INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI®

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

A MODEL FOR PRODUCTION SCHEDULING AND SEQUENCING USING CONSTRAINTS MANAGEMENT AND GENETIC ALGORITHM

A Dissertation

Submitted

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Industrial Technology

Approved:

Dr. Mohammed F. Fahmy, Faculty Advisor

Dr. MD Salim, Co-Advisor

Douglas Pine

Dr. Douglas Pine, Committee Member

Dr. Michael Spencer, Committee Member

Barry Wilson

Dr. Barry Wilson, Committee Member

Ahmad Nadeem Choudhry

University of Northern Iowa

December 2000

UMI Number: 9992045

Bell & Howell Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

A MODEL FOR PRODUCTION SCHEDULING AND SEQUENCING USING CONSTRAINTS MANAGEMENT AND GENETIC ALGORITHM

An Abstract of a Dissertation
Submitted
In Partial Fulfillment
of the Requirements for the Degree
Doctor of Industrial Technology

Approved:

Dr. Mohammed F. Fahmy, Faculty Advisor

John W. Somervill

Dr. John W. Somervill, Dean of the Graduate College

Ahmad Nadeem Choudhry
University of Northern Iowa
December 2000

ABSTRACT

Production planning and control (PP&C) are among the most critical activities in manufacturing. Proper use of PP&C methods can give organizations a competitive advantage in the global economy. The expected results of this research will allow manufacturing organizations to maximize the effectiveness of PP&C methods, thereby improving their competitive position in the global economy.

This research was an extension of a previous unpublished study, which investigated the PP&C methods being used at a midwestern manufacturer of agricultural equipment (MMAE). The current research study identified the constraints inherent in the production planning and control system and then developed and validated a master production scheduling and sequencing optimization model based on constraints management and utilizing genetic algorithms.

The specific objectives of this research were as follows: (a) identify the system's constraint, (b) develop a scheduling and sequencing model to address the identified constraints, (c) develop and validate the proposed model by simulation, and (d) identify and document improvements attributed to the operational change resulting from the implementation of the optimization model.

The research examined the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on five variables for the final assembly line and four downstream processes at an engine manufacturing plant of a MMAE. The variables were cycle time, queue size, utilization of work centers, flow rate of engines, and total output of engines.

A two-part model, based on constraints management philosophy of production planning and control methods, was developed by the researcher in Excel, one part for scheduling and the other for sequencing. Using data from 100 production days during the fall of 1999 and the spring of 2000, simulations for the current scheduling and sequencing method (the control condition) and for the proposed method (the experimental condition) were compared. Output from the simulations for the experimental and control conditions was statistically analyzed.

The results of this research indicated (a) cycle time for the experimental condition was reduced, but the reduction was not statistically significant; (b) queue size for the experimental condition was also reduced, as expected, but once again, the reduction was not statistically significant; (c) total utilization of work centers was increased, as expected, and the increase was statistically significant; (d) the experimental condition's simulation results indicated very minimal improvements for the even flow of engines; and (e) the average total number of engines processed for the experimental condition was increased, as expected, and the increase was statistically significant.

DEDICATION

This doctoral dissertation is dedicated

To my mother, who dedicated

her life for the sake of mine.

To my mother, I am eternally grateful

ACKNOWLEDGMENTS

The writer praises Almighty Allah for His blessing, support, and guidance that was necessary for the completion of this research project. The author would also like to acknowledge all the people who lent their support, guidance, love, and assistance toward the successful completion of this study. Most notably, the author would like to thank his doctoral committee members for their unconditional support and guidance.

The author would like to thank Dr. Mohammed F. Fahmy, department head and committee advisor, for his unconditional support and guidance throughout the DIT program. The author would also like to thank Dr. MD Salim, co-advisor, for his valuable suggestions. The author would also like to express his appreciation to Dr. Douglas Pine, committee member, for his imperative suggestions from the very early stages of this project. Likewise, special thanks are extended to Dr. Michael S. Spencer, committee member, for providing guidance in the early stages of this project and for valuable comments and suggestions; Dr. Barry Wilson, committee member, for providing his statistical expertise; and Cheryl Smith for her valuable editing and suggestions.

This research project could not have been completed without support from Deere & Company. I would like to extend my sincere thanks to all the Deere & Company employees for their technical help, valuable suggestions, and assistance in relating academic aspects of production planning and control systems in a practical way. The author would like to especially thank Mr. Gordon Rehn for providing simulation expertise for this study.

Finally, the author would like to express very special gratitude to my wife, Arfana, and my children, Aisha, Omer, Bilal, and Fatima, for their understanding and

support. To my wife, I owe a great debt for her patience throughout this project. Without her unconditional support, this research would not have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURE PAGE

PAGE

CHAPTER I

INTRODUCTION

Background

Manufacturing after World War II

World War II brought about many changes to the manufacturing industry worldwide. Manufacturing in America flourished during the war because its industrial infrastructure base had remained intact whereas the industrial infrastructures in Europe, Russia, and Asia were destroyed. Even Asian countries not directly involved in the war were not able to compete in the international market due to the lack of technological advances in their manufacturing industries. As a result, the only nation left to lead the world in manufacturing was the United States. American manufacturers understood this opportunity and become the undisputed mass production leaders of the world.

From the 1940s to the 1960s, American manufacturers enjoyed a period of prosperity. During this time, mass production was emphasized, but quality was not much of a concern for many manufacturers. In the middle 1960s, a few foreign countries started to compete with American products in the international and U.S. markets. This trend continued so that by the 1970s and 1980s, the United States was beginning to "look like an economic colony of Japan" (Wight, 1984, p. 9). American manufacturers were forced to look critically at their cost structures. During the oil embargo and inflation cycle of the 1970s, American manufacturing firms recognized the need to reduce waste and control costs.

One way for the manufacturing industry to stay competitive was to reduce total costs, focusing particularly on inventory and inventory-related costs. That is the goal of the production planning and control (PP&C) system, which is one of the most critical activities in the manufacturing environment (Vollmann, Berry, & Whybark, 1988). Proper use of PP&C methods can give organizations a competitive advantage in the global economy (Bai & Tsai, 1994). Hopp and Spearman (1996) suggest a hierarchical planning framework of production planning and control. Their framework is divided into three basic levels, as depicted in Figure 1: (a) strategy (long-term planning), (b) tactics (intermediate-term planning), and (c) control (short-term planning).

Evolution of the Production Planning and Control Systems

Before the development of computer technology, production planning and control functions were mainly accomplished manually. Some of the common techniques used were the two-bin system, economic order quantity (EOQ), and reorder point (Gilbert & Schonberger, 1983).

During the 1960s, when computers began to be used in the manufacturing industry, the material requirement planning (MRP) technique was developed by Joseph Orlicky (Taylor, 1994). MRP is a tool used for material and priority planning, the basic function of an MRP system is to plan for material requirements based on planned production levels. The remarkable growth in computing power, along with the reduction in the size and price of computers, allowed for the accelerated implementation of MRP in the United States. This system was considered to be far superior to the older reorder point systems (Orlicky, 1975; Wight, 1974), and it became a phenomenal success. Organizations that implemented the MRP technique increased their inventory turnover per year by more than 100% compared with more traditional production planning and control methods (Hall, 1983). MRP has been used in America since the 1970s,

