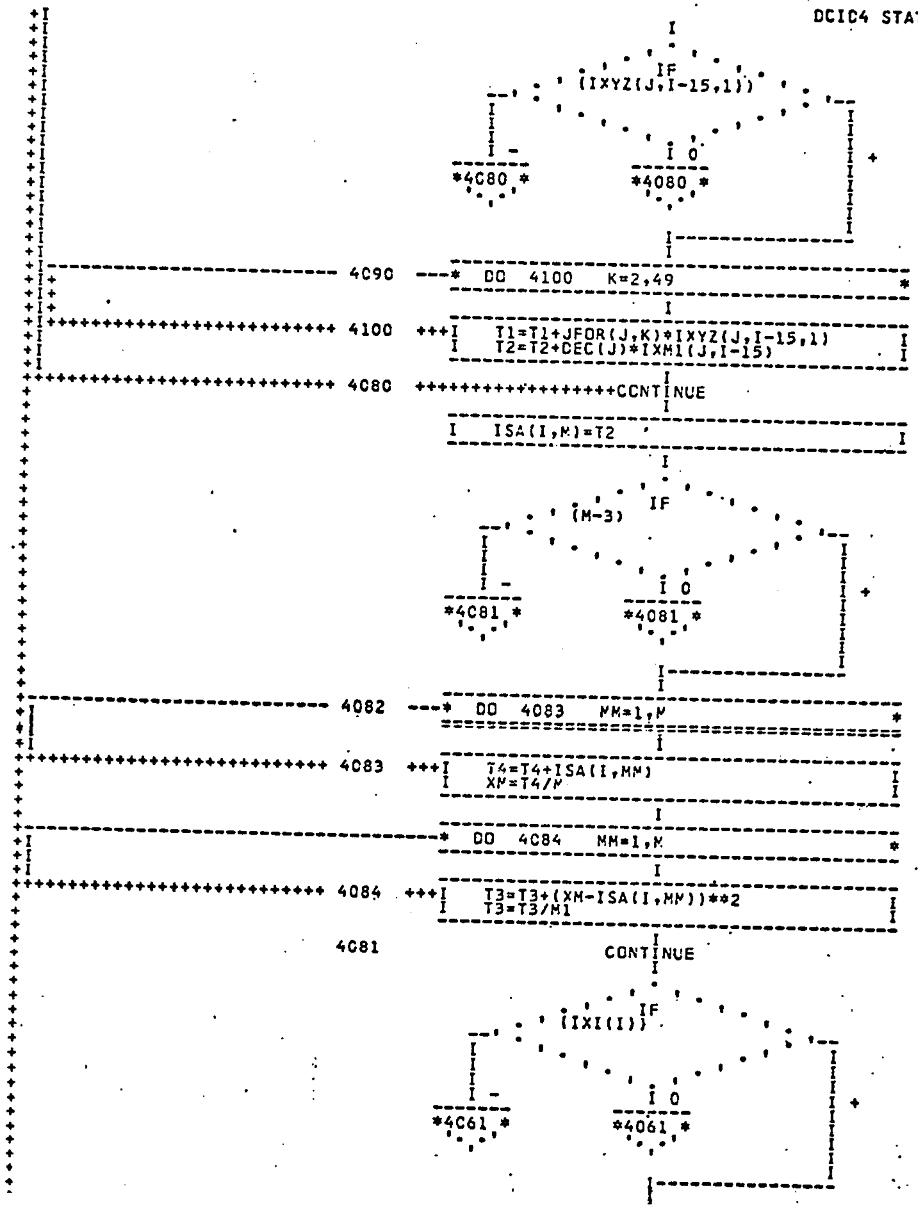
```
      GO TO 4000
 4120 DEC(I)=0.
      GO TO 4000
 4070 IF(I-37)4130,4130,4140
 4130 DO 4150 J=1,15
      IF(IXYZ(J,I-15,1))4160,4160,4170
 4170 DO 4180 K=2,13
 4180 T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)
 4160 IF(IXYZ(J,I-15,2))4150,4150,4190
 4190 DO 4210 K=3,14
 4210 T1=T1+JFOR(J,K)*IXYZ(J,I-15,2)
 4150 CONTINUE
      DO 4220 K=1,22
      IF(DEC(K))4220,4220,4230
 4230 T2=T2+DEC(K)*IXM1(K,I-15)
 4220 CONTINUE
      ISA(I,M)=T2
      IF(M-3)4231,4231,4232
 4232 DO 4233 MM=1,M
 4233 T4=T4+ISA(I,MM)
      XM=T4/M
      DO 4234 MM=1,M
 4234 T3=T3+(XM-ISA(I,MM))**2
      T3=T3/M1
 4231 CONTINUE
      IF(IXI(I))4221,4221,4222
 4222 IQQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)/FLOAT(IXI(I)))*12.*XIC))+0.5
 4221 ISS(I)=ZX*SQRT(T3)+0.5
      DO 4225 J=1,22
      IF(IXM1(J,I-15))4225,4225,4226
 4226 IF(ICC(J)-ICC(I))4225,4225,4227
 4227 IQQ(I)=ICC(J)
 4225 CONTINUE
      IF(IXI(I)-T2-ISS(I))4240,4240,4250
 4240 DEC(I)=IQQ(I)+ISS(I)-IXI(I)+T2
      GO TO 4000
 4250 DEC(I)=0.
      GO TO 4000
 4140 IF(I-41)4260,4260,4270
 4260 DO 4280 J=1,15
      IF(IXYZ(J,I-15,2))4290,4290,4300
 4300 DO 4310 K=3,14
 4310 T1=T1+JFOR(J,K)*IXYZ(J,I-15,2)
 4290 IF(IXYZ(J,I-15,3))4280,4280,4311
 4311 DO 4320 K=4,15
 4320 T1=T1+JFOR(J,K)*IXYZ(J,I-15,3)
 4280 CONTINUE
      DO 4330 K=16,37
      IF(DEC(K))4330,4330,4340
 4340 T2=T2+DEC(K)*IXM1(K,I-15)
 4330 CONTINUE
      ISA(I,M)=T2
      IF(M-3)4341,4341,4342
 4342 DO 4343 MM=1,M
 4343 T4=T4+ISA(I,MM)
      XM=T4/M
      DO 4344 MM=1,M
 4344 T3=T3+(XM-ISA(I,MM))**2
      T3=T3/M1
 4341 CONTINUE
```

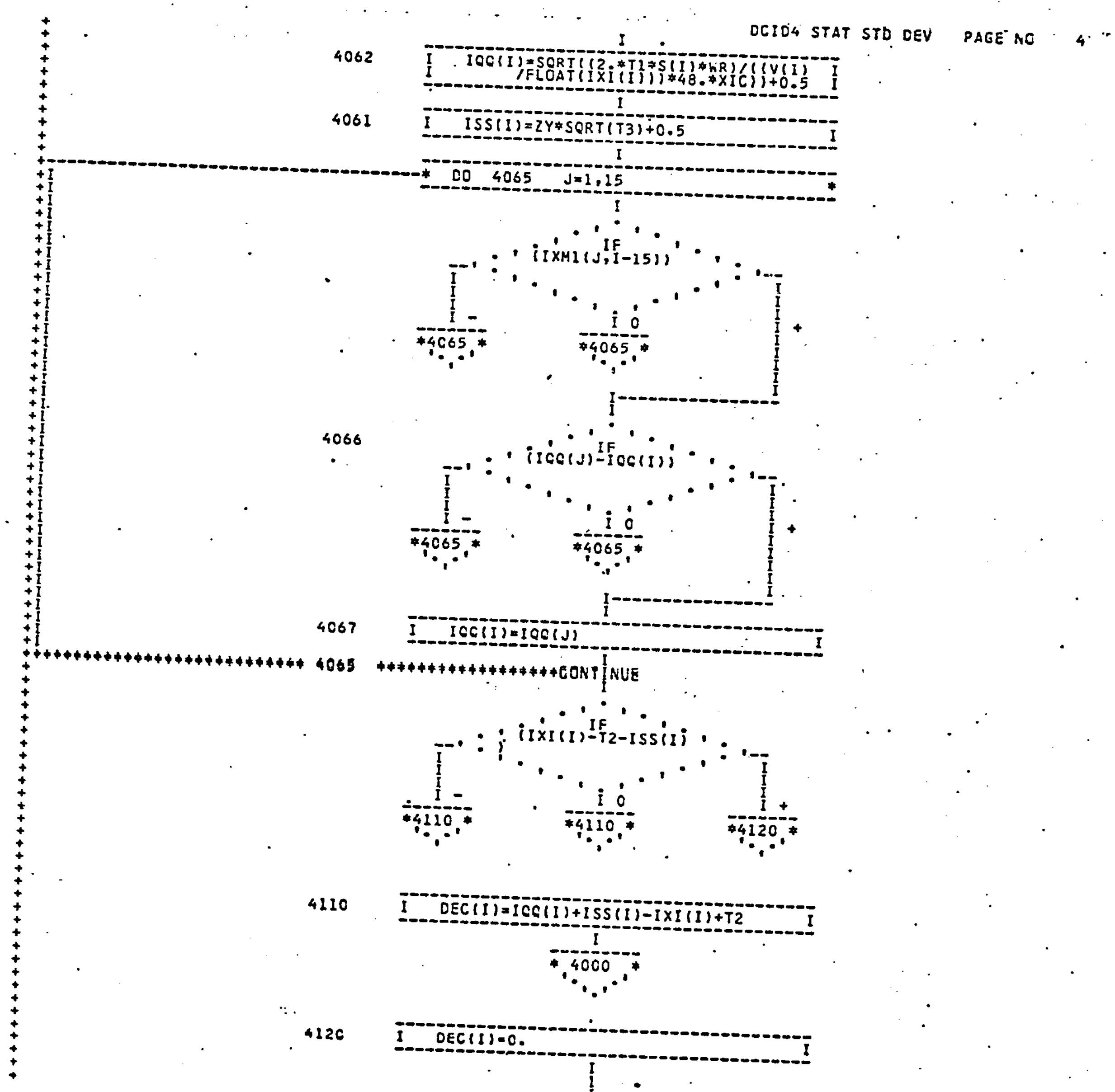
```
4360  IQQ(I)=SQRT((2.*T1*FLOAT(ISC))/(RC(I-37)*12.*XIC))+0.5          DCID4120
4350  ISS(I)=ZW*SQRT(T3)+0.5                                          DCID4121
      DO 4365 J=1,37                                                  DCID4122
      IF(IXM1(J,I-15))4365,4365,4366                                  DCID4123
4366  IF(ICQ(J)-ICQ(I))4365,4365,4367                                 DCID4124
4367  ICQ(I)=IQQ(J)                                                   DCID4125
4365  CONTINUE                                                        DCID4126
      IF(IXI(I)-T2-ISS(I))4370,4370,4380                              DCID4127
4370  DEC(I)=IQQ(I)+ISS(I)-IXI(I)+T2                                  DCID4128
      GO TO 4000                                                      DCID4129
4380  DEC(I)=0.                                                       DCID4130
      GO TO 4000                                                      DCID4131
4270  DO 4390 J=1,15                                                  DCID4132
      IF(IXYZ(J,I-15,3))4400,4400,4410                                DCID4133
4410  DO 4420 K=4,15                                                  DCID4134
4420  T1=T1+JFCR(J,K)*IXYZ(J,I-15,3)                                  DCID4135
4400  IF(IXYZ(J,I-15,4))4390,4390,4430                                DCID4136
4430  DO 4440 K=5,16                                                  DCID4137
4440  T1=T1+JFCR(J,K)*IXYZ(J,I-15,4)                                  DCID4138
4390  CONTINUE                                                        DCID4139
      DO 4450 K=16,37                                                 DCID4140
      IF(DEC(K))4450,4450,4460                                        DCID4141
4460  T2=T2+DEC(K)*IXM1(K,I-15)                                       DCID4142
4450  CONTINUE                                                        DCID4143
      ISA(I,M)=T2                                                     DCID4144
      IF(M-3)4451,4451,4452                                           DCID4145
4452  DO 4453 MM=1,M                                                  DCID4146
4453  T4=T4+ISA(I,MM)                                                 DCID4147
      XM=T4/M                                                         DCID4148
      DO 4454 MM=1,M                                                  DCID4149
4454  T3=T3+(XM-ISA(I,MM))**2                                         DCID4150
      T3=T3/M1                                                        DCID4151
4451  CONTINUE                                                        DCID4152
4461  IQQ(I)=SQRT((2.*T1*FLOAT(ISC))/(RC(I-37)*12.*XIC))+0.5          DCID4153
4470  ISS(I)=ZW*SQRT(T3)+0.5                                          DCID4154
      DO 4465 J=1,37                                                  DCID4155
      IF(IXM1(J,I-15))4465,4465,4466                                  DCID4156
4466  IF(ICQ(J)-ICQ(I))4465,4465,4467                                 DCID4157
4467  ICQ(I)=ICQ(J)                                                   DCID4158
4465  CONTINUE                                                        DCID4159
      IF(IXI(I)+ICC(I-37)-T2-ISS(I))4480,4480,4490                    DCID4160
4480  DEC(I)=ICQ(I)+ISS(I)-IXI(I)-IOO(I-37)+T2                        DCID4161
      GO TO 4000                                                      DCID4162
4490  DEC(I)=0.                                                       DCID4163
4000  CONTINUE                                                        DCID4164
      IF(M-3)4550,4550,4560                                           DCID4165
4550  DO 4570 I=1,45                                                  DCID4166
4570  DEC(I)=ICQ(I)                                                   DCID4167
4560  CONTINUE                                                        DCID4168
      DO 20 IC=1,3                                                    DCID4169
      TOT=0.                                                          DCID4170
      IL=IBMR(IC)+1                                                   DCID4171
      IU=IBMR(IC)+IBN(IC)                                             DCID4172
      DO 61 IP=IL,IU                                                  DCID4173
      IF(DEC(IP))61,61,62                                             DCID4174
62    TOT=TOT+S(IP)+DEC(IP)*U(IP)                                     DCID4175
61    CONTINUE                                                        DCID4176
20    DEC(IC+45)=IFIX(TOT/160.+0.5)                                   DCID4177
      RETURN                                                          DCID4178
      END                                                             DCID4179
```

*** DCID4 - STATISTICAL WITH STANDARD DEVIATIONS ***

```
        I  M1=M-1                                 I
                          I
        *  DO  4000   I=1,45                      *
                          I
        I  T1=0.                                  I
        I  T2=0.                                  I
        I  T3=0.                                  I
        I  T4=0.                                  I
        I  DEC(I)=0.                              I
                          I
                     IF (I-15)
              -           0            +
           *4010 *     *4010 *      *4020 *

4010    I  T1=0.                                  I
                          I
        *  DO  4030   K=1,48                      *
                          I
4030 +++I  T1=T1+JFOR(I,K)                        I
                          I
                     IF (IXI(I))
              -           0            +
           *4031 *     *4031 *

4032    I  IQQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)     I
        I        /FLOAT(IXI(I)))*48.*XIC))+0.5    I
                          I
4031    I  ISS(I)=ZZ*JSE(I,1)+0.5                 I
                          I
```

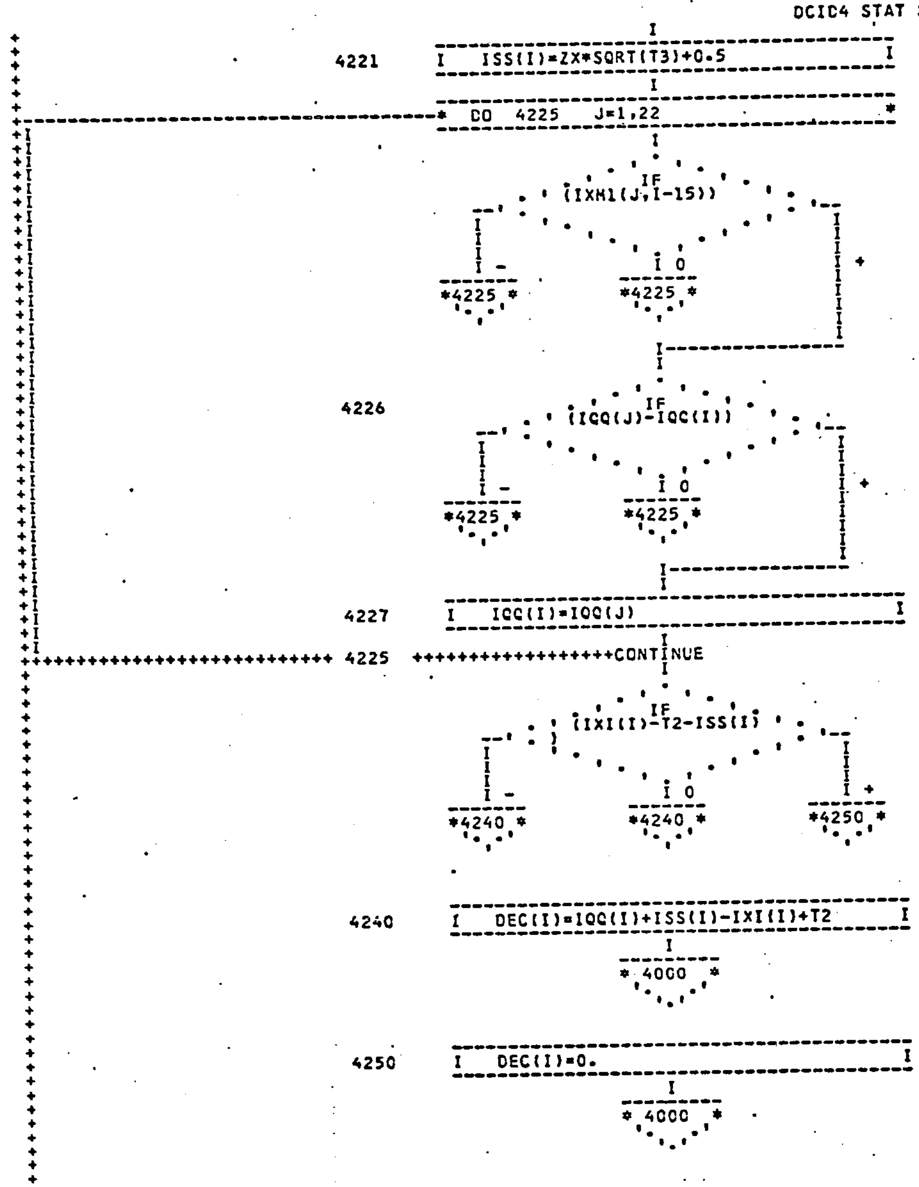
```
                         I
                         IF
              (IQQ(I)-JFCR(I,1)
         I -             I 0              I +
       *4033 *         *4033 *          *4034 *

4033    I   ICQ(I)=JFCR(I,1)                        I

4034             CONTINUE

                         IF
              (IXI(I)-JFCR(I,1)
              -ISS(I))
         I -             I 0              I +
       *4040 *         *4050 *          *4050 *

4040    I   DEC(I)=IQQ(I)+IBG(I)+ISS(I)-IXI(I)   I
                         I
                      * 4000 *

4050    I   DEC(I)=0.                               I
                         I
                      * 4000 *

4020                     IF
              (I-22)
         I -             I 0              I +
       *4060 *         *4060 *          *4070 *

4060 ---*   DO  4080   J=1,15                       *
                         I
```

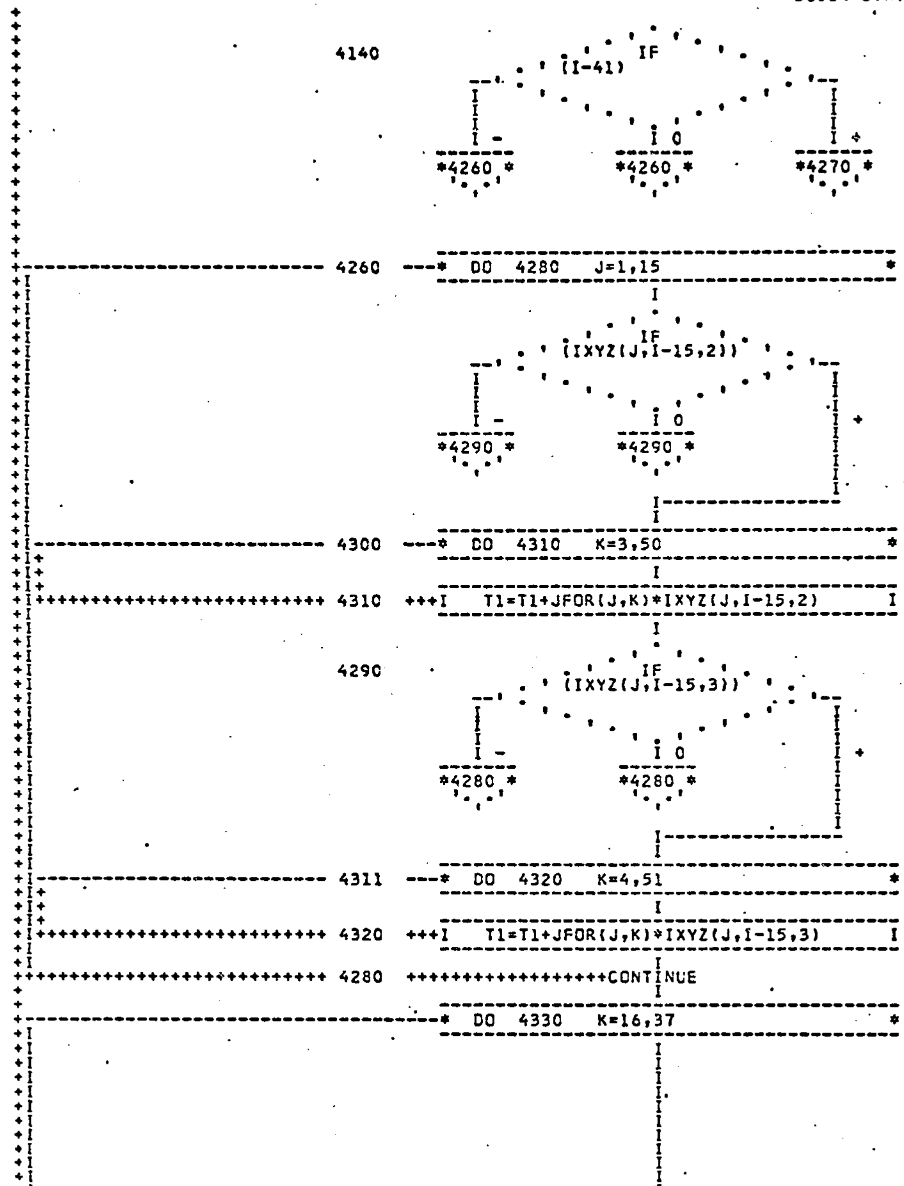

IF
(IXYZ(J,I-15,1))
*4080 *
*4080 *
4090
DO 4100 K=2,49
4100
T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)
T2=T2+DEC(J)*IXM1(J,I-15)
4080
CONTINUE
ISA(I,M)=T2
IF
(M-3)
*4081 *
*4081 *
4082
DO 4083 MM=1,M
4083
T4=T4+ISA(I,MM)
XM=T4/M
DO 4084 MM=1,M
4084
T3=T3+(XM-ISA(I,MM))**2
T3=T3/M1
4081
CONTINUE
IF
(IXI(I))
*4061 *
*4061 *


DCID4 STAT STD DEV PAGE NO 4
4062 IQC(I)=SQRT((2.*T1*S(I)*WR)/((V(I) /FLOAT(IXI(I)))*48.*XIC))+0.5
4061 ISS(I)=ZY*SQRT(T3)+0.5
DO 4065 J=1,15
IF (IXM1(J,I-15))
4065
4065
4066
IF (ICQ(J)-IQC(I))
4065
4065
4067 IQC(I)=IQC(J)
4065 CONTINUE
IF (IXI(I)-T2-ISS(I))
4110
4110
4120
4110 DEC(I)=IQC(I)+ISS(I)-IXI(I)+T2
4000
412C DEC(I)=0.

```
+                                          I
+                                      ---------
+                                      * 4000  *
+                                        *   *
+                                          *
+
+                                          .
+                     4070            . (I-37)  IF  .
+                                --.                   .--
+                                I        .       .      I
+                                I           .           I
+                                I -         I 0         I +
+                              -------    -------     -------
+                              *4130 *    *4130 *     *4140 *
+                                 *          *           *
+
+------------------------- 4130 ---*  DO  4150   J=1,15                 *
+I                                 -------------------------------------
+I                                         I
+I                                         .
+I                                  . (IXYZ(J,I-15,1)) IF  .
+I                               --.                      .--
+I                               I      .          .         I
+I                               I           .               I
+I                               I -         I 0             I +
+I                             -------    -------            I
+I                             *4160 *    *4160 *            I
+I                                *          *               I
+I                                          I----------------
+I                                          I
+I------------------------- 4170 ---*  DO  4180   K=2,49                 *
+I+                                         I
+I+
+I+++++++++++++++++++++++++ 4180 +++I  T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)    I
+I                                         I
+I                                         .
+I                    4160          . (IXYZ(J,I-15,2)) IF  .
+I                               --.                      .--
+I                               I      .          .         I
+I                               I           .               I
+I                               I -         I 0             I +
+I                             -------    -------            I
+I                             *4150 *    *4150 *            I
+I                                *          *               I
+I                                          I----------------
+I                                          I
+I------------------------- 4190 ---*  DO  4210   K=3,50                 *
+I+                                         I
+I+
+I+++++++++++++++++++++++++ 4210 +++I  T1=T1+JFOR(J,K)*IXYZ(J,I-15,2)    I
+I                                         I
++++++++++++++++++++++++++++ 4150 +++++++++++++++++CONTINUE
+                                          I
+-----------------------------------*  DO  4220   K=1,22                 *
+I                                         I
+I                                         I
+I                                         I
```