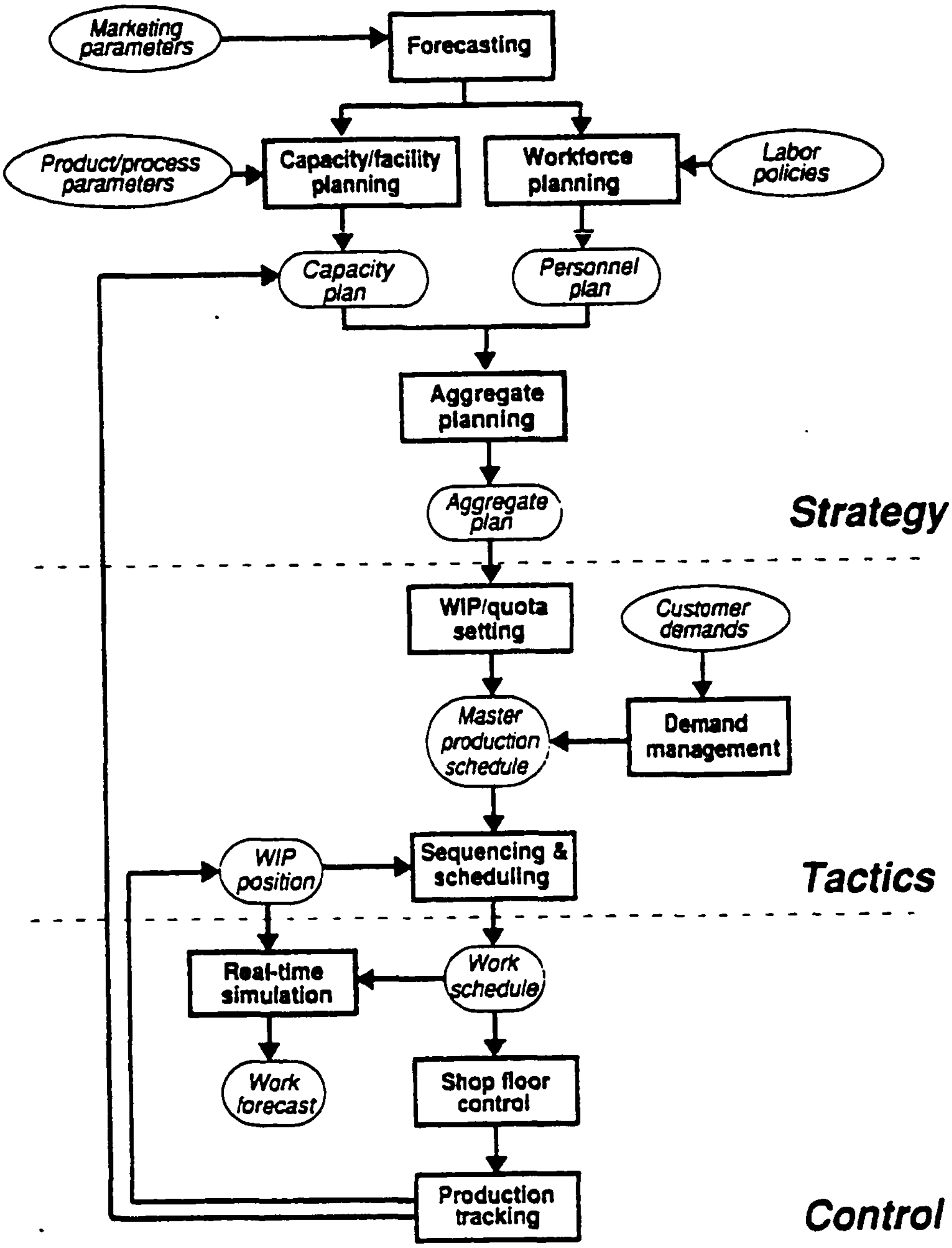


Figure 1. Production planning and control hierarchy for pull system. From Factory Physics: Fundamentals of Manufacturing Management (p. 388), by W. J. Hopp and M. L. Spearman, 1996, Chicago: Irwin. Copyright 1996 by Richard D. Irwin. Adapted by permission.

and now the number of companies who employ MRP is in the hundreds of thousands. More than 100 software companies are engaged in the development of MRP software (Das, 1995).

Even though manufacturers derived many benefits from MRP, some limitations were inherent in the technique. MRP ignored very dynamic elements of the shop-floor environment such as capacity limitations and lead time (Berry, Schmitt, & Vollmann, 1982; Schmitt, Berry, & Vollmann, 1988). Lambrecht and Decaluwe (1988) suggest at the operational level of MRP, many batch sizing and timing decisions are "push" in nature because they are created using fixed planning parameters. Many new modules were added to the original MRP system to minimize these limitations. In the early 1970s a new version of MRP, called manufacturing resource planning (MRP II), was introduced as a more comprehensive, system-wide production planning and control technique. Many new modules were also added in MRP II, but it was still a push system. The problems inherent in MRP stem from the failure to reconcile the differences between pull and push elements in production control systems (Veral, 1995). This underlying condition within the MRP environment has caused many difficulties for a large number of organizations striving to meet ever-changing customer demands.

While Western manufacturers were engaged in developing MRP and MRP II, Japanese organizations were formulating their own production planning and control methods. The just-in-time (JIT) concept emerged from the study of the Japanese automobile industry during the 1970s (Spencer, 1992). JIT is based on the philosophy of eliminating any activities that do not add value. Its goal is to get the material to its next processing station just at the time it is needed (Amerine, Ritchey, Moodie, & Kmec, 1993), in the interests of minimizing the inventories for raw material, work-in-process, and finished goods.

Another production planning and control approach, developed by an Israeli physicist Eli Goldratt in the late 1970s, is the theory of constraints. The concept of theory of constraints has subsequently evolved to become known as constraints management (Spencer & Cox, 1995), and this more contemporary term is used hereafter. Constraints management CM is a set of management principles that help to identify obstacles in achieving the goal of an organization and to establish the changes necessary to remove those obstacles. CM recognizes that the strength of any chain is dependent upon its weakest link, which is what restrains the system's throughput. CM assumes that the goal of manufacturing organizations is to make (more) money now and in the future, and describes three avenues to achieve this goal: (a) increase throughput, (b) reduce inventory, and (c) reduce operating expense.

There seems to be no one right production planning and control system for all manufacturing problems. For some organizations, MRP and MRP II work well; for others JIT or CM are better choices. Deciding which production planning and control system to implement can become time consuming yet difficult to implement for only a "trial period."

These three techniques, MRP, JIT, and CM, are the most commonly used in manufacturing today. However, they are not interchangeable; one system may be appropriate for a particular manufacturing situation but not for another.

Statement of the Problem

Because no single production planning and control (PP&C) technique is suitable for all situations, deciding which system to implement can become time consuming. Yet implementing one for a trial period can be costly and difficult. A technology is needed that can employ various types of PP&C methodologies and generate the optimal production plan.

This research is an extension of a previous unpublished study (Choudhry, 1998), which investigated the PP&C methods being used at a midwestern manufacturing organization involved in the production of agriculture equipment. The current research study identified the constraints inherent in the production planning and control system, and based on these constraints, developed and validated a master production scheduling and sequencing optimization model based on constraints management and utilizing genetic algorithms.

Statement of the Purpose

As noted earlier, production planning and control are among the most critical activities in manufacturing. The expected results of this research will allow manufacturing organizations to maximize the effectiveness of PP&C methods, thereby improving their competitive position in the global economy. To that end, the goal of this research is to develop an optimization model based on constraints management and genetic algorithm to address the constraints in the PP&C methods being used at the factory under study.

This research, based on an analysis of five areas of PP&C (master production scheduling, priority planning, capacity planning, priority control, and capacity control), identifies the constraints in that system, and develops and validates master production scheduling and sequencing optimization model based on constraints management and genetic algorithm. The specific objectives of this research were as follows:

1. Identify the system's constraint.

2. Develop a scheduling and sequencing model to address the identified constraints.

3. Develop and validate the proposed model by simulation using GPSS/H and PROOF, products of the Wolverine Software Corporation located in Annandale, Virginia. GPSS/H is a simulation language, and PROOF is a animation software used within Excel file format.

4. Identify and document improvements attributed to the operational change resulting from the implementation of the optimization model.

Importance of the Research

Which production planning and control technique or methodology is best for a company? This question has puzzled many managers in the past. The three main production planning and control systems are material requirement planning, just-in-time, and constraints management. According to Aggarwal (1985), MRP, JIT, and CM are the three most popular management philosophies in current use. There is no consensus between academicians and practitioners as to which approach is best. According to Spencer (1992), "the three techniques are, to a degree, somewhat mutually exclusive.

There appears to be a need to study the three systems in a framework in which their characteristics and behaviors can be examined in detail" (p. 5). These three techniques are discussed in more detail later in this chapter.

Aggarwal reports in his 1985 article:

> During the last 15 years, three important approaches—material requirement planning (MRP), kanban (JIT), and optimized production technology (OPT)—have invaded operations planning and control in quick succession, one after the other. Each new system has challenged old assumptions and ways of doing things....factory managers must decide which approach to adopt to meet current and future needs. Installing any one requires several years to train company personnel and millions of dollars of investment. (p. 99)

Most organizations don't have the resources to try out a method before making a final choice; therefore the managers are left with the grave decision of which one to use.

According to Goldratt and Fox (1986):

> The Western manager is challenged to solve a very fundamental problem from this alphabet soup of solutions. To understand each of these new technologies can, by itself, be a time-consuming challenge. Deciding which is best is a formidable task. Figuring out how to put them all together seems beyond our reach. Since we don't have the time, resources or funds to do everything, everywhere, we had better be convinced that we are taking the actions that will leapfrog us back into the race. There is no longer margin for error and no time for risky experiments. (p. 16)

There needs to be a better way of selecting and implementing a production planning system.

This research can assist practitioners who are trying to learn more about the three techniques. The advantages and disadvantages of each management philosophy, as well as problems that might arise during or after implementation, are discussed by examining one company's experiences in an in-depth case study. The developed scheduling model for optimization, presented after this discussion, could be used in various manufacturing environments.

Research Questions

The previous unpublished study (Choudhry, 1998) focused on the PP&C methods then in use at an engine manufacturing plant (EMP) of a midwestern manufacturer of agricultural equipment. Methods for master production schedule, production priority, and production capacity were explored and documented. Problems in planning and controlling master production schedule, production priority, and production capacity were also identified and documented. The findings of this study are summarized in chapter II.

The current research addresses the following questions. The findings are reported in chapter IV.

1. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the cycle time for the final assembly line and four downstream processes at an engine manufacturing plant (EMP) of a midwestern manufacturer of agricultural equipment (MMAE)?

2. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the queue size for the final assembly line and four downstream processes at EMP?

3. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the utilization of work centers in the final assembly line and four downstream processes at EMP?

4. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the flow rate of engines through the final assembly line and four downstream processes at EMP?

5. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the total output of engines through the final assembly line and four downstream processes at EMP?

Guide (1992) collected and analyzed time in system (cycle time) and work-in-process levels (queue size, inventory levels) to determine if synchronous manufacturing principles produced improved performance in comparison with current production planning and control methodology at a Naval Aviation depot. Taylor (1994) also uses some of these performance measurements to compare the three work-in-process inventory control systems: MRP, JIT, and CM. Performance measurements analyzed by Taylor were: inventory (queue size), throughput (total output of engines), lead time (cycle time), and utilization (utilization of work centers). Manoharan (1997) analyzed total system output (total output of engines), flow time (flow rate of engines), and WIP inventory (queue size) to evaluate the performance of two manufacturing systems, JIT and CM.

Assumptions

The following assumptions were made in pursuit of this research study:

1. That Microsoft Excel is the common production planning tool utilized by various facilities within the total organization.

2. That the production planning and control methods stay the same during the course of this research study at the manufacturing facility under study.

Limitations

This research study was conducted in view of the following limitations:

1. This model was developed in Microsoft Excel and will only work in an Excel environment.

2. For optimization, this research utilizes genetic algorithm-based Evolver software developed by Palisade Inc. This model is limited in application within an Evolver environment.

Definition of Terms

The following terms are defined to clarify their use in the context of this research study.

- Capacity planning: The process of determining the amount of capacity to produce in the future. This process may be performed at an aggregate or product-line level (resource planning), or at the master-scheduling level (rough-cut planning), at the detailed or work-center level (capacity requirements planning). (Cox, Blackstone, & Spencer, 1995, p. 11)

- Capacity control: "The process of measuring production output and comparing it to the capacity plan, determining if the variance exceeds preestablished limits, and taking corrective actions to get back on plan if the limits are exceeded" (Cox et al., 1995, p. 11).

- Flow rate: As defined in the APICS Dictionary, "running rate; the inverse of cycle time" (Cox et al., 1995, p. 33). Flow rate is also defined by number of units per shift or per hour.
- Genetic algorithm (GA): Holland (1992) defines genetic algorithm as "a probabilistically guided search method, developed originally in the 1970s as a computer science tool to improve programming structures and performance" (pp. 66-72). Chambers (1991) defines it as a "problem solving method that uses genetics as its model of problem solving" (p. 9).
- Just-in-time (JIT): A philosophy of manufacturing based on planned elimination of all waste and continuous improvement of productivity. It encompasses the successful execution of all manufacturing activities required to produce a final product, from design engineering to delivery and including all stages of conversion from raw material onward. The primary elements of zero inventories are to have only the required inventory needed; to improve quality to zero defects; to reduce lead times by reducing setup times, queue lengths, and lot sizes; to incrementally revise the operations themselves; and to accomplish these things at minimum cost. (Cox et al., 1995, p. 42)
- Material Requirements Planning (MRP): A set of techniques that use bill of material data, inventory data, and the master production schedule to calculate requirements for materials. It makes recommendations to release replenishment orders for material. Further, because it is time-phased, it makes recommendations to reschedule open orders when due dates are not in phase. Time-phased MRP begins with the items listed on the MPS and determines (a) the quantity of all components and materials required to fabricate those items and (b) the date that the components and materials are required. Time-phased MRP is accomplished by exploding the bill of material, adjusting for inventory quantities on hand or on order, and offsetting the net requirements by the appropriate lead times. (Cox et al., 1995, pp. 49-50)
- Master production schedule (MPS): The anticipated build schedule for those items assigned to the master scheduler. The master scheduler maintains this schedule, and in turn, it becomes a set of planning numbers that drives

material requirements planning. It represents what the company plans to produce in specific configurations, quantities, and dates. The master production is not a sales forecast that represents a statement of demand. The master production schedule must take into account the forecast, the production plan, and other important considerations such as backlog, availability of material, availability of capacity, and management policies and goals. (Cox et al., 1995, p. 49)

- Priority control: "The process of communicating start and completion dates to manufacturing departments in order to execute a plan. The dispatch list is the tool used to provide these dates and priorities based on the current plan and status of all open orders" (Cox et al., 1995, p. 63).
- Priority planning: "The function of determining what material is needed and when. Master production scheduling and material requirements planning are elements used for the planning and re-planning process to maintain proper due dates on required materials" (Cox et al., 1995, p. 63).
- Theory of constraints, now known as constraints management (CM): A management philosophy developed by Dr. Eliyahu M. Goldratt that can be viewed as three separate but interrelated areas–logistics, performance measurement, and logical thinking. Logistics include drum-buffer-rope scheduling, buffer management, and VAT analysis. Performance measurement includes throughput, inventory and operating expense, and the five focusing steps. Thinking process tools are important in identifying the root problem (current reality tree), identifying and expanding win-win solutions (evaporating cloud and future reality tree), and developing implementation plans (prerequisite tree and transition tree). (Cox et al., 1995, p. 85)

CHAPTER II

REVIEW OF LITERATURE

To understand the nature of the ever-changing manufacturing production environment, we need to develop a common set of functions that are not only unique to production itself but can be generalized to all production organizations (Cox & Spencer, 1998). This research is organized around five functions common to production planning and control. These five functions are master production schedule (MPS), priority planning, capacity planning, priority control, and capacity control. According to Cox and Spencer (1998), the origin of the five production planning and control functions is unclear, but the first source of written reference appears in Oliver Wight's 1984 book, Manufacturing Resource Planning (MRP II): Unlocking American Productivity Potential.

The purpose of production planning and control (PP&C) is to plan and control the production process with regard to time and quantity. According to Corsten and May (1996, p. 69), for the PP&C function, the following four questions have to be answered:

- Which products and parts are to be produced and what is their quantity level?
- Which parts are to be delivered by the supplier in what quantity and when?
- Which capacity utilization results from the master production schedule and how can a capacity adjustment take place?
- In what sequence are the production orders to be worked off and at which workstation?

This chapter provides a review and analysis of the literature related to material requirements planning (MRP), just-in-time (JIT), constraints management (CM), and genetic algorithms (GA) and discusses how each relates to five functions common to production management.

Material Requirements Planning

Evolution

MRP is a tool used for material and priority planning. The basic function of an MRP system is to plan for material requirements based on planned production levels. Wight (1984, p. 47) suggests that MRP tries to answer the following fundamental manufacturing questions:

- What are we going to manufacture?
- What does it take to make it?
- What do we have in our inventory?
- What do we have to acquire?

These fundamental questions, used throughout the manufacturing industry, serve to generate a list of parts needed for the next month in order to avoid part shortages. From this informal system, a powerful one has evolved called material requirements planning. "MRP is simply the logic of the informal system – the shortage list – developed into a formal scheduling system" (Wight, 1984, p. 47).

Although MRP has been in practice informally for many decades in the manufacturing industry, the first published work that formally discussed MRP was Material Requirements Planning, written by Joseph Orlicky in 1975. In his book he states:

> In some rudimentary form, MRP has no doubt existed as long as manufacturing. It has been evolving gradually, moving onto successively higher plateaus with every enhancement in data processing capability. MRP had its origin on the firing line of a plant. It has been painstakingly developed into its present stage of relative perfection by practicing inventory managers and inventory planners. (p. 38)

Eventually MRP developed into an overall system called closed loop MRP. Figure 2, is a schema of a closed loop MRP system. The production plan establishes production volumes for product families. The master schedule takes the production plan in units for product families and breaks it down into component parts. Material requirements planning looks at the parts in inventory and determines what component parts are needed to accomplish the production plan. The capacity requirements plan determines the standard hour requirements for the production plan. Once planning for material and capacity requirements is completed, it must be determined if the plans are realistic. If they are realistic, then both material and capacity plans need to be monitored to ensure that the plans are being executed.

Despite the formalization of the MRP system, its limitations were still confining to the organization's ability to perform better production planning and control functions. Finance, a big piece of the puzzle, was still missing in the closed loop MRP; financial systems were not tied to the closed loop MRP. In the 1970s, manufacturing resource planning (MRP II) evolved out of the closed loop MRP, tying the financial system to the operating system. As Wight (1984, p. 49) noted, "tying the financial and the operating systems together was the big step from closed loop MRP to MRP II." Figure 3 is a schema of an MRP II system.