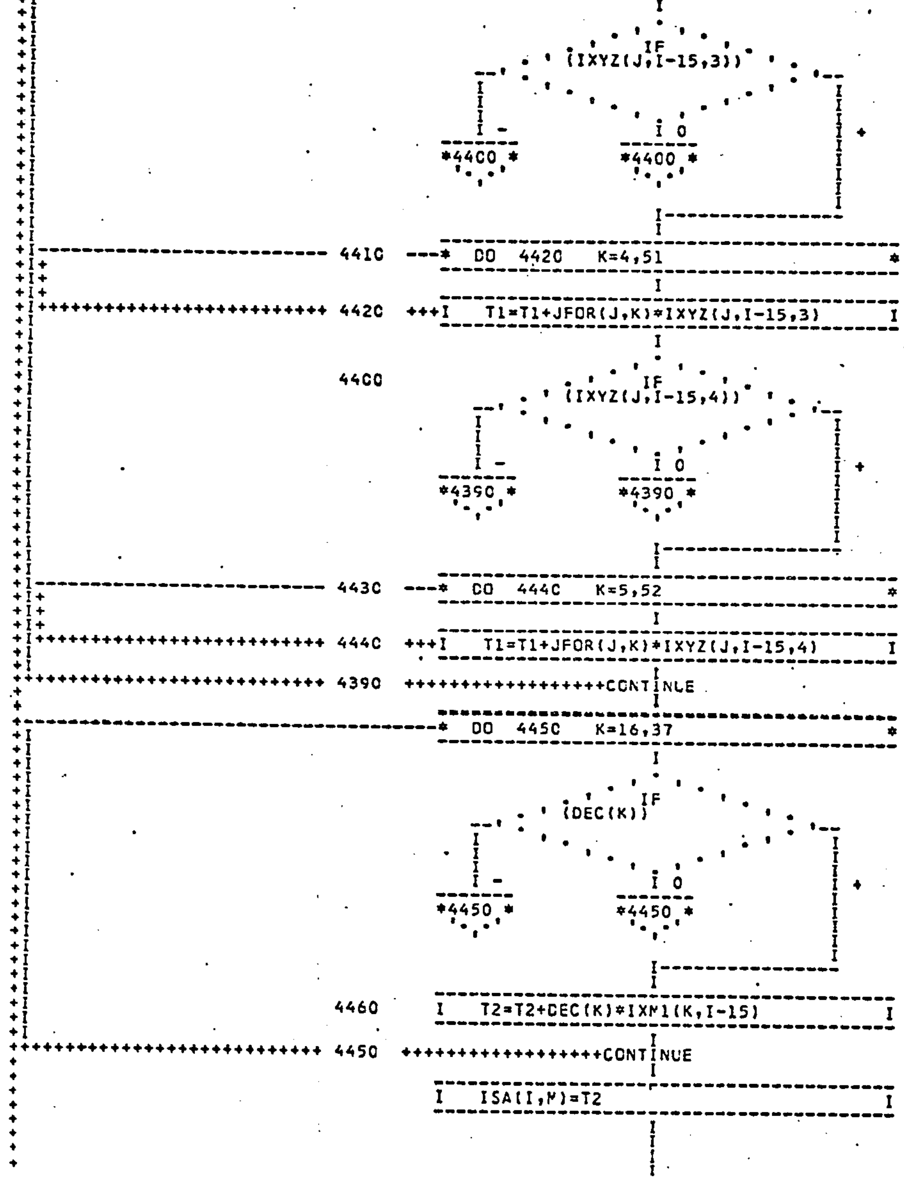
```
                                                    I
                                              IF
                                    (DEC(K))
                                  -          0          +
                               *4220 *    *4220 *
                                              I----------
               4230         I  T2=T2+DEC(K)*IXM1(K,I-15)            I
++++++++++++++ 4220  +++++++++++++++++CONTINUE
                            I  ISA(I,M)=T2                          I
                                              IF
                                    (M-3)
                                  -          0          +
                               *4231 *    *4231 *
                                              I----------
-------------- 4232  ---*  DO  4233   MM=1,M                        *
++++++++++++++ 4233  +++I  T4=T4+ISA(I,MM)                          I
                        I  XM=T4/M                                  I
--------------------------*  DO  4234   MM=1,M                      *
++++++++++++++ 4234  +++I  T3=T3+(XM-ISA(I,MM))**2                  I
                        I  T3=T3/M1                                 I
               4231             CONTINUE
                                              IF
                                    (IXI(I))
                                  -          0          +
                               *4221 *    *4221 *
                                              I----------
               4222         I. IQQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)   I
                            I        /FLOAT(IXI(I)))*48.*XIC))+0.5  I
```

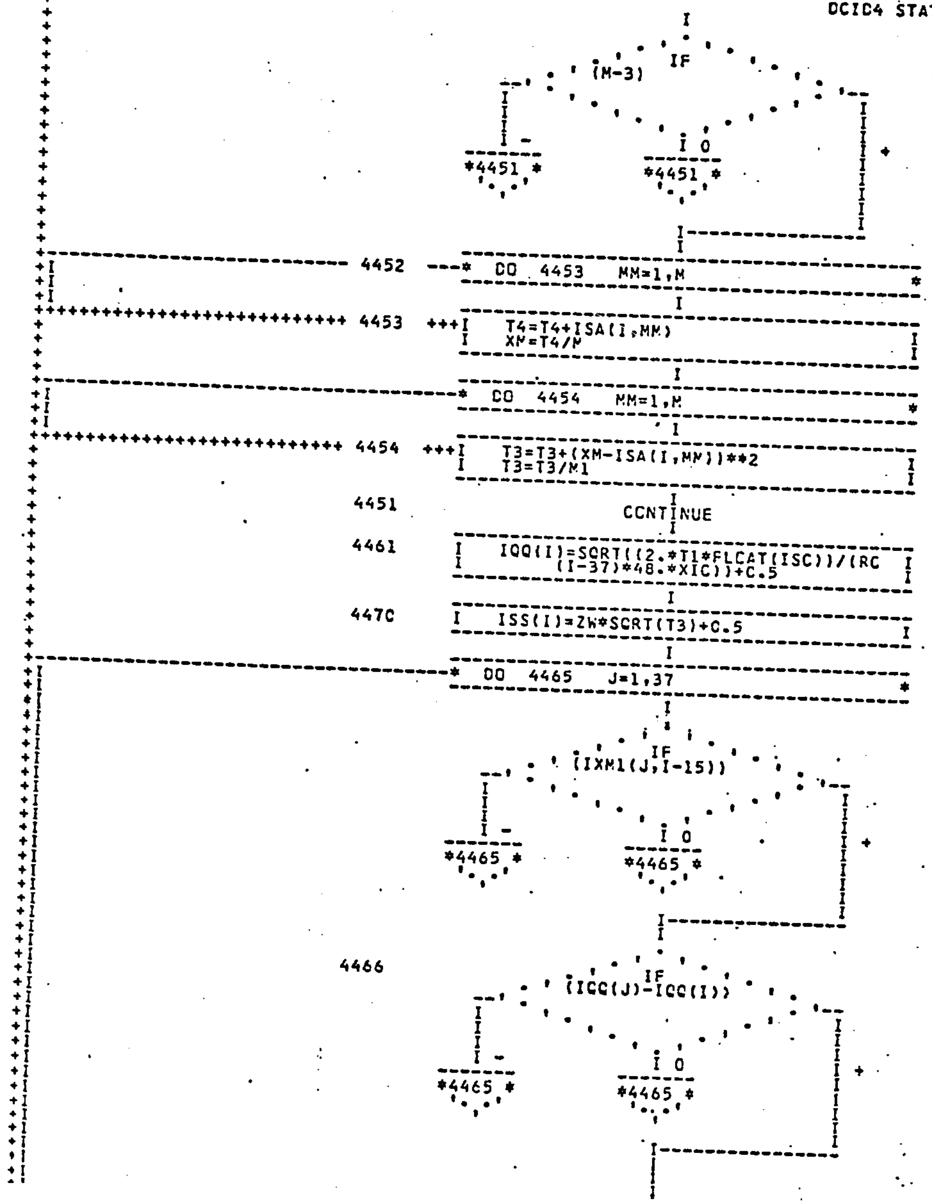

4221
ISS(I)=ZX*SQRT(T3)+0.5
DO 4225 J=1,22
IF (IXM1(J,I-15))
- *4225*
0 *4225*
+
4226
IF (IQQ(J)-IQC(I))
- *4225*
0 *4225*
+
4227
IQC(I)=IQQ(J)
4225
CONTINUE
IF (IXI(I)-T2-ISS(I))
- *4240*
0 *4240*
+ *4250*
4240
DEC(I)=IQQ(I)+ISS(I)-IXI(I)+T2
* 4000 *
4250
DEC(I)=0.
* 4000 *


4140
IF (I-41)
- 0 +
4260 4260 4270
4260 DO 4280 J=1,15
IF (IXYZ(J,I-15,2))
- 0 +
4290 4290
4300 DO 4310 K=3,50
4310 T1=T1+JFOR(J,K)*IXYZ(J,I-15,2)
4290
IF (IXYZ(J,I-15,3))
- 0 +
4280 4280
4311 DO 4320 K=4,51
4320 T1=T1+JFOR(J,K)*IXYZ(J,I-15,3)
4280 CONTINUE
DO 4330 K=16,37

```
                                  IF
                               (DEC(K))
                     I -               I 0                    +
                   *4330 *           *4330 *

 4340                I   T2=T2+DEC(K)*IXM1(K,I-15)             I
 4330          +++++++++++++++++CONTINUE
                     I   ISA(I,M)=T2                           I

                                  IF
                                (M-3)
                     I -               I 0                    +
                   *4341 *           *4341 *

 4342          ---*  DO  4343    MM=1,M                        *
 4343          +++I   T4=T4+ISA(I,MM)                          I
                  I   XM=T4/M                                  I
                  *  DO  4344    MM=1,M                        *
 4344          +++I   T3=T3+(XM-ISA(I,MM))**2                  I
                  I   T3=T3/M1                                 I
 4341                         CONTINUE
 4360                I   IQQ(I)=SQRT((2.*T1*FLCAT(ISC))/(RC     I
                     I        (I-37)*48.*XIC))+0.5             I
 435C                I   ISS(I)=ZW*SQRT(T3)+0.5                 I
                  *  DO  4365    J=1,37                        *
```

```
                                                        DCID4 STAT STD DEV   PAGE NO   10
+I                                      I
+I                                  IF
+I                          (IXM1(J,I-15))
+I                 I -           I 0          I +
+I              *4365 *       *4365 *         I
+I                                   I---------
+I
+I      4366                        IF
+I                          (IQQ(J)-IQQ(I))
+I                 I -           I 0          I +
+I              *4365 *       *4365 *         I
+I                                   I---------
+I      4367    I   IQQ(I)=IQQ(J)                      I
++++++++++++ 4365 ++++++++++++++++CONTINUE
+                                   IF
+                           (IXI(I)-T2-ISS(I)
+                  I -           I 0           I +
+               *4370 *       *4370 *       *4380 *
+
+       4370    I   DEC(I)=IQQ(I)+ISS(I)-IXI(I)+T2     I
+                                I
+                            * 4000 *
+
+       4380    I   DEC(I)=0.                          I
+                                I
+                            * 4000 *
+
+------------ 4270 ---*  DO  4390    J=1,15             *
+I                               I
+I                               I
```


IF (IXYZ(J,I-15,3))
-
0
+
4400
4400
4410
DO 4420 K=4,51
4420
T1=T1+JFOR(J,K)*IXYZ(J,I-15,3)
4400
IF (IXYZ(J,I-15,4))
-
0
+
4390
4390
4430
DO 4440 K=5,52
4440
T1=T1+JFOR(J,K)*IXYZ(J,I-15,4)
4390
CONTINUE
DO 4450 K=16,37
IF (DEC(K))
-
0
+
4450
4450
4460
T2=T2+DEC(K)*IXM1(K,I-15)
4450
CONTINUE
ISA(I,M)=T2


IF (M-3)
*4451 *
*4451 *
4452 DO 4453 MM=1,M
4453 T4=T4+ISA(I,MM)
XM=T4/M
DO 4454 MM=1,M
4454 T3=T3+(XM-ISA(I,MM))**2
T3=T3/M1
4451 CONTINUE
4461 IQQ(I)=SQRT((2.*T1*FLOAT(ISC))/(RC
(I-37)*48.*XIC))+C.5
447C ISS(I)=ZW*SQRT(T3)+C.5
DO 4465 J=1,37
IF (IXM1(J,I-15))
*4465 *
*4465 *
4466 IF (ICQ(J)-ICQ(I))
*4465 *
*4465 *

```
4467  I  IQQ(I)=IQQ(J)                                  I
4465  ++++++++++++++++++CONTINUE

          IF
      (IXI(I)+ICC(I-37)
      -T2-ISS(I))
       -            0             +
   *4480 *      *4480 *       *4490 *

4480  I  DEC(I)=IQQ(I)+ISS(I)-IXI(I)-ICC(I-            I
      I        37)+T2                                  I
                  * 4000 *

4490  I  DEC(I)=0.                                     I
4000  ++++++++++++++++++CONTINUE

          IF
        (M-3)
       -            0             +
   *4550 *      *4550 *       *4560 *

4550  ---*  DO  4570    I=1,45                         *
4570  +++I  DEC(I)=IQQ(I)                              I
4560        CONTINUE
      -----*  DO  20      ID=1,3                       *
      I  TCT=0.                                        I
      I  IL=IBMR(ID)+1                                 I
      I  IU=IBMR(ID)+IBN(ID)                           I
      -----*  DO  61      IP=IL,IU                     *
```

```
                                                                          DCIC4 STAT STD DEV   PAGE NO   14
+I                                                          I
+I                                                   . ' . ' . ' . .
+I                                                 . '     IF      ' .
+I                                              --' : ' (DEC(IP))    : '--
+I                                              I   ' .           . '   I
+I                                              I      ' . ' . ' .      I
+I                                              I -          I 0         I +
+I                                            -------      -------       I
+I                                            *61   *      *61   *       I
+I                                              '.'          '.'         I
+I                                                          I------------I
+I                                                          I
+I                        62    ----------------------------------------------
+I                              I   TOT=TOT+S(IP)+DEC(IP)*U(IP)              I
+I                              ----------------------------------------------
+I                                                          I
++++++++++++++++++++++++++++ 61  +++++++++++++++++++CONTINUE
+                                                          I
+                                ----------------------------------------------
++++++++++++++++++++++++++++ 20  +++I   DEC(ID+45)=IFIX(TOT/40.+0.5)          I
                                 ----------------------------------------------
                                                           I
                                                   ------------------
                                                   I RETURN          '.
                                                   I                   :
                                                   I                 .'
                                                   ------------------
***END*******END*******END*******END*******END*******END*******END*******END*******END*******END*******END***
```