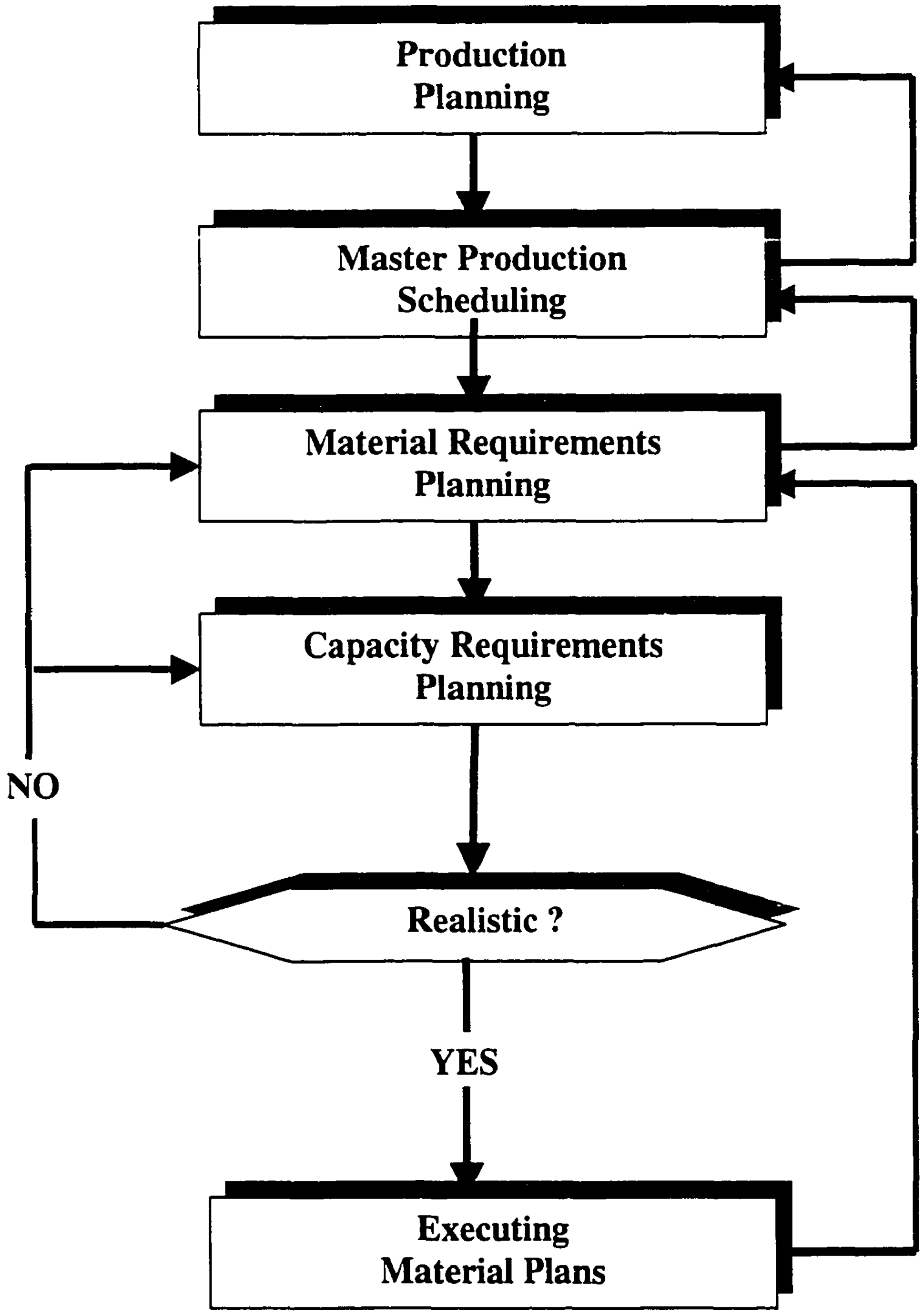


Figure 2. Closed loop MRP.

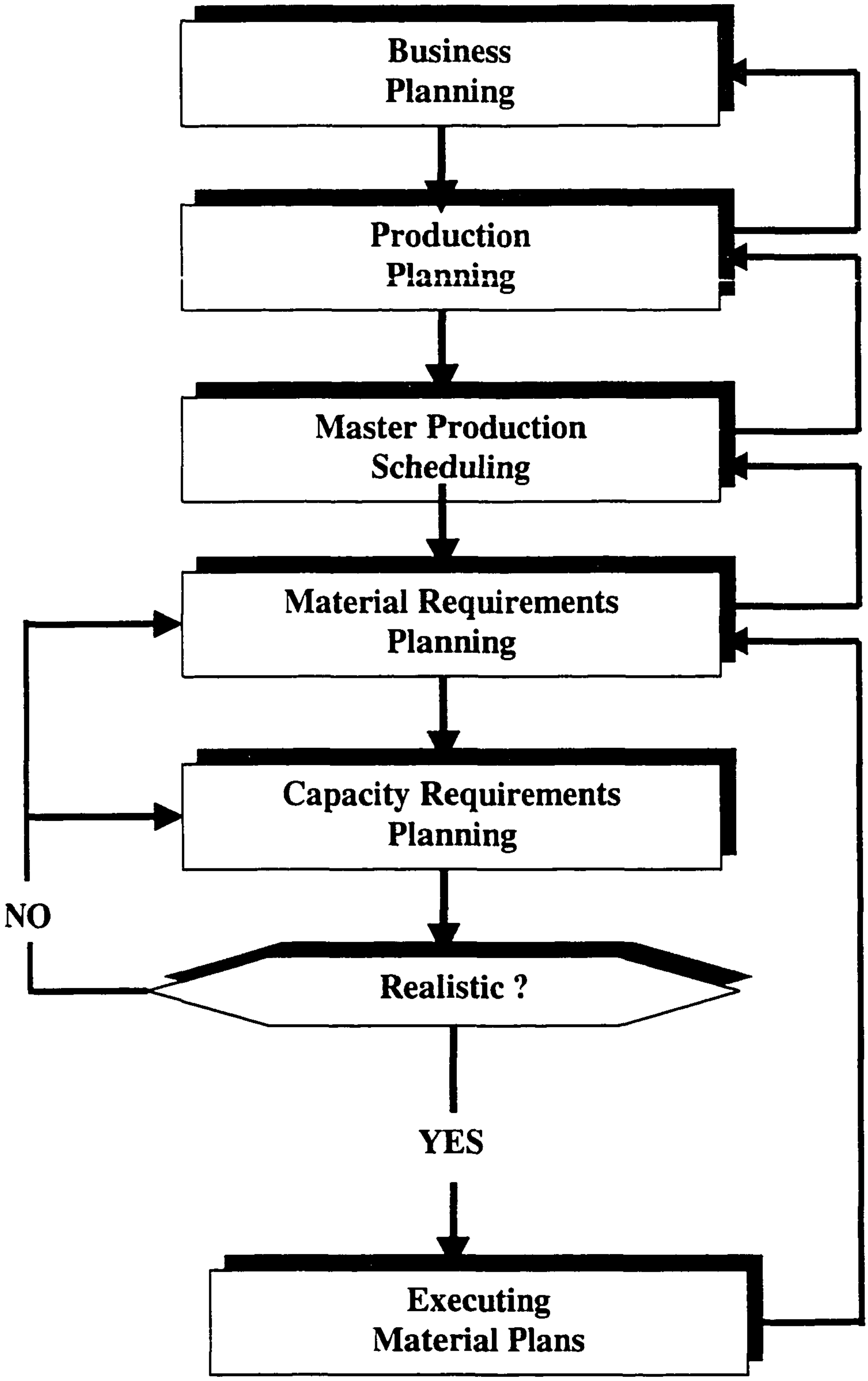


Figure 3. Manufacturing resource planning (MRP II).

Functionality

MRP deals with end-items (finished products) and the component parts (lower level items) that make up the end items. The bill of material (BOM) connects the end items with the lower level items. Figure 4 illustrates a typical bill of material for the end-item X. To facilitate the MRP processing, each component part in the bill of material is assigned a low level code (LLC). The LLC indicates the lowest level for which a part is used in a bill of material. In the following figure, the end item X has an LLC of 0. The component parts 10 and 20 have an LLC of 1, parts 30, 40, and 50 have an LLC of 2; and part 60 and 70 an LLC of 3.

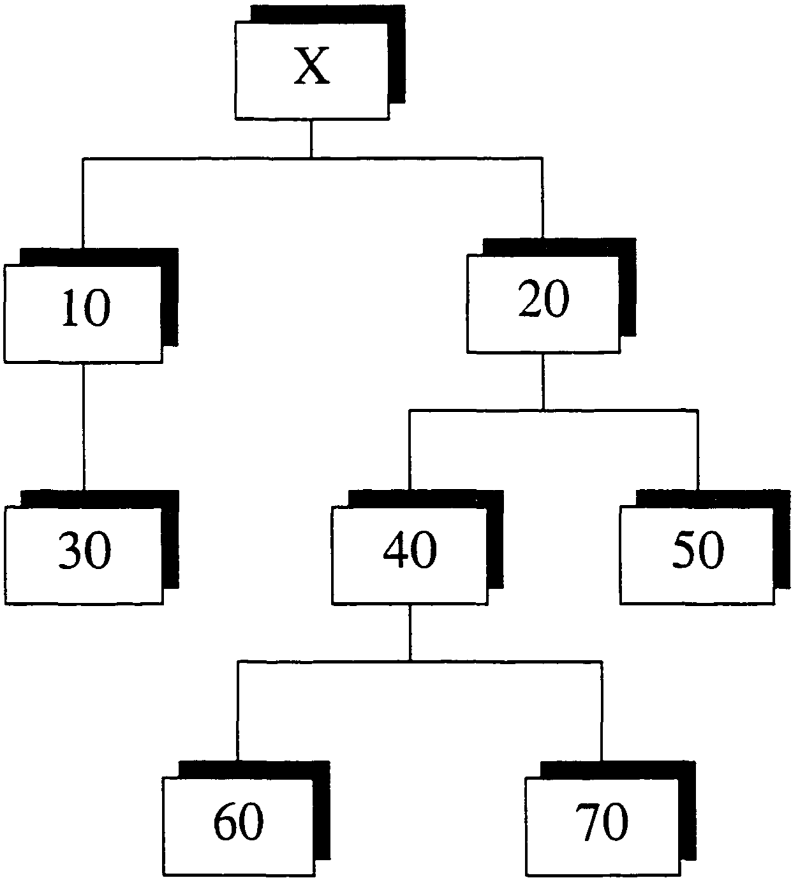


Figure 4. A typical bill of material (BOM).