APPENDIX VII

THE STATISTICAL SERVICE LEVELS MODEL

```
      SUBROUTINE DECIDE                                                 DCID5000
      INTEGER WSTAR                                                     DCID5001
      COMMON IXM1(37,30),S(37),U(37),IXI(45),V(45),STH(3),OTH(3),SUH(3),DCID5002
     1XIH(3),IBMR(3),IBN(3),IXLW(3),ISFG(15),IUS(15),IBO(15),ISL(15),  DCID5003
     2ITD(15),ITB(15),ITSL(15),AS(15),BS(15),CS(15),IFOR(15,6),CCSTY(5),DCID5004
     3COST(5),COSTT,SHAT(12),ALPHA(15),BETA(15),XMU(15),SIG(15),FGP(15),DCID5005
     4XMUX(3),SIGX(3),ICO(8),KC(8),E(15),A(15),B(15),IXM(45),DEC(48),  DCID5006
     5AP(45),ITI(45),TV(45),TSTH(3),TOTH(3),TSUH(3),TRH(3),RH(3),IC(45) DCID5007
      COMMON STAR,WSTAR,PSTAR,WR,XMOT,IHC,ITC,IFC,XIC,IBC,ISC,N,MMAX,M  DCID5008
      COMMON COSTM,TIH(3),JFOR(45,18),JSE(45,18),ZZ,ZY,ZX,ZW            DCID5009
      COMMON IXYZ(15,30,4),IQQ(45),ISS(45),ISA(45,72),STAR5             DCID5010
      COMMON PZ,PY,PX,PW,SIGMA(45),EX(400)                              DCID5011
C        DCID5 - STATISTICAL WITH SERVICE LEVELS                        DCID5012
      M1=M-1                                                            DCID5013
      DO 4000 I=1,45                                                    DCID5014
      T1=0.                                                             DCID5015
      T2=0.                                                             DCID5016
      T3=0.                                                             DCID5017
      T4=0.                                                             DCID5018
      DEC(I)=0.                                                         DCID5019
      IF(I-15)4010,4010,4020                                            DCID5020
 4010 T1=0.                                                             DCID5021
      DO 4030 K=1,12                                                    DCID5022
 4030 T1=T1+JFOR(I,K)                                                   DCID5023
      IF(IXI(I))4031,4031,4032                                          DCID5024
 4032 IQQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)/FLOAT(IXI(I)))*12.*XIC))+0.5   DCID5025
 4031 IF(IQQ(I)-JFOR(I,1))4033,4033,4034                                DCID5026
 4033 IQQ(I)=JFOR(I,1)                                                  DCID5027
 4034 CONTINUE                                                          DCID5028
      T3=JSE(I,1)                                                       DCID5029
      IF(T3)4035,4035,4039                                              DCID5030
 4039 CONTINUE                                                          DCID5031
      PEXP=IQQ(I)*(1.-PZ)/T3                                            DCID5032
      IF(PEXP-EX(1))4036,4038,4035                                      DCID5033
 4036 CALL PAREXP(PEXP,Z)                                               DCID5034
      GO TO 4037                                                        DCID5035
 4038 Z=0.01                                                            DCID5036
      GO TO 4037                                                        DCID5037
 4035 Z=0.                                                              DCID5038
 4037 ISS(I)=Z*T3+0.5                                                   DCID5039
      SIGMA(I)=Z                                                        DCID5040
      IF(IXI(I)-JFOR(I,1)-ISS(I))4040,4050,4050                         DCID5041
 4040 DEC(I)=IQQ(I)+IBC(I)+ISS(I)-IXI(I)                                DCID5042
      GO TO 4000                                                        DCID5043
 4050 DEC(I)=0.                                                         DCID5044
      GO TO 4000                                                        DCID5045
 4020 IF(I-22)4060,4060,4070                                            DCID5046
 4060 DO 4080 J=1,15                                                    DCID5047
      IF(IXYZ(J,I-15,1))4080,4080,4090                                  DCID5048
 4090 DO 4100 K=2,13                                                    DCID5049
 4100 T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)                                    DCID5050
      T2=T2+DEC(J)*IXM1(J,I-15)                                         DCID5051
 4080 CONTINUE                                                          DCID5052
      ISA(I,M)=T2                                                       DCID5053
      IF(M-3)4081,4081,4082                                             DCID5054
 4082 DO 4083 MM=1,M                                                    DCID5055
 4083 T4=T4+ISA(I,MM)                                                   DCID5056
      XM=T4/M                                                           DCID5057
      DO 4084 MM=1,M                                                    DCID5058
 4084 T3=T3+(XM-ISA(I,MM))**2                                           DCID5059
```

```
      T3=T3/M1                                                          DCID5060
 4081 CONTINUE                                                          DCID5061
      IF(IXI(I))4061,4061,4062                                          DCID5062
 4062 IQQ(I)=SCRT((2.*T1*S(I)*WR)/((V(I)/FLOAT(IXI(I)))*12.*XIC))+0.5  DCID5063
 4061 DO 4065 J=1,15                                                    DCID5064
      IF(IXM1(J,I-15))4065,4065,4066                                    DCID5065
 4066 IF(ICC(J)-ICC(I))4065,4065,4067                                   DCID5066
 4067 IQQ(I)=ICC(J)                                                     DCID5067
 4065 CONTINUE                                                          DCID5068
      IF(T3)4075,4075,4078                                              DCID5069
 4078 T3=SCRT(T3)                                                       DCID5070
      PEXP=ICC(I)*(1.-PY)/T3                                            DCID5071
      IF(PEXP-EX(1))4076,4079,4075                                      DCID5072
 4076 CALL PAREXP(PEXP,Z)                                               DCID5073
      GO TO 4077                                                        DCID5074
 4079 Z=0.01                                                            DCID5075
      GO TO 4077                                                        DCID5076
 4075 Z=0.                                                              DCID5077
 4077 ISS(I)=Z*T3+0.5                                                   DCID5078
      SIGMA(I)=Z                                                        DCID5079
      IF(IXI(I)-T2-ISS(I))4110,4110,4120                                DCID5080
 4110 DEC(I)=ICC(I)+ISS(I)-IXI(I)+T2                                    DCID5081
      GO TO 4000                                                        DCID5082
 4120 DEC(I)=0.                                                         DCID5083
      GO TO 4000                                                        DCID5084
 4070 IF(I-37)4130,4130,4140                                            DCID5085
 4130 DO 4150 J=1,15                                                    DCID5086
      IF(IXYZ(J,I-15,1))4160,4160,4170                                  DCID5087
 4170 DO 4180 K=2,13                                                    DCID5088
 4180 T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)                                    DCID5089
 4160 IF(IXYZ(J,I-15,2))4150,4150,4190                                  DCID5090
 4190 DO 4210 K=3,14                                                    DCID5091
 4210 T1=T1+JFOR(J,K)*IXYZ(J,I-15,2)                                    DCID5092
 4150 CONTINUE                                                          DCID5093
      DO 4220 K=1,22                                                    DCID5094
      IF(DEC(K))4220,4220,4230                                          DCID5095
 4230 T2=T2+DEC(K)*IXM1(K,I-15)                                         DCID5096
 4220 CONTINUE                                                          DCID5097
      ISA(I,M)=T2                                                       DCID5098
      IF(M-3)4231,4231,4232                                             DCID5099
 4232 DO 4233 MM=1,M                                                    DCID5100
 4233 T4=T4+ISA(I,MM)                                                   DCID5101
      XM=T4/M                                                           DCID5102
      DO 4234 MM=1,M                                                    DCID5103
 4234 T3=T3+(XM-ISA(I,MM))**2                                           DCID5104
      T3=T3/M1                                                          DCID5105
 4231 CONTINUE                                                          DCID5106
      IF(IXI(I))4221,4221,4222                                          DCID5107
 4222 IQQ(I)=SCRT((2.*T1*S(I)*WR)/((V(I)/FLOAT(IXI(I)))*12.*XIC))+0.5  DCID5108
 4221 DO 4225 J=1,22                                                    DCID5109
      IF(IXM1(J,I-15))4225,4225,4226                                    DCID5110
 4226 IF(ICC(J)-ICC(I))4225,4225,4227                                   DCID5111
 4227 ICQ(I)=ICC(J)                                                     DCID5112
 4225 CONTINUE                                                          DCID5113
      IF(T3)4275,4275,4278                                              DCID5114
 4278 T3=SCRT(T3)                                                       DCID5115
      PEXP=ICC(I)*(1.-PX)/T3                                            DCID5116
      IF(PEXP-EX(1))4276,4279,4275                                      DCID5117
 4276 CALL PAREXP(PEXP,Z)                                               DCID5118
      GO TO 4277                                                        DCID5119
```

```
 4279 Z=0.01
      GO TO 4277
 4275 Z=0.
 4277 ISS(I)=Z*T3+0.5
      SIGMA(I)=Z
      IF(IXI(I)-T2-ISS(I))4240,4240,4250
 4240 DEC(I)=IQQ(I)+ISS(I)-IXI(I)+T2
      GO TO 4000
 4250 DEC(I)=0.
      GO TO 4000
 4140 IF(I-41)4260,4260,4270
 4260 DO 4280 J=1,15
      IF(IXYZ(J,I-15,2))4290,4290,4300
 4300 DO 4310 K=3,14
 4310 T1=T1+JFCR(J,K)*IXYZ(J,I-15,2)
 4290 IF(IXYZ(J,I-15,3))4280,4280,4311
 4311 DO 4320 K=4,15
 4320 T1=T1+JFCR(J,K)*IXYZ(J,I-15,3)
 4280 CONTINUE
      DO 4330 K=16,37
      IF(DEC(K))4330,4330,4340
 4340 T2=T2+DEC(K)*IXM1(K,I-15)
 4330 CONTINUE
      ISA(I,M)=T2
      IF(M-3)4341,4341,4342
 4342 DO 4343 MM=1,M
 4343 T4=T4+ISA(I,MM)
      XM=T4/M
      DO 4344 MM=1,M
 4344 T3=T3+(XM-ISA(I,MM))**2
      T3=T3/M1
 4341 CONTINUE
 4360 IQQ(I)=SQRT((2.*T1*FLOAT(ISC))/(RC(I-37)*12.*XIC))+0.5
 4350 DO 4365 J=1,37
      IF(IXM1(J,I-15))4365,4365,4366
 4366 IF(ICC(J)-ICC(I))4365,4365,4367
 4367 ICC(I)=ICC(J)
 4365 CONTINUE
      IF(T3)4355,4355,4358
 4358 T3=SQRT(T3)
      PEXP=ICC(I)*(1.-PW)/T3
      IF(PEXP-EX(1))4356,4359,4355
 4356 CALL PAREXP(PEXP,Z)
      GO TO 4357
 4359 Z=0.01
      GO TO 4357
 4355 Z=0.
 4357 ISS(I)=Z*T3+0.5
      SIGMA(I)=Z
      IF(IXI(I)-T2-ISS(I))4370,4370,4380
 4370 DEC(I)=ICC(I)+ISS(I)-IXI(I)+T2
      GO TO 4000
 4380 DEC(I)=0.
      GO TO 4000
 4270 DO 4390 J=1,15
      IF(IXYZ(J,I-15,3))4400,4400,4410
 4410 DO 4420 K=4,15
 4420 T1=T1+JFCR(J,K)*IXYZ(J,I-15,3)
 4400 IF(IXYZ(J,I-15,4))4390,4390,4430
 4430 DO 4440 K=5,16
```

```
4440  T1=T1+JFOR(J,K)*IXYZ(J,I-15,4)                                    DCIC5180
4390  CONTINUE                                                          DCIC5181
      DO 4450 K=16,37                                                   DCIC5182
      IF(DEC(K))4450,4450,4460                                          DCIC5183
4460  T2=T2+DEC(K)*IXM1(K,I-15)                                         DCIC5184
4450  CONTINUE                                                          DCIC5185
      ISA(I,M)=T2                                                       DCIC5186
      IF(M-3)4451,4451,4452                                             DCIC5187
4452  DO 4453 MM=1,M                                                    DCIC5188
4453  T4=T4+ISA(I,MM)                                                   DCIC5189
      XM=T4/M                                                           DCIC5190
      DO 4454 MM=1,M                                                    DCIC5191
4454  T3=T3+(XM-ISA(I,MM))**2                                           DCIC5192
      T3=T3/M1                                                          DCIC5193
4451  CONTINUE                                                          DCIC5194
4461  IOQ(I)=SQRT((2.*T1*FLOAT(ISC))/(RC(I-37)*12.*XIC))+0.5            DCIC5195
4470  DO 4465 J=1,37                                                    DCIC5196
      IF(IXM1(J,I-15))4465,4465,4466                                    DCIC5197
4466  IF(ICC(J)-ICC(I))4465,4465,4467                                   DCIC5198
4467  IQQ(I)=ICC(J)                                                     DCIC5199
4465  CONTINUE                                                          DCIC5200
      IF(T3)4475,4475,4478                                              DCIC5201
4478  T3=SQRT(T3)                                                       DCIC5202
      PEXP=IQQ(I)*(1.-PW)/T3                                            DCIC5203
      IF(PEXP-EX(1))4476,4479,4475                                      DCIC5204
4476  CALL PAREXP(PEXP,Z)                                               DCIC5205
      GO TO 4477                                                        DCIC5206
4479  Z=0.01                                                            DCIC5207
      GO TO 4477                                                        DCIC5208
4475  Z=0.                                                              DCIC5209
4477  ISS(I)=Z*T3+0.5                                                   DCIC5210
      SIGMA(I)=Z                                                        DCIC5211
      IF(IXI(I)+ICC(I-37)-T2-ISS(I))4480,4480,4490                      DCIC5212
4480  DEC(I)=ICC(I)+ISS(I)-IXI(I)-IOO(I-37)+T2                          DCIC5213
      GO TO 4000                                                        DCIC5214
4490  DEC(I)=0.                                                         DCIC5215
4000  CONTINUE                                                          DCIC5216
      IF(M-3)4550,4550,4560                                             DCIC5217
4550  DO 4570 I=1,45                                                    DCIC5218
4570  DEC(I)=ICC(I)                                                     DCIC5219
4560  CONTINUE                                                          DCIC5220
      DO 20 IC=1,3                                                      DCIC5221
      TOT=0.                                                            DCIC5222
      IL=IBMR(IC)+1                                                     DCIC5223
      IU=IBMR(IC)+IBN(IC)                                               DCIC5224
      DO 61 IP=IL,IU                                                    DCIC5225
      IF(DEC(IP))61,61,62                                               DCIC5226
62    TOT=TOT+S(IP)+DEC(IP)*U(IP)                                       DCIC5227
61    CONTINUE                                                          DCIC5228
20    DEC(IC+45)=IFIX(TOT/160.+0.5)                                     DCIC5229
      RETURN                                                            DCIC5230
      END                                                               DCIC5231
```

```
                                  DCID5 STAT SER LEVEL PAGE NO    1
***       DCID5 - STATISTICAL WITH SERVICE LEVELS                            ***

          I.  M1=M-1                                    I          DCID5C11
          * DO  4000   I=1,45                           *          DCID5C12
          I   T1=C.                                     I          DCID5C13
          I   T2=0.                                     I          DCID5C14
          I   T3=C.                                     I          DCID5C15
          I   T4=0.                                     I          DCID5C16
          I   DEC(I)=0.                                 I          DCID5C17
                       IF (I-15)                                   DCID5C18
               -             0             +
           *4010 *       *4010 *       *4020 *

4010      I   T1=0.                                     I          DCID5C19
          * DO  4030   K=1,12                           *          DCID5C20
4030  +++I   T1=T1+JFOR(I,K)                            I          DCID5C21
                       IF (IXI(I))                                 DCID5C22
               -             C             +
           *4031 *       *4031 *

4032      I   IQQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)        I          DCID5C23
          I      /FLOAT(IXI(I)))*12.*XIC))+0.5           I
4031                   IF (IQQ(I)-JFOR(I,1))                       DCID5C25
               -             0             +
           *4033 *       *4033 *       *4034 *
```

```
                                DCIDS STAT SER LEVEL PAGE NO   2
4033   I   IQQ(I)=JFOR(I,1)                    I               DCID5C26
4034              CONTINUE                                     DCID5C27
       I   T3=JSE(I,1)                         I
                     IF
              (T3)
         I -           I 0           I +
       *4035 *       *4035 *
4039              CONTINUE
       I   PEXP=ICC(I)*(1.-PZ)/T3              I
                     IF
              (PEXP-EX(1))
         I -           I 0           I +
       *4036 *       *4038 *       *4035 *
4036          CALL PAREXP(PEX
              P,Z)
                    * 4037 *
4038   I   Z=0.01                              I
                    * 4037 *
4035   I   Z=C.                                I
4037   I   ISS(I)=Z*T3+0.5                     I
       I   SIGMA(I)=Z                          I
```

```
                              IF
                   (IXI(I)-JFOR(I,1)                          DCID5C28
                   -ISS(I))
          -                 0                  +
       *4040 *          *4050 *            *4050 *

4040   I  DEC(I)=IQQ(I)+IBC(I)+ISS(I)-IXI(I)  I               DCID5C29
                          I
                      * 4000 *                                DCID5C30

4050   I  DEC(I)=0.                           I               DCID5C31
                          I
                      * 4000 *                                DCID5C32

4020                      IF                                  DCID5C33
             (I-22)
          -                 0                  +
       *4060 *          *4060 *            *4070 *

4060   ---*  DO  4080   J=1,15                *               DCID5C34
                          I
                          IF                                  DCID5C35
             (IXYZ(J,I-15,1))
          -                 0                  +
       *4080 *          *4080 *

4090   ---*  DO  4100   K=2,13                *               DCID5C36
                          I
4100   +++I  T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)   I               DCID5C37
          I  T2=T2+DEC(J)*IXM1(J,I-15)        I               DCID5C38

4080   +++++++++++++++++CONTINUE                              DCID5C39
```

```
                 ISA(I,M)=T2                                   DCIDSC40

                 IF (M=3)                                      DCIDSC41
                   -  *4081*     0  *4081*     +

      4082       DO  4083   MM=1,M                             DCIDSC42

      4083       T4=T4+ISA(I,MM)                               DCIDSC43
                 XM=T4/M                                       DCIDSC44

                 DO  4084   MM=1,M                             DCIDSC45

      4084       T3=T3+(XM-ISA(I,MM))**2                       DCIDSC46
                 T3=T3/M1                                      DCIDSC47

      4081       CONTINUE                                      DCIDSC48

                 IF (IXI(I))                                   DCIDSC49
                   -  *4061*     0  *4061*     +

      4062       IQQ(I)=SQRT((2.*T1*S(I)*HR)/((V(I)            DCIDSC50
                      /FLOAT(IXI(I)))*12.*XIC))+0.5

      4061       DO  4065   J=1,15                             DCIDSC52

                 IF (IXM1(J,I-15))                             DCIDSC53
                   -  *4065*     0  *4065*     +
```

DCIDS STAT SER LEVEL PAGE NO. 5

```
4066      IF (IQQ(J)-IQQ(I))                                        DCIDSC54
            -  -> *4065 *
            0  -> *4065 *
            +  -> 4067

4067      I   ICQ(I)=IQQ(J)                                  I      DCIDSC55

4065      ++++++++++++++++++CONTINUE                                 DCIDSC56

          IF (T3)
            -  -> *4075 *
            0  -> *4075 *
            +  -> 4078

4078      I   T3=SQRT(T3)                                    I
          I   PEXP=ICQ(I)*(1.-PY)/T3                         I

          IF (PEXP-EX(1))
            -  -> *4076 *
            0  -> *4079 *
            +  -> *4075 *

4076      CALL PAREXP(PEX
          P,Z)
               -> * 4077 *

4079.     I   Z=C.01                                         I
               -> * 4077 *
```

329

DCID5 STAT SER LEVEL PAGE NO 6

```
4075   I  Z=0.                                          I
                              I
4077   I  ISS(I)=Z*T3+0.5                               I
       I  SIGMA(I)=Z                                    I
                              I
                             IF
                 (IXI(I)-T2-ISS(I))                          DCID5C57
             I -              I 0             I +
          *4110 *          *4110 *         *4120 *

4110   I  DEC(I)=IQC(I)+ISS(I)-IXI(I)+T2                I    DCID5C58
                              I
                          *  4000  *                         DCID5C59

4120   I  DEC(I)=0.                                     I    DCID5C60
                              I
                          *  4000  *                         DCID5C61

4070                     IF
                 (I-37)                                      DCID5C62
             I -              I 0             I +
          *4130 *          *4130 *         *4140 *

4130  ---*  DO  4150    J=1,15                          *    DCID5C63
                              I
                             IF
                 (IXYZ(J,I-15,1))                            DCID5C64
             I -              I 0             I +
          *4160 *          *4160 *
                              I
4170  ---*  DO  4180    K=2,13                          *    DCID5C65
```