Table 1 illustrates the material requirements plan for Part A. The gross requirements for Part A come from the production plan. Schedule receipts are the orders that are already in production. To calculate when an order needs to be placed, gross requirements are subtracted from the available balance and schedule receipts are added to it. In Table 1, for example, the on-hand balance is 400 units, the gross requirements for Week 1 are 120 units, so the projected on-hand balance for Week 1 is 280 units. The first uncovered demand in this example is in week 8 for 60 units. The lead time for Part A is 4 weeks; therefore, the order needs to be placed in Week 4 to cover the demand of 60 units in Week 8. The example above illustrates a simple MRP procedure. Because of space constraints, full discussion on the components of MRP procedure--netting, lotsizing, offsetting, and BOM exploding--is not covered in this research. For a full discussion of MRP, see Wight (1984) or Hopp and Spearman (1996).

Table 1

Time-Phased MRP Requirements Processing

Part A		Week							
		1	2	3	4	5	6	7	8
Gross requirements		120	120	0	0	120	150	0	150
Schedule receipts				200					
Projected available balance	400	280	160	360	360	240	90	90	-60
Planned order releases									

Advantages and Disadvantages

In the late 1960s and early 1970s, with the rapid advancement in computer technology, MRP took over the manufacturing industry. "Starting in the sixties and on into the seventies, the basic elements of an integrated production planning and control system known as MRP, were established" (Taylor, 1994, p. 8). Initially, computer-based MRP was thought to be so powerful that it made the classical methods of inventory management obsolete. One of the major advantages of the MRP system is its adaptability to dynamic changes and the ability to know what is required several periods in advance (Nagendra, 1995).

Many success stories are reported in the literature about MRP. According to Aggarwal (1985). MRP has indeed helped many organizations in the effort to reduce inventories and streamline scheduling. In discussing the advantages of MRP, Orlicky (1975) notes,

> this subject, broadly viewed, marks the coming of age of the field of production and inventory control, and a new way of life in the management of manufacturing business. In the area of manufacturing inventory management the most successful innovations are embodied in what has become known as the material requirements planning (MRP) system. (p. 4)

Umble and Srikanth (1990) state, "MRP became a crusade that helped to shift the emphasis away from the traditional 'just-in-case' inventory mentality and toward a manufacturing control system based on actual need dates and quantities" (p. 8).

Manufacturing organizations around the world invested billions of dollars and human resources in the implementation of MRP. In the United States alone, by 1989, sales of MRP software and support exceeded one billion dollars (Hopp & Spearman, 1996), but not all of the outcomes were successful. Taylor (1994), in summarizing the findings of Anderson, Schroeder, Tupy, and White (1982), reports that a great number of the firms that attempted MRP implementation were not always satisfied. According to Rice and Yoshikawa (1982), the weakest MRP area is in capacity planning. Nagendra (1995) also reports the inability of MRP to perform comprehensive capacity planning. Ashton, Johnson, and Cook (1990) likewise note part-shortage problems that disrupt operations due to MRP. Cox and Clark (1984) report other technical problems such as inventory management and infinite capacity assumption.

MRP has to be constantly modified to cope with the changing manufacturing environment. Over the years, many modules have been added to MRP giving it the more deserved name of manufacturing resource planning (MRP II). With MRP II, manufacturing interacts with other functions of the organization, such as accounting, finance, and human resource planning.

MRP has been an effective tool for several decades for many organizations, even with its built-in limitations. With the changing business environment, production planning and control methods also need to be changed. MRP-based production planning and control solutions are appropriate for organizations with repetitive manufacturing. However, the advantages of MRP for high-mix, low-volume manufacturing organizations are very limited.

Just-in-Time

Evolution

Even though the elements of just-in-time (JIT) has been around since the 1900s, the American manufacturing industry did not start paying serious attention to it until the late 1970s. "The first records of the JIT management philosophy stem from the efforts of Henry Ford and his assembly line operations" (Taylor 1994, p. 13). JIT received much attention in the Western manufacturing world during the early 1980s when a large number of books and articles were written on this subject. Between 1970 and 1991, more than 860 articles about the just-in-time philosophy were published in professional journals (Golhar & Stamm, 1991). The JIT system has become extremely popular in recent years and has been implemented in many kinds of companies around the world.

The just-in-time philosophy is based on the work of Taiichi Ohno of the Toyota Motor Company (Sugimoro, 1977). In the early 1980s, many American manufacturers regarded JIT as a Japanese manufacturing philosophy suited only for Japanese organizations. Initially, most Westerners viewed it as an inventory reduction system, beneficial only for large repetitive manufacturers (White, 1993). As more and more Western organizations successfully applied JIT principles, its benefits became evident for a wide range of manufacturing environments (Hall, 1983). U.S. managers also became more knowledgeable of JIT and described it as a holistic management approach consisting of various practices that contribute to the elimination of waste and a philosophy of continuous improvement of a manufacturing system (Hall, 1987; Schonberger, 1986; White, 1993). Today, many American manufacturing companies regard JIT as vital to their survival (Hobbs, 1997).

Functionality

The JIT philosophy is based on the concept of the elimination of waste in the system. JIT's purpose is to minimize in-process and final inventories (Hall, 1983; Monden, 1983). Early academic research focused on utilizing JIT systems within the internal manufacturing environment (Spencer, Daugherty, & Rogers, 1996), but this approach to JIT is evolving toward a broader concept--a total business philosophy. According to Ramasesh (1992), "JIT represents an integrative philosophy of operations which encompasses several functional systems both within the firm and outside of the firm" (p. 44).

Hall (1983), Sage (1984), and Heard (1984) all agree that the JIT philosophy is based on the pull method of production called "kanban." According to the APICS Dictionary (Cox et al., 1995), kanban is a "method of Just-In-Time production that uses standard containers or lot sizes with a single card attached to each. It is a pull system in which work centers signal with a card that they wish to withdraw parts from a feeding operation supplier" (p. 42). The APICS Dictionary defines pull system as "the production of items only as demanded for use, or to replace those taken for use. In material control, the withdrawal of inventory as demanded by the using operations. Material is not issued until a signal comes from the user" (p. 68).

Advantages and Disadvantages

One of the main advantages of JIT is its emphasis on shop-floor control rather than inventory control (Ohno, 1982). Im and Lee (1989) and Burnham (1987) report many benefits derived from the successful implementation of JIT, including improvements in production planning, improvements in MPS and MRP, and reduction in inventory. A study conducted by Gilbert (1990), of 250 American manufacturing organizations, found significant reduction in the investment of inventory associated with the implementation of JIT. Other benefits reported by researchers included reduced throughput time, improved labor productivity, improved quality, decreased inventory levels, and reduction in space required for operations (Celley, Clegg, Smith, & Vonderembase, 1986; Golhar, Stamm, & Smith, 1990; Hay, 1988).

Reducing inventory levels toward zero requires eliminating variability within a system. It is very difficult, if not impossible, to eliminate all the variability from a complex manufacturing system. To tackle this problem, managers on the shop floor would have to increase buffer size, which, in turn, would increase the work-in-process inventory. However, this goes against the JIT philosophy. According to Rice and Yoshikawa (1982), the weakest area in JIT is master production planning.

Another drawback is the time required for implementing JIT (Schonberger, 1986). For most Western organizations, the JIT implementation process spans many tedious years. Umble and Srikanth (1990) report four major limitations inherent in JIT and kanban:

> First, the number of processes to which JIT logistical systems such as kanban may be successfully applied is limited. Second, the effects of disruptions to the product flow under the kanban system can be disastrous to current throughput. Third, the implementation period required for JIT/kanban systems are often lengthy and difficult. Fourth, the process of continuous improvement inherent in the JIT approach is system wide and therefore does not focus on the critical constraints, where the greatest gain is possible. (p. 125)

Overall, the just-in-time approach to PP&C is based on the philosophy of elimination of all waste in the system. Organizations around the world have been implementing JIT for the last few decades and many of them have reported numerous benefits (Bartezzaghi & Turco, 1989; Burnham, 1987; Crawford, Blackstone, & Cox, 1988; Im & Lee, 1989). Even though there are some drawbacks to implementing JIT, organizations can gain competitive advantage once it is accurately implemented.

Constraints Management

Evolution

Originally known as theory of constraints, constraints management was developed at about the same time as the just-in-time philosophy started to make an impact on Western organizations. Goldratt developed an optimized production timetable (OPT) to assist a friend in the production and assembly of prefabricated chicken coops (Jayson, 1987). The OPT schedule enabled the producer to triple his production without increasing any human resources (Taylor, 1994). The logic behind the OPT software was not revealed because of proprietary reasons. Contrary to MRP philosophy, OPT assumes that production capacity is finite, restricted by the bottleneck operation (Dugdale & Jones, 1995). According to Nahmias (1989), OPT follows these nine principles:

1. Balance the flow, not the capacity.
2. The level of utilization of the non-bottleneck resource is determined not by its own potential, but by some other constraints in the system.
3. Utilization and activation of a resource are not synonymous.
4. One hour lost at the bottleneck operation is an hour lost for the total system.
5. An hour saved at the bottleneck is a mirage.
6. Bottleneck operations govern both throughput and inventory in the system.
7. The transfer batch might not, and many times should not, be equal to the process batch.

> 8. The process batch should be variable, not fixed.
> 9. Schedules should be established by looking at all of the constraints simultaneously. Lead times are the result of a schedule and cannot be predetermined. (p. 13)

According to Taylor (1994), constraints management was originally known as OPT, when it was first formulated in 1979. In 1982, the name was changed to optimized production technology, in 1984 to synchronous manufacturing, 1987 it became theory of constraints, and recently it became constraints management.

CM was originally regarded as a management technique suitable for the shop floor, but eventually it was used to manage and solve problems that extended far beyond that (Hobbs, 1997). CM applies the methods of science to the general problem of management (McMullen, 1997). Rack and Rack (1993) define it as follows:

> a thinking process used to analyze problems, create or choose appropriate solutions and get buy-in to achieve successful results. Although it is demonstrably very powerful, it is not difficult to understand. Because the process utilizes how man was designed to think, it works for almost everyone interested in tapping into his/her own abilities. The appropriate use of the thinking process significantly impacts the goal and is intrinsically rewarding to the one(s) using it. (p. 3)

Functionality

The main focus of the CM approach is to concentrate effort on the system's constraint(s). Goldratt (1990a) emphasized this point by addressing the need of focusing on a small portion of the system at a time. He went on to say, "spreading attention equally to all portions of the area means no concentration whatsoever, no focusing" (p. 58).

CM methodology is based on five focusing steps:

1. Identify the system constraint(s).

2. Decide how to exploit the system's constraint(s).

3. Subordinate all else to the constraint(s) of the system.

4. Elevate the system's constraint(s).

5. If, in step 4, the constraint has been broken, go back to step 1, do not let inertia become the system's new constraint.

A constraint is anything that limits the organization's achievement of its goal. If the scarce resources of an organization can be used to elevate the system's constraint(s), the organization's goal, which is to make money now and in the future, can be achieved successfully. Goldratt (1994) suggests that the five focusing steps follow a framework based on the following questions:

1. What to change (finding the core problem)?

2. What to change to (devise simple, practical solutions)?

3. How to cause the change (cause others to invent or discover the ideas)?

"The three elements of change are techniques for verbalizing our intuition so we can check its soundness and communicate it clearly to others" (Taylor, 1994, p. 21).

Goldratt has developed approaches to deal with problems using the Socratic method, rather than the more traditional Aristotelian way. According to Taylor (1994), Goldratt developed the following techniques to deal with change:

1. Effect-cause-effect: A technique for finding the core problem. This method allows for verbalization of intuition and its cause.

2. Evaporating clouds: A technique for stating a problem as a conflict. This allows for the conflict assumptions to be challenged. Faulty assumptions allow the problem to disappear.

3. Socratic method: This allows for others to invent or discover answers themselves and conceive ownership in them.

According to Woeppel (1991), all of the above techniques have proven to be very effective for increasing one's ability to verbalize intuitively. These techniques have been used in the manufacturing industry to develop and implement effective procedures.

Constraints management also addresses the issue of inventory in process with drum-buffer-rope (DBR) technique, defined by the APICS Dictionary as "the generalized technique used to manage resources to maximize throughput. The drum is the rate or pace of production set by the system's constraint. The buffers establish the protection against uncertainty so that the system can maximize throughput. The rope is a communication process from the constraint to the gating operation that checks or limits material released into the system to support the constraint" (Cox et al., 1995, p. 25).

CM emphasizes the need of inventory buffer in front of the constraint operation. DBR concentrates on managing the flow of products to meet the bottleneck constraint's needs. The buffer inventory in front of the constraint protects the constraint from stockouts due to upstream process interruptions. Since the bottleneck acts as a valve controlling the system's throughput, managing the bottleneck's throughput manages the system's throughput. To maximize the system's throughput, the bottleneck must utilize all of its available capacity.

The three commonly used PP&C methods discussed MRP, JIT, and CM, all offer some advantages for organizations engaged in various types of manufacturing activities. To choose any one of these three PP&C methods and apply it for all types of manufacturing environments would not be an easy task, especially for managers with little exposure to academic research. The present research would help managers in repetitive industry to compare and evaluate the three popular PP&C approaches and choose the one that would work best for their manufacturing environment. The next section discusses genetic algorithms, the history and functionality.

Genetic Algorithms

Genetic algorithms are becoming a widely used tool for difficult optimization problems (Bennett, Ferris, & Ioannidis, 1991; Goldberg, 1989; Grefenstette, 1987). In recent years, GA have received remarkable attention all over the world, a fact reflected in the amount of literature published on this topic in the last few years (Back, 1996). Researchers have explored the possibilities of GA applications in various fields, including game theory, process planning, classifier systems, machine learning, and function optimization (Crossley, 1995). The use of GA for scheduling in manufacturing has also been explored by many researchers (Bagchi, Uckun, Miyabe, & Kawamura, 1991; Davis, 1985, 1991; Nissen, 1993; Whitley, Starkweather, & Fuquay, 1989).

History

The history of genetic algorithms goes back more than four decades (Back, Hammel, & Schwefel, 1997). Bremermann (1962, 1967, 1968, 1973), Fraser (1957, 1962, 1968), Reed, Toombs, and Barricelli, (1967), and Holland (1969, 1975) report early research related to genetic algorithms. Genetic algorithms in the present form were developed by Dr. John Holland, computer scientist and psychologist at the University of Michigan. Dr. Holland, along with his students and colleagues during the 1960s and 1970s, developed the research area of artificial intelligence (AI), now known as genetic algorithms. His book Adaptation in Natural and Artificial Systems (1975) is considered to be the starting point of almost all known applications and implementations of genetic algorithms (Back, 1996).

Research in the field of artificial intelligence is based on the idea that "evolution could be used as an optimization tool for engineering problems" (Mitchell, 1996, p. 5). The common theme in almost all evolutionary systems is the belief that it is possible to evolve a population of candidate solutions to a given problem, using operators inspired by natural genetic variation and natural selection (Chambers, 1991). Many researchers have expanded on Holland's research on genetic algorithms since 1975.

The growing complexity of scheduling and sequencing problems in manufacturing has led many researchers to experiment with genetic algorithms as an optimization tool. Genetic algorithms have been used to solve scheduling problems with increasing frequency since the early 1980s. Various researchers (Bagchi et al., 1991; Cleveland & Smith, 1989; Davis, 1985; Nakano & Yamada, 1991; Syswerda, 1991; Whitley et al., 1989) have reported experimentation with genetic algorithms to solve scheduling problems.

Functionality

The genetic algorithm is a probabilistically guided search method, "developed originally in the 1970's as a computer science tool to improve programming structures and performance" (Holland, 1992, p. 66). Chambers (1991) defines GA as a "problem solving method that uses genetics as its model of problem solving" (p. 13). GA are search techniques based on the mechanics of natural selection and genetics, and they involve a structured yet randomized information exchange resulting in the survival of the fittest amongst a population of string structures. GA operates on a population of structures that are fixed-length strings representing all possible solutions to a problem domain (Mars, Chen, & Nambiar, 1996). Genetic algorithms work by mimicking the "survival of the fittest" patterns of natural selection and reproduction similar to those in biological populations (Crossley, 1995).

Davis (1991) identifies four features of the evolution process that are the bases of genetic algorithms. These four features are as follows:

1. Evolution is a process that operates on chromosomes rather than on living beings they encode.
2. Natural selection is the link between chromosomes and the performance of their decoded structures. Process of natural selection causes those chromosomes that encode successful structures to reproduce more often than those that do not.
3. The process of reproduction is the point at which evolution takes place. Mutation may cause the chromosomes of biological children to be different from those of their biological parents, and recombination processes may create quite different chromosomes in the children by combining material from the chromosomes of two parents.
4. Biological evolution has no memory. Whatever it knows about producing individuals that will function well in their environment is contained in the gene pool the set of chromosomes carried by the current individuals--and in the structure of the chromosome decoders. (pp. 2-3)

The features described above allow genetic algorithms to solve complex problems without having any knowledge of the problem or the search space. Michalewicz (1994) identifies five components that must be contained by genetic algorithms:

1. A genetic representation for potential solutions to the problem
2. A way to create an initial population of potential solutions
3. An evaluation function that plays the role of the environment, rating solutions in terms of their fitness
4. Genetic operators that alter the composition of children
5. Values for various parameters that the genetic algorithm uses. (p. 6)

The three basic operators that are found in every genetic algorithm are (a) reproduction, (b) crossover, and (c) mutation.