DCIDS STAT SER LEVEL PAGE NO. 7

```
4180   +++I   T1=T1+JFOR(J,K)*IXYZ(J,I-15,1)        I      DCID5C66

4160              IF
              (IXYZ(J,I-15,2))                             DCID5C67
         -                 0               +
     *4150 *           *4150 *

4190   ---*  DO  4210   K=3,14                      *      DCID5C68
4210   +++I   T1=T1+JFOR(J,K)*IXYZ(J,I-15,2)        I      DCID5C69
4150   ++++++++++++++++++CONTINUE                          DCID5C70
       ---*  DO  4220   K=1,22                      *      DCID5C71

                  IF
              (DEC(K))                                     DCID5C72
         -                 0               +
     *4220 *           *4220 *

4230      I   T2=T2+DEC(K)*IXM1(K,I-15)             I      DCID5C73
4220   ++++++++++++++++++CONTINUE                          DCID5C74
          I   ISA(I,M)=T2                           I      DCID5C75

                  IF
              (M-3)                                        DCID5C76
         -                 0               +
     *4231 *           *4231 *

4232   ---*  DO  4233   MM=1,M                      *      DCID5C77
```

```
4233   T4=T4+ISA(I,MM)                                   DCID5C78
       XM=T4/M                                           DCID5C79
       DO 4234  MM=1,M                                   DCID5C80
4234   T3=T3+(XM-ISA(I,MM))**2                           DCID5C81
       T3=T3/M1                                          DCID5C82
4231   CONTINUE                                          DCID5C83
       IF (IXI(I))   - 4221, 0 4221, +                   DCID5C84
4222   ICQ(I)=SQRT((2.*T1*S(I)*WR)/((V(I)                DCID5C85
            /FLOAT(IXI(I)))*12.*XIC))+0.5
4221   DO 4225  J=1,22                                   DCID5C87
       IF (IXM1(J,I-15))   - 4225, 0 4225, +             DCID5C88
4226   IF (IQQ(J)-IQQ(I))   - 4225, 0 4225, +            DCID5C89
4227   IQC(I)=IQC(J)                                     DCID5C90
4225   CONTINUE                                          DCID5C91
```

```
                         I
                       IF
                 (T3)
          --                        --
          I                          I
          I -          I 0           I +
       *4275 *      *4275 *          I
                                     I
                       I-------------I
                       I
4278   ----------------------------------------
       I   T3=SQRT(T3)                        I
       I   PEXP=ICO(I)*(1.-PX)/T3             I
       ----------------------------------------
                       I
                       IF
                 (PEXP-EX(1))
          --                        --
          I                          I
          I -          I 0           I +
       *4276 *      *4279 *       *4275 *

4276         ---------------------
            CALL PAREXP(PEX
            P,Z)
             ---------------------
                       I
                    * 4277 *

4279   ----------------------------------------
       I   Z=C.01                             I
       ----------------------------------------
                       I
                    * 4277 *

4275   ----------------------------------------
       I   Z=C.                               I
       ----------------------------------------
                       I
4277   ----------------------------------------
       I   ISS(I)=Z*T3+0.5                    I
       I   SIGMA(I)=Z                         I
       ----------------------------------------
                       I
                       IF
                (IXI(I)-T2-ISS(I)
          --                        --
          I                          I
          I -          I C           I +
       *4240 *      *4240 *       *4250 *
```

DCID5C92

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
                                              CCIC5 STAT SER LEVEL PAGE NC .  1C

        4240   I  DEC(I)=IQC(I)+ISS(I)-IXI(I)+T2        I          CCID5C93
                              I
                           * 4000 *                                 CCID5C94

        4250   I  DEC(I)=C.                             I          CCID5C95
                              I
                           * 4000 *                                 CCID5C96

        4140                  IF                                    CCID5C97
                     (I-41)
                 I -          I 0           I +
               *4260 *      *4260 *      *4270 *

        4260  ---*  DO  4280    J=1,15                  *          CCID5C98
                              I
                              IF                                    CCID5C99
                     (IXYZ(J,I-15,2))
                 I -          I 0                   I +
               *4290 *      *4290 *
                              I

        4300  ---*  DO  4310    K=3,14                  *          CCID51C0
                              I
        4310  +++I   T1=T1+JFOR(J,K)*IXYZ(J,I-15,2)     I          CCID51C1
                              I
        4290                  IF                                    CCID51C2
                     (IXYZ(J,I-15,3))
                 I -          I 0                   I +
               *4280 *      *4280 *
                              I

        4311  ---*  DO  432C    K=4,15                  *          CCID51C3
```

```
4320  T1=T1+JFOR(J,K)*IXYZ(J,I-15,3)                          DCID5104
4280  CONTINUE                                                DCID5105
      DO 4330   K=16,37                                       DCID5106
      IF (DEC(K))                                             DCID5107
         -  *4330*     0  *4330*     +
4340  T2=T2+DEC(K)*IXM1(K,I-15)                               DCID5108
4330  CONTINUE                                                DCID5109
      ISA(I,M)=T2                                             DCID5110
      IF (M-3)                                                DCID5111
         -  *4341*     0  *4341*     +
4342  DO 4343   MM=1,M                                        DCID5112
4343  T4=T4+ISA(I,MM)                                         DCID5113
      XM=T4/M                                                 DCID5114
      DO 4344   MM=1,M                                        DCID5115
4344  T3=T3+(XM-ISA(I,MM))**2                                 DCID5116
      T3=T3/M1                                                DCID5117
4341  CONTINUE                                                DCID5118
4360  ICC(I)=SCRT((2.*T1*FLCAT(ISC))/(RC                      DCID5119
         (I-37)*12.*XIC))+0.5
4350  DO 4365   J=1,37                                        DCID5121
```

```
                      I
                   IF
              (IXM1(J,I-15))                                   DCID5122
       I -           I 0           I +
    *4365 *       *4365 *
                      I-------------
4366               IF
              (IQQ(J)-IQQ(I))                                  DCID5123
       I -           I 0           I +
    *4365 *       *4365 *
                      I-------------
4367  I   ICC(I)=IQQ(J)                            I           DCID5124
4365  ++++++++++++++++++CONTINUE                               DCID5125
                      I
                     IF
            (T3)
       I -           I 0           I +
    *4355 *       *4355 *
                      I-------------
4358  I   T3=SQRT(T3)                              I
      I   PEXP=ICQ(I)*(1.-PW)/T3                   I
                      I
                   IF
            (PEXP-EX(1))
       I -           I 0           I +
    *4356 *       *4359 *       *4355 *

4356        CALL PAREXP(PEX
            P,Z)
                      I
```

```
                          I
                      * 4357 *

4359    I   Z=0.01                                  I
                          I
                      * 4357 *

4355    I   Z=0.                                    I
                          I
4357    I   ISS(I)=Z*T3+0.5                         I
        I   SIGMA(I)=Z                              I
                          I
                          IF
                  (IXI(I)-T2-ISS(I)                       DCID5126
            I                              I
            I -           I 0              I +
        *4370 *       *4370 *          *4380 *

4370    I   DEC(I)=IQQ(I)+ISS(I)-IXI(I)+T2          I     DCID5127
                          I
                      * 4000 *                            DCID5128

4380    I   DEC(I)=0.                               I     DCID5129
                          I
                      * 4000 *                            DCID5130

4270 ---*  DO  4390    J=1,15                       *     DCID5131
                          I
                          IF
                  (IXYZ(J,I-15,3))                        DCID5132
            I                              I
            I -           I 0              I +
        *4400 *       *4400 *              I
                          I----------------I
                          I
```

```
4420  T1=T1+JFOR(J,K)*IXYZ(J,I-15,3)                  DCIC5134
4400  IF (IXYZ(J,I-15,4))                             DCIC5135
        -  *4390 *     0  *4390 *     +
4430  DO  4440   K=5,16                               DCIC5136
4440  T1=T1+JFOR(J,K)*IXYZ(J,I-15,4)                  DCIC5137
4390  CONTINUE                                        DCIC5138
      DO  4450   K=16,37                              DCIC5139
      IF (DEC(K))                                     DCIC5140
        -  *4450 *     0  *4450 *     +
4460  T2=T2+DEC(K)*IXM1(K,I-15)                       DCIC5141
4450  CONTINUE                                        DCIC5142
      ISA(I,M)=T2                                     DCIC5143
      IF (M-3)                                        DCIC5144
        -  *4451 *     0  *4451 *     +
4452  DO  4453   MM=1,M                               DCIC5145
```

```
4453   T4=T4+ISA(I,MM)                                   DCID5146
       XM=T4/M                                           DCID5147

       DO  4454   MM=1,M                                 DCID5148

4454   T3=T3+(XM-ISA(I,MM))**2                           DCID5149
       T3=T3/M1                                          DCID5150

4451   CONTINUE                                          DCID5151

4461   IQQ(I)=SQRT((2.*T1*FLOAT(ISC))/(RC                DCID5152
         (I-37)*12.*XIC))+0.5

4470   DO  4465   J=1,37                                 DCID5154

       IF (IXM1(J,I-15))                                 DCID5155
         -  *4465*     0  *4465*     +

4466   IF (IQQ(J)-IQQ(I))                                DCID5156
         -  *4465*     0  *4465*     +

4467   IQQ(I)=IQQ(J)                                     DCID5157

4465   CONTINUE                                          DCID5158

       IF (T3)
         -  *4475*     0  *4475*     +

4478   T3=SQRT(T3)
       PEXP=IQQ(I)*(1.-PW)/T3
```

```
                          I
                         IF
                   (PEXP-EX(1))
          I -             I 0              I +
        *4476 *         *4479 *          *4475 *

4476          CALL PAREXP(PEX
              P,Z)
                          I
                      * 4477 *

4479    I  Z=0.01                                   I
                          I
                      * 4477 *

4475    I  Z=0.                                     I
                          I
4477    I  ISS(I)=Z*T3+0.5                          I
        I  SIGMA(I)=Z                               I
                          I
                         IF
               (IXI(I)+ICC(I-37)                              DCID5159
               -T2-ISS(I))
          I -             I 0              I +
        *4480 *         *4480 *          *4490 *

4480    I  DEC(I)=IQC(I)+ISS(I)-IXI(I)-IDD(I-      I       DCID5160
        I      37)+T2                               I
                          I
                      * 4000 *                                DCID5161

449C    I  DEC(I)=C.                                I       DCID5162
4CCC    +++++++++++++++++CCNTINUE                           DCID5163
```

```
                                         I
                                   (M-3)  IF                                          DCID5164
                              I-                 I 0               I +
                            *4550 *            *4550 *           *4560 *

------------------ 4550  ---* DO  4570   I=1,45                        *           DCID5165
+                                          I
++++++++++++++++++ 4570  +++I  DEC(I)=IQC(I)                           I           DCID5166
                   4560              CONTINUE                                      DCID5167
------------------------------* DO  20      ID=1,3                     *           DCID5168
+                                          I
+                             I  TCT=0.                                I           DCID5169
+                             I  IL=IBPR(ID)+1                         I           DCID5170
+                             I  IU=IBPR(ID)+IBN(ID)                   I           DCID5171
+                                          I
+-----------------------------* DO  61      IP=IL,IU                   *           DCID5172
+I                                         I
+I                                     IF                                          DCID5173
+I                                 (DEC(IP))
+I                            I -          I 0                  I +
+I                           *61  *       *61  *                I
+I                                         I--------------------I
+I                 62          I  TCT=TCT+S(IP)+DEC(IP)*U(IP)           I           DCID5174
++++++++++++++++++ 61   +++++++++++++++++CONTINUE                                  DCID5175
+                                          I
++++++++++++++++++ 20   +++I  DEC(ID+45)=IFIX(TCT/160.+0.5)            I           DCID5176
                                           I
                                    I RETURN                                       DCID5177

***END*******END*******END*******END*******END*******END*******END*******END*******END*******END*******END***    DCID5178
```

```
      SUBROUTINE PAREXP(PEXP,Z)                                         ZPREX000
      INTEGER WSTAR                                                     ZPREX001
      COMMON IXM1(37,30),S(37),U(37),IXI(45),V(45),STH(3),OTH(3),SUH(3),ZPREX002
     1XIH(3),IBMR(3),IBN(3),IXLH(3),ISFG(15),IUS(15),IBU(15),ISL(15),   ZPREX003
     2ITD(15),ITB(15),ITSL(15),AS(15),BS(15),CS(15),IFOR(15,6),COSTY(5),ZPREX004
     3COST(5),COSTT,SHAT(12),ALPHA(15),BETA(15),XMU(15),SIG(15),FGP(15),ZPREX005
     4XMUX(3),SIGX(3),ICO(8),RC(8),E(15),A(15),B(15),IXM(45),DEC(48),   ZPREX006
     5AP(45),ITI(45),TV(45),TSTH(3),TOTH(3),TSUH(3),TRH(3),RH(3),IQ(45) ZPREX007
      COMMON STAR,WSTAR,PSTAR,WR,XMOT,IHC,ITC,IFC,XIC,IBC,ISC,N,MMAX,M  ZPREX008
      COMMON COSTM,TIH(3),JFOR(45,18),JSE(45,18),ZZ,ZY,ZX,ZW            ZPREX009
      COMMON IXYZ(15,30,4),ICQ(45),ISS(45),ISA(45,72),STAR5             ZPREX010
      COMMON PZ,PY,PX,PW,SIGMA(45),EX(400)                              ZPREX011
C     CALCULATION OF Z VALUES FOR PARTIAL EXPECTATIONS OF NORMAL DISTRIBZPREX012
      DO 100 IZ=1,400                                                   ZPREX013
      Z=IZ/100.                                                         ZPREX014
      IF(PEXP-EX(IZ))100,200,200                                        ZPREX015
  100 CONTINUE                                                          ZPREX016
  200 RETURN                                                            ZPREX017
      END                                                               ZPREX018
```