Reproduction. The reproduction operator permits individual strings to be copied in the next generation. The string's chance to be copied to the next generation depends on its fitness value calculated from a fitness function. The reproduction operator chooses strings that were placed in the waiting pool for each generation. The next generation is based on this pool.

Table 2 demonstrates that string 01100 is the best fit. This string should be selected for reproduction approximately 66% of the time. String 01101 is the second best fit and should be selected 21% of the time. And string 10101, the weakest, should be selected only 13% of the time.

Table 2

Fitness Test

String	Fitness value	%
01101	8	21
10101	5	13
01100	25	66

Crossover. After the mating pool is created through the selection operator, the next genetic algorithm operation is called crossover. In biological terms, crossover occurs when two parents exchange parts of their corresponding chromosomes to produce an offspring. Figure 5 illustrates the crossover operation within genetic algorithms.

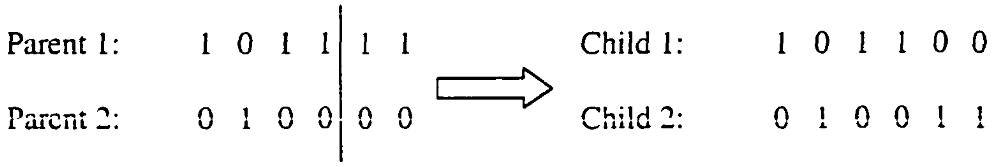


Figure 5. Crossover operation.

Each child in the example receives four of the six parts of each parent's genetic material. In a genetic algorithms search, crossover is performed until a new population is created, and then the cycle starts again with a new selection. According to Davis (1991), crossover is an extremely important component of a genetic algorithm. Use of the crossover operator distinguishes the genetic algorithm from all other optimization algorithms.

Mutation. The mutation operator brings a certain amount of randomness to the genetic search. Mutation can help the genetic search to find solutions that crossover alone might not encounter. Selection and crossover operations in a genetic search can generate a large quantity of different strings. However, depending on the initial population of the search, the resulting strings may not have enough variety. The mutation operator can offset this shortcoming. When a genetic algorithm performs a mutation, it randomly changes the element value to a new one. If, to use the example in Figure 5, Position 5 of the Parent 1 string were mutated, the resulting string would be 101101. In the binary strings, 0s are changed to 1s and 1s are changed to 0s.

There are significant differences between genetic algorithms and other optimization tools. Crossley (1995) identifies four major differences between calculus-based optimization and genetic algorithms as follows:

> 1. GA works with a coding of the design variables and parameters in the problem, rather than with the actual parameters themselves.
> 2. GA makes use of a population-type search. Many different points are evaluated during each iteration, instead of moving from one point to the next.
> 3. GA needs only a fitness or objective function value. No derivatives or gradients are necessary.
> 4. GA uses probabilistic transition rules to find new points for exploration rather than using deterministic rules based on gradient information to find new design points. (p. 24)

One of the most significant advantages of using genetic algorithms is flexibility and adaptability to the problem at hand (Back et al., 1997).

Foundational Study for Current Research

In an earlier study, which provided the basis for the present research, Choudhry (1998) investigated the current status of production planning and control methods at an engine manufacturing plant (EMP) of a midwestern manufacturer of agricultural equipment, hereafter referred to as MMAE. In that study, the writer focused on 11 questions dealing with current methods and problem areas. The results are reported under the following listing of those 11 research questions.

Current Production Planning and Control Methods

1. What are the production planning and control (MRP, JIT, CM) methods currently being used at EMP?

Production planning is the primary responsibility of the logistics manager, who reports directly to the plant manager. The seven employees in the production planning department include a supervisor of production planning and an employee who performs the daily final assembly scheduling (line-up). Three employees are involved in the distribution of the daily schedule to the shop floor. One employee is responsible for the inventory accuracy, and the seventh employee is in charge of fulfilling service store requirements. The purchasing department orders components based on the master schedule in the MRP and is also responsible for component sourcing and price negotiations.

The key performance measurements for the logistics department were not clear because at the time of this study, the department had only been in existence for a few months. The key performance measurements for the production planning supervisor and the department are (a) due date performance as a percentage of total order shipped (for the three months prior to this study, this figure was close to 100%); (b) customer acceptance; and (c) a target inventory as a percentage of sales.

In late 1979 EMP developed and implemented an in-house material requirements planning system, which has undergone significant modifications throughout the following years. The system continues to be modified at the present time as the need arises. MMAE is in the process of implementing an enterprise resource planning (ERP) system by SAP throughout its plants around the world. At its midwestern locations, this implementation will start in the middle of 2000 and will be fully implemented in about two years.

Accuracy of the bill of material (BOM) is around 96%, and part routing accuracy is 95%. Changes are made daily to the bills of material. Communication seems to be the main problem between the specification and engineering departments. Routings are not changed frequently, two per part for new engines and about 5% for the repetitive builds. For the inventory management, an ABC analysis was performed, and EMP uses six categories--A, B, C, D, E, and F. A cycle counting system is in operation, which is a physical count of inventory that is conducted every quarter; once a year, auditors from the company corporate office count the inventory. Inventory turns are about 13 per year. Inventory breakdown at EMP is as follows: raw, about 34.4%; WIP, 57.1%; and finished goods, about 8.5%.

The current MRP system is regenerated on a weekly basis and is using weekly buckets for requirements. Daily net changes for the master production schedule and inventory netting are performed. Even though the logistics manager is pleased with the accuracy of the MRP reports, he considers them very time insensitive. In the new global economy, customer requirements are being changed regularly without regard to weekly buckets.

EMP has been relying on the MRP system for production planning and control activities since its implementation in 1979. Some aspects of just-in-time (kanban) are also being implemented in a few subassembly work centers. Constraints management is not being practiced formally, but management does consider the two bottleneck operations in the plant when production planning activities are undertaken. The management at EMP is trying to minimize reliance on MRP. Many new projects are under way to develop Excel-based tools for PP&C.

2. What methods are currently being employed to develop the master production schedule at EMP?

The process of master scheduling at EMP begins when an order is received from the customer with the required ship date. For interfactory customers, the common worldwide interfactory system (CWIS) is used; for various original equipment manufacturers (OEM), the complete goods order management and reporting system (COMAR) is utilized. The difference between the two types of orders is that options are attached to OEM orders. Engines built for each OEM customer are unique, whereas engines built for interfactory customers are build via repetitive manufacturing methods.

The master scheduler enters these orders into the master schedule system and accounts for the number of days it takes to build an engine (lead-time). After the leveling activity is completed, information is passed on to a planner to perform the line-up. The same information is entered into the system's material requirements planning (MRP), which in turn passes it to CPS (common purchasing system), so the purchasing department is informed when to procure the parts.

MRP generates the shop production schedule (SPS) for the machining department, informing them when to start production for these parts based on the parameters maintained in MRP (lead-time, scrap %, order policy, etc.) by the planners in the machining department. The planners in the machining department report to the machining business unit leader. MRP information is driven by the line-up for 20 days and the master schedule beyond the 20-day time frame.

If a shortage is foreseen for any parts, the critical shortage report comes into play. When purchasing cannot procure a part or machining cannot manufacture one, that information is generated on the critical shortage report and passed on to a scheduler.

Most of the computer systems used at EMP are "legacy" systems. They were called common systems (MRP, COMAR, etc.) because they were supposed to work in a uniform manner for all MMAE units around the world. If any changes were proposed in the system, those changes had to be approved by a committee consisting of members from each plant. If the changes were approved by the committee, each unit incorporated them into the system. However, in the last few years, this situation has changed. Now each unit makes changes independently. As a result, MMAE does not pay headquarters for system support, and the company is moving toward implementation of an enterprise resource planning system by SAP.

When there are changes to be made in the engineering specification of a particular engine, the product engineering center (PEC) provides this information to the head of the specification department. This department works through the approved specifications and loads them in the system along with the effectivity dates. The information is routed to appropriate departments affected by the changes. If the changes have to do with options for OEM customers, that information also needs to be routed through the marketing department, so they can forecast for parts or options.

Of the engines manufactured at EMP, 85% are sold to interfactory customers, and the rest are sold to OEM customers. These engines are used in tractors, combines, and other agriculture and construction equipment for the interfactory customers. Interactions with dealers are then minimal; the marketing department, specifically the OEM representative, interacts with OEM dealers and customers.