```
                                        PARTIAL EXPECTACTION PAGE NO .  1
 ***     CALCULATION OF Z VALUES FCR PARTIAL EXPECTATIONS OF NCRMAL CISTRIB***UTIONS
                         -----------------------------
 ------------------------* DO  1CC    IZ=1,400       *
 +                       -----------------------------
 +                                     I
 +                       -----------------------------
 +                       I   Z=IZ/1CC.               I
 +                       -----------------------------
 +                                     I
 +                                     IF
 +                              (PEXP-EX(IZ))
 +                        I-                            I
 +                        I -          I 0              I +
 +                       *1CC *      *2CO *          *200 *
 +
 ++++++++++++++++++++ 1CO  ++++++++++++++CONTINUE
                                       I
                      2CO         I RETURN
                                  I
                                  I
***END*******END*******END*******END*******END*******END*******END*******END*******END*******END*******END*******ENC***
```

APPENDIX VIII

THE COMBINED STATISTICAL-REQUIREMENTS PLANNING MODEL

```
      SUBROUTINE DECIDE
      INTEGER WSTAR
      COMMON IXM1(37,30),S(37),U(37),IXI(45),V(45),STH(3),OTH(3),SUH(3),
     1XIH(3),IBMR(3),IBN(3),IXLW(3),ISFG(15),IUS(15),IBO(15),ISL(15),
     2ITD(15),ITB(15),ITSL(15),AS(15),BS(15),CS(15),IFOR(15,6),CCSTY(5),
     3COST(5),COSTT,SHAT(12),ALPHA(15),BETA(15),XMU(15),SIG(15),FGP(15),
     4XMUX(3),SIGX(3),ICO(8),RC(8),E(15),A(15),B(15),IXM(45),DEC(48),
     5AP(45),ITI(45),TV(45),TSTH(3),TOTH(3),TSUH(3),TRH(3),RH(3),IC(45)
      COMMON STAR,WSTAR,PSTAR,WR,XMOT,IHC,ITC,IFC,XIC,IBC,ISC,N,MMAX,M
      COMMON COSTM,TIH(3),JFOR(45,18),JSE(45,18),ZZ,ZY,ZX,ZW
      COMMON IXYZ(15,30,4)
      COMMON IDEC(15,6),ICQ(15,6)
      COMMON IT2(15)
C
C
C     DCID3 - COMBINED STATISTICAL AND REQUIREMENTS PLANNING
C
      DO 3000 I=1,45
      DEC(I)=0.
      IF(I-15)3010,3010,3020
3010  DO 3030 J=1,6
      IF(M-1)3012,3012,3013
3013  IF(J-5)3011,3011,3012
3011  ICQ(I,J)=ICQ(I,J+1)
      GO TO 3030
3012  LL=J+11
      T1=0.
      DO 3040 K=J,LL
3040  T1=T1+JFOR(I,K)
      IF(IXI(I))3031,3031,3032
3032  ICQ(I,J)=SQRT((2.*T1*S(I)*WR)/((V(I)/FLOAT(IXI(I)))*12.*XIC))+0.5
3031  IF(ICQ(I,J)-JFOR(I,J))3033,3030,3030
3033  ICQ(I,J)=JFOR(I,J)
3030  CONTINUE
      T2=IXI(I)-IBO(I)
      DO 3050 J=1,5
      IF(T2-JFOR(I,J))3060,3060,3070
3060  IDEC(I,J)=ICQ(I,J)
      IF(J-5)3064,3050,3050
3064  T3=ICQ(I,J)
      IF(M-3)3065,3065,3066
3065  IDEC(I,J)=ICQ(I,J)+IBO(I)
      GO TO 3050
3066  T2=T2+T3
      GO TO 3050
3070  IDEC(I,J)=0
3050  T2=T2-JFOR(I,J)
      DEC(I)=IDEC(I,1)
      WRITE(6,9000)I,(ICQ(I,J),J=1,6),(IDEC(I,J),J=1,6)
9000  FORMAT(1H ,I5,6I10,5X,6I10)
      GO TO 3000
3020  IF(I-22)3080,3080,3090
3080  DO 3100 J=1,15
      IF(IXYZ(J,I-15,1))3100,3100,3120
3120  DEC(I)=DEC(I)+IDEC(J,2)*IXYZ(J,I-15,1)
      IF(DEC(J))3100,3100,3101
3101  DEC(I)=DEC(I)+DEC(J)*IXM1(J,I-15)
3100  CONTINUE
      DEC(I)=DEC(I)-IXI(I)
      GO TO 3000
```

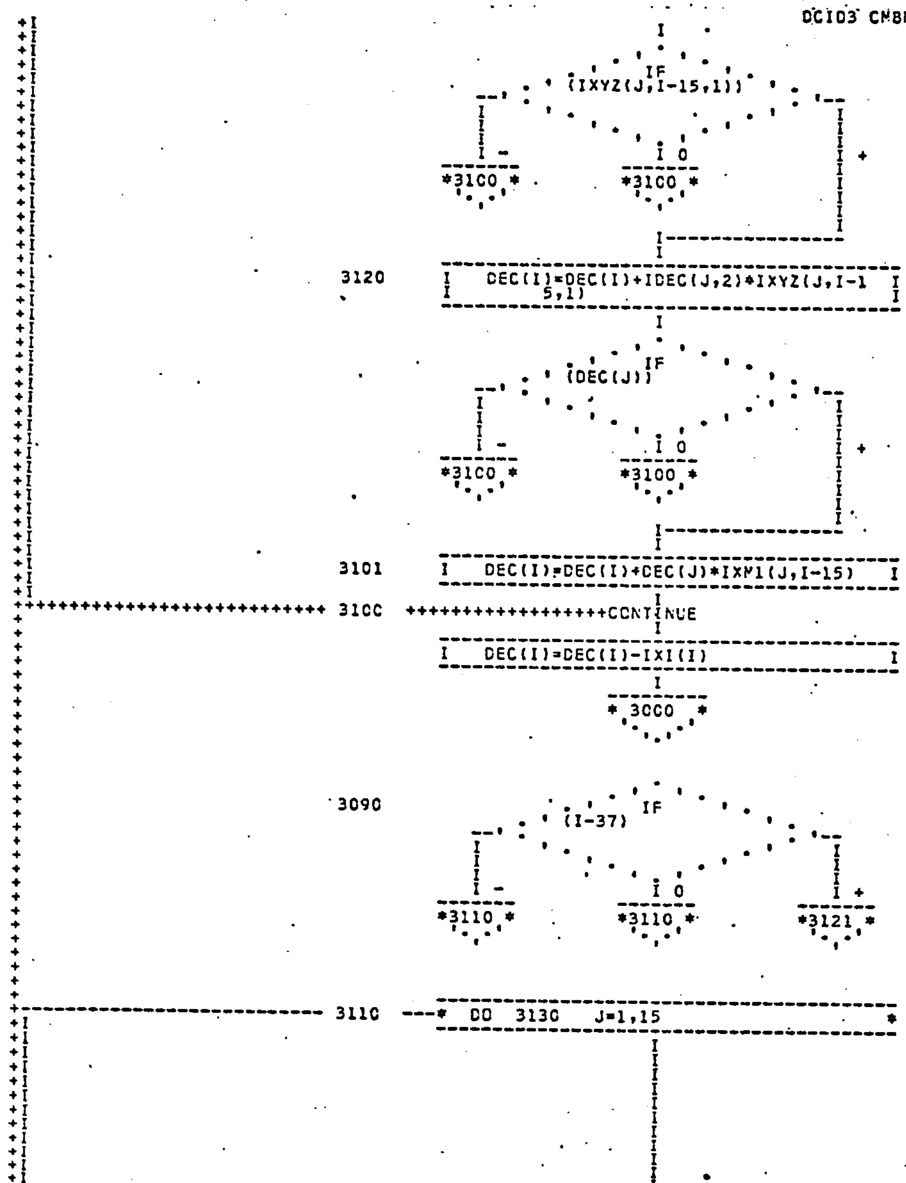
```
3090  IF(I-37)3110,3110,3121                                   DCID3060
3110  DO 3130 J=1,15                                           DCID3061
      IF(IXYZ(J,I-15,1))3140,3140,3150                         DCID3062
3150  DEC(I)=DEC(I)+IDEC(J,2)*IXYZ(J,I-15,1)                   DCID3063
3140  IF(IXYZ(J,I-15,2))3130,3130,3160                         DCID3064
3160  DEC(I)=DEC(I)+IDEC(J,3)*IXYZ(J,I-15,2)                   DCID3065
3130  CONTINUE                                                 DCID3066
      DO 3170 K=1,22                                           DCID3067
      IF(DEC(K))3170,3170,3180                                 DCID3068
3160  DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)                        DCID3069
3170  CONTINUE                                                 DCID3070
      DEC(I)=DEC(I)-IXI(I)                                     DCID3071
      GO TO 3000                                               DCID3072
3121  IF(I-41)3190,3190,3200                                   DCID3073
3190  DO 3210 J=1,15                                           DCID3074
      IF(IXYZ(J,I-15,2))3220,3220,3230                         DCID3075
3230  DEC(I)=DEC(I)+IDEC(J,3)*IXYZ(J,I-15,2)                   DCID3076
3220  IF(IXYZ(J,I-15,3))3210,3210,3240                         DCID3077
3240  DEC(I)=DEC(I)+IDEC(J,4)*IXYZ(J,I-15,3)                   DCID3078
3210  CONTINUE                                                 DCID3079
      DO 3250 K=16,37                                          DCID3080
      IF(DEC(K))3250,3250,3260                                 DCID3081
3260  DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)                        DCID3082
3250  CONTINUE                                                 DCID3083
      DEC(I)=DEC(I)-IXI(I)                                     DCID3084
      GO TO 3000                                               DCID3085
3200  DO 3270 J=1,15                                           DCID3086
      IF(IXYZ(J,I-15,3))3280,3280,3290                         DCID3087
3290  DEC(I)=DEC(I)+(IDEC(J,3)+IDEC(J,4))*IXYZ(J,I-15,3)       DCID3088
3280  IF(IXYZ(J,I-15,4))3270,3270,3300                         DCID3089
3300  DEC(I)=DEC(I)+(IDEC(J,4)+IDEC(J,5))*IXYZ(J,I-15,4)       DCID3090
3270  CONTINUE                                                 DCID3091
      DO 3310 K=16,37                                          DCID3092
      IF(DEC(K))3310,3310,3320                                 DCID3093
3320  DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)                        DCID3094
3310  CONTINUE                                                 DCID3095
      DEC(I)=DEC(I)-IXI(I)-ICO(I-37)                           DCID3096
3000  CONTINUE                                                 DCID3097
      DO 20 IC=1,3                                             DCID3098
      TOT=0.                                                   DCID3099
      IL=IBMR(IC)+1                                            DCID3100
      IU=IBMR(IC)+IBN(IC)                                      DCID3101
      DO 61 IP=IL,IU                                           DCID3102
      IF(DEC(IP))61,61,62                                      DCID3103
62    TOT=TOT+S(IP)+DEC(IP)*U(IP)                              DCID3104
61    CONTINUE                                                 DCID3105
20    DEC(IC+45)=IFIX(TOT/160.+0.5)                            DCID3106
      RETURN                                                   DCID3107
      END                                                      DCID3108
```

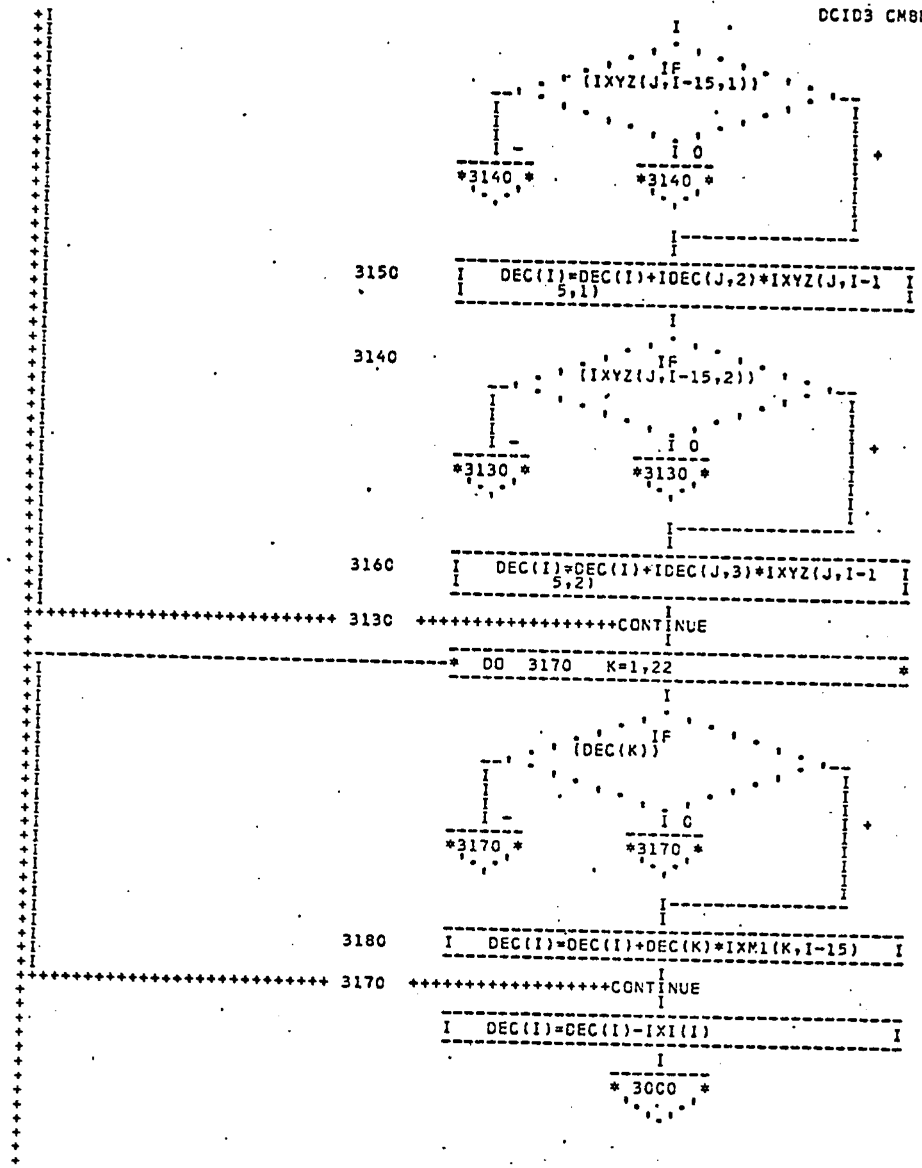
*** DCID3 - COMBINED STATISTICAL AND REQUIREMENTS PLANNING ***

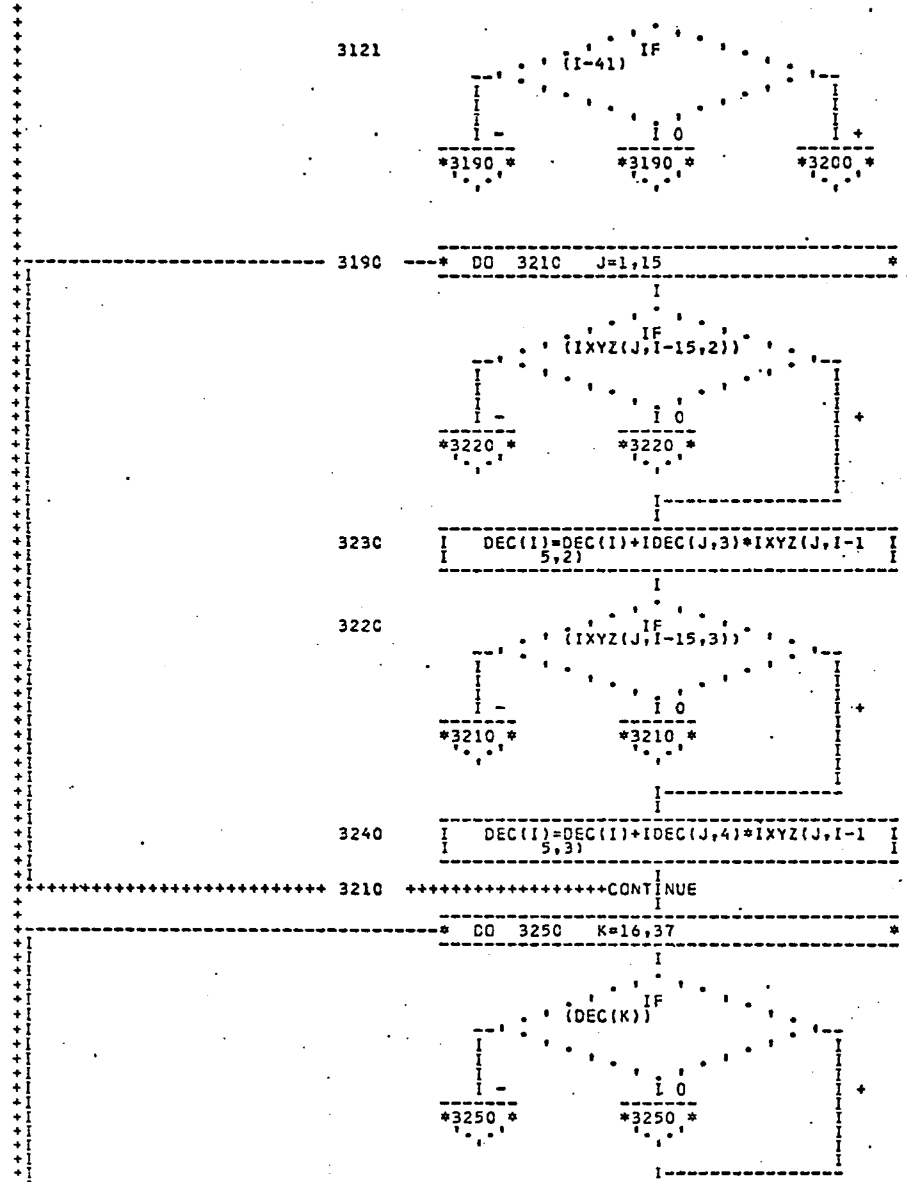
```
      DO 3000   I=1,45
      DEC(I)=0.
      IF (I-15)   -  *3010*   0  *3010*   +  *3020*

3010  DO 3030   J=1,5
      IF (M-1)    -  *3012*   0  *3012*   +

3013  IF (J-4)    -  *3011*   0  *3011*   +  *3012*

3011  IQQ(I,J)=IQQ(I,J+1)
      IDEC(I,J)=IDEC(I,J+1)
      * 3030 *

3012  LL=J+11
      T1=0.
      DO 3040   K=1,11
```

```
3040  T1=T1+JFOR(I,K)

      IF (IXI(I))
        -  *3031*
        0  *3031*
        +

3032  IQQ(I,J)=SQRT((2.*T1*S(I)*WR)/((V(
      I)/FLOAT(IXI(I)))*12.*XIC))+0
      .5

3031  IF (IQQ(I,J)-JFOR(I,
      J))
        -  *3033*
        0  *3030*
        +  *3030*

3033  IQQ(I,J)=JFOR(I,J)

3030  CONTINUE

      IF (M-1)
        -  *3050*
        0  *3050*
        +  *3060*

3050  DO  3051   J=1,4

3051  IDEC(I,J)=IQQ(I,J)

3060  T2=0.
      IT3=0

      DO  3052   J=1,5
```

```
                                     T2=T2+JSE(I,J)**2

         3052                        IT3=IT3+JFOR(I,J)
                                     ISS=ZZ*SQRT(T2)+0.5
                                     IT4=0

                                   * DO 3053 J=1,4                *

         3053                        IT4=IT4+IDEC(I,J)

                                     IF
                                     (IXI(I)-IBO(I)+IT
                                     4-IT3-ISS)
                              -            0            +
                           *3054 *      *3054 *      *3055 *

         3054                        IDEC(I,5)=ICO(I,5)+ISS-IXI(I)+IBO(
                                        I)-IT4+IT3
                                        * 3056 *

         3055                        IDEC(I,5)=0

         3056                        DEC(I)=IDEC(I,1)
                                        * 3000 *

         3020                        IF
                                     (I-22)
                              -            0            +
                           *3080 *      *3080 *      *3090 *

         3080                      * DO 3100 J=1,15               *
```


IF (IXYZ(J,I-15,1))
*3100 *
*3100 *
+
3120
DEC(I)=DEC(I)+IDEC(J,2)*IXYZ(J,I-15,1)
IF (DEC(J))
*3100 *
*3100 *
+
3101
DEC(I)=DEC(I)+DEC(J)*IXM1(J,I-15)
3100
CONTINUE
DEC(I)=DEC(I)-IXI(I)
* 3000 *
3090
IF (I-37)
*3110 *
*3110 *
*3121 *
+
3110
DO 3130 J=1,15


IF (IXYZ(J,I-15,1))
- 0 +
3140
3140
3150 DEC(I)=DEC(I)+IDEC(J,2)*IXYZ(J,I-15,1)
3140 IF (IXYZ(J,I-15,2))
3130
3130
3160 DEC(I)=DEC(I)+IDEC(J,3)*IXYZ(J,I-15,2)
3130 CONTINUE
DO 3170 K=1,22
IF (DEC(K))
3170
3170
3180 DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)
3170 CONTINUE
DEC(I)=DEC(I)-IXI(I)
3000


3121
IF (I-41)
- 0 +
*3190 * *3190 * *3200 *
3190 DO 3210 J=1,15
IF (IXYZ(J,I-15,2))
- 0 +
*3220 * *3220 *
3230 DEC(I)=DEC(I)+IDEC(J,3)*IXYZ(J,I-15,2)
3220 IF (IXYZ(J,I-15,3))
- 0 +
*3210 * *3210 *
3240 DEC(I)=DEC(I)+IDEC(J,4)*IXYZ(J,I-15,3)
3210 CONTINUE
DO 3250 K=16,37
IF (DEC(K))
- 0 +
*3250 * *3250 *

```
3260   I DEC(I)=DEC(I)+DEC(K)*IXM1(K,I-15)  I
3250   ++++++++++++++++CONTINUE
       I DEC(I)=DEC(I)-IXI(I)  I
       * 3000 *
3200   ---* DO 3270  J=1,15  *
            IF
       (IXYZ(J,I-15,3))
       -  *3280*    0  *3280*    +
3290   I DEC(I)=DEC(I)+(IDEC(J,3)+IDEC(J,4)  I
       I        )*IXYZ(J,I-15,3)             I
3280        IF
       (IXYZ(J,I-15,4))
       -  *3270*    0  *3270*    +
3300   I DEC(I)=DEC(I)+(IDEC(J,4)+IDEC(J,5)  I
       I        )*IXYZ(J,I-15,4)             I
3270   ++++++++++++++++CONTINUE
       * DO 3310  K=16,37  *
            IF
       (DEC(K))
       -  *3310*    0  *3310*    +
```

```
                                                DCID3 CMBND STAT-RP   PAGE NO    8
          3320   I  DEC(I)=DEC(I)+DEC(K)*IXP1(K,I-15)   I
          3310   +++++++++++++++++CONTINUE
                 I  DEC(I)=DEC(I)-IXI(I)-ICC(I-37)      I
          3000   +++++++++++++++++CONTINUE
                 *  DO  20       ID=1,3                 *
                 I  TOT=0.
                 I  IL=IBMR(ID)+1
                 I  IU=IBMR(ID)+IBN(ID)                 I
                 *  DO  61       IP=IL,IU               *
                          IF
                      (DEC(IP))
                   -           0           +
                 *61 *       *61 *
            62   I  TOT=TOT+S(IP)+DEC(IP)*U(IP)         I
            61   +++++++++++++++++CCNTINUE
            20   +++I  DEC(ID+45)=IFIX(TOT/160.+0.5)    I
                       I RETURN
***END*******END*******END*******END*******END*******END*******END*******END*******END*******END*******END***
```