3. What methods are currently being employed to plan production priority at EMP?

The 85% of engines produced for interfactory customers are manufactured via repetitive build, whereas the rest of the engines, for OEM customers are customized with many options for each model. The MRP process of explosion and netting lose this identity. Production orders for the shop floor are created by the MRP based on the lead times of each component.

Even though MRP creates shop orders for a majority of the manufactured components, EMP has been in the process of establishing kanbans, in this case a replenishment cycle of about two to three days for 80% of the components. Priority planning at EMP is accomplished through the use of the MRP trigger system for purchased components. Kanban is used to plan priorities for 50% of in-house manufactured parts. Management at EMP has initiated projects in the last two months to include all in-house parts for kanban delivery.

The primary priority planning document used for the final assembly line is the report generated manually by the production scheduler titled "daily line-up". This report lists all engines to be built in the sequence that day, based on customer ship orders. The report is distributed to 60 work centers on the final assembly and subassembly lines. The new logistics manager has initiated many projects to streamline the master scheduling and daily line-up process at EMP. In the new PP&C process, distribution of daily line-up sheets will be either eliminated or minimized. EMP is in the process of implementing kanbans for the majority of the subassembly stations.

4. What methods are currently being employed to plan production capacity at EMP?

Capacity is defined at EMP by the number of engines built per day. Long-term capacity planning occurs during the next fiscal year's production planning process. Capacity has never been a major issue at EMP. This facility was built to produce 300 engines per day, but demand for engines has never exceeded that number. Production can be easily increased, if the forecast indicates a growth in sales.

EMP operates on two shifts for the final assembly on a five-day-per-week basis; however, it is possible to drop to one shift if the demand declines for a few weeks. Because of the current union contract, MMAE's four local plants cannot lay off any hourly employees. When production is cut, shop floor employees are put in a "resource pool" which is comprised of extra employees and used for rapid continuous improvement (RCI) projects.

Short-term capacity planning for the assembly areas is accomplished through the use of a final assembly schedule for the following 20 days and a computer program (Workforce & Machine Load) that converts units into the workforce required. Adjustments to the final assembly schedule are rarely made at the final assembly line due to the unavailability of operators.

The test and paint departments are the current constraints at EMP; many times, test and paint problems cause delays in customer shipments. The test and paint departments run on a three-shift, five days/week basis. Only eight test cells must handle about 171 engines per day. Capacity for the paint department is 30 engines per shift, 90 engines per day. About 60% of the engines manufactured at EMP require paint. Capacity is adjusted by adding overtime shifts on Saturdays and Sundays.

5. What methods are currently being employed to control production priority at EMP?

In the final assembly and subassembly areas, priority is controlled by the daily line-up schedule. Once the daily line-up is created for the following three days, unique serial numbers are assigned to each engine, and serial plates and serial tags are generated. If there is a change in the build schedule, the master scheduler has to make manual changes on the distributed line-up sheets. There are about 10 changes per week in the final assembly line-up.

Order changes are established through negotiations between the EMP management and its interfactory and OEM customers. Both types of customers can change their orders in the CWIS beyond 90 days without approval from the master scheduler. If changes are made within 90 days, customers must request the changes through CWIS, which generates an "action file." The changes in the action file have to be reviewed and accepted by the master scheduler. If EMP cannot fulfill the requirements, the master scheduler proposes a date when those requirements can be fulfilled. This interaction with the customer continues until both parties agree on a mutually satisfactory date. Changes in customer requirements affect 13% of the total sales at EMP.

6. What methods are currently being employed to control production capacity at EMP?

Department supervisors control capacity at the two bottleneck areas, test and trim and paint, on a daily basis along with the assembly general supervisor. Overtime is scheduled as required if production exceeds capacity. Assembly supervisors request overtime authorization from the plant manager. The test and trim department schedules overtime on a regular basis to avoid any delays in shipping. The new logistics manager has initiated a project to streamline these departments.

Identification of the current methods of production planning and control practiced at EMP was not an easy task. Interviewees often could not describe the current process in place. The researcher had to illustrate and explain the majority of the production planning and control terminology to extract information. In the next section the problems inherent in the current production planning and control system at EMP are presented.

Problem Areas by Production Function

1. What problems are currently being encountered in master production scheduling at EMP?

The first area of concern for management regarding the master production schedule is the reliance on legacy computer systems, CWIS and COMAR. These systems are very labor intensive, requiring too much duplication of work by the master scheduler and the schedulers. A second area of concern is the limitations of the MRP system, which is unable to support changes during the week. Changes in the master production schedule only become apparent after the weekend report is generated by the system. Another concern is the development of the MPS by the master scheduler. According to the master scheduler, no formal procedure is in place for the development of the MPS for the following fiscal year. The master scheduler uses a rolling 12 months for the development of the MPS instead of using a fiscal year.

2. What problems are currently being encountered in planning production priority at EMP?

The first area of concern is the limitations of the MRP system and the execution of the master production schedule. MRP is limited to weekly buckets, which create unseen changes made during the week by the master scheduler. Management has implemented controlled delivery for a few subassembly work centers to establish priorities. A final assembly schedule is prepared from the master production schedule and is also used to identify the priorities in machining. The final assembly schedule, which is in weekly buckets, is also used by the scheduler to line-up engines for the next 20 days. The line-up schedule is used to generate the part shortage list, "critical shortage day-one." Another area of concern is the marketing department's ability to alter relative production priorities as required for OEM customers. Reprioritization in the final assembly schedule also creates problems for the machining department. A third problem is the long lead-times for three critical parts: turbo, injection pump, and pistons. Lead-time for these parts averages about 120 days. Long lead-times limit the flexibility of MMAE to respond to customer changes in requirements.

3. What problems are currently being encountered in planning production capacity at EMP?

Capacity planning at EMP occurs concurrently with master production scheduling. Long lead-times for component parts is a concern for management. Due to the union contract, there is a long lead-time to change labor capacity relative to the order horizon. Another concern for management is the shut-down days of sister factories. Various interfactory customers plan their shut-down days/weeks according to their own needs. This creates changes in the requirement dates, and the master scheduler has to pull ahead orders and repeat the leveling activity.

4. What problems are currently being encountered in controlling production priority at EMP?

The key area of concern for priority control occurs at the two bottleneck areas: test and paint. Daily monitoring by the department supervisors and the general supervisor of assembly is the control method used for priority control in these areas. In these two departments reprioritization is common to meet customer ship dates. Another concern is the amount of changes in customer orders, which is about 13% monthly. Changes in customer orders can require the reprioritization and expediting of orders to make sure customer delivery dates are met. Frequency of set-up required on the assembly line is also problematic. The set-up frequency and time are factors not taken into considerations in the MRP calculations. Since the early stages of implementation, problems related to kanban have not been addressed by EMP.

5. What problems are currently being encountered in controlling production capacity at EMP?

Changes in available capacity at EMP occur due to machine down-time or changes in customer requirements. Capacity problems are typically resolved by using overtime or reassigning workers to areas where they are needed. Overtime in any assembly area must be approved by the factory manager. Department supervisors adjust workforce assignment, if allowed by the union contract, to resolve capacity problems.

During the course of this research, the logistics manager initiated several projects to address these problems and streamline the production planning process. A number of these projects will take more than a year to make an impact on the current production planning process.

Summary

This chapter examined the literature pertinent to the three most common production planning and control methods: material requirement planning (MRP), just-in-time (JIT), and constraints management (CM). The history, functionality, and advantages/disadvantages of each were discussed. The origin of genetic algorithms, as well as a discussion of the functionality of this method, was presented. One of its most significant advantages, it was pointed out, is flexibility. The findings of a foundational study for the current research, both current production planning and control methods and problems areas by production function, were reported.

CHAPTER III

RESEARCH DESIGN AND METHODOLOGY

Research Design

This experimental research (proposed method/present method) was designed to identify production planning and control (PP&C) constraints and to develop and validate scheduling and sequencing model based on constraints management and using genetic algorithms. The five research questions stated in Chapter I were used as a basis for this study.

1. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the cycle time for the final assembly line and four downstream processes at an engine manufacturing plant (EMP) of a midwestern manufacturer of agricultural equipment (MMAE)?

2. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the queue size for the final assembly line and four downstream processes at EMP?

3. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the utilization of work centers in the final assembly line and four downstream processes at EMP?

4. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the flow rate of engines through the final assembly line and four downstream processes at EMP?

5. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the total output of engines through the final assembly line and four downstream processes at EMP?

Independent Variable

The independent variable in this research is the method of scheduling and sequencing. The control condition is the current scheduling and sequencing method, and the experimental condition is the proposed scheduling and sequencing model based on constraints management and utilizing genetic algorithms.

Dependent Variables

The dependent variables in this research are as follows:

1. Cycle time of engines for the final assembly line and four down-stream processes
2. Queue size in front of four downstream processes after final assembly line
3. Utilization of work centers in the final assembly line and four downstream processes
4. Flow rate of engines through the final assembly line and four downstream processes
5. Total output of engines through final assembly line and four downstream processes

Present Method / Proposed Method

Control Group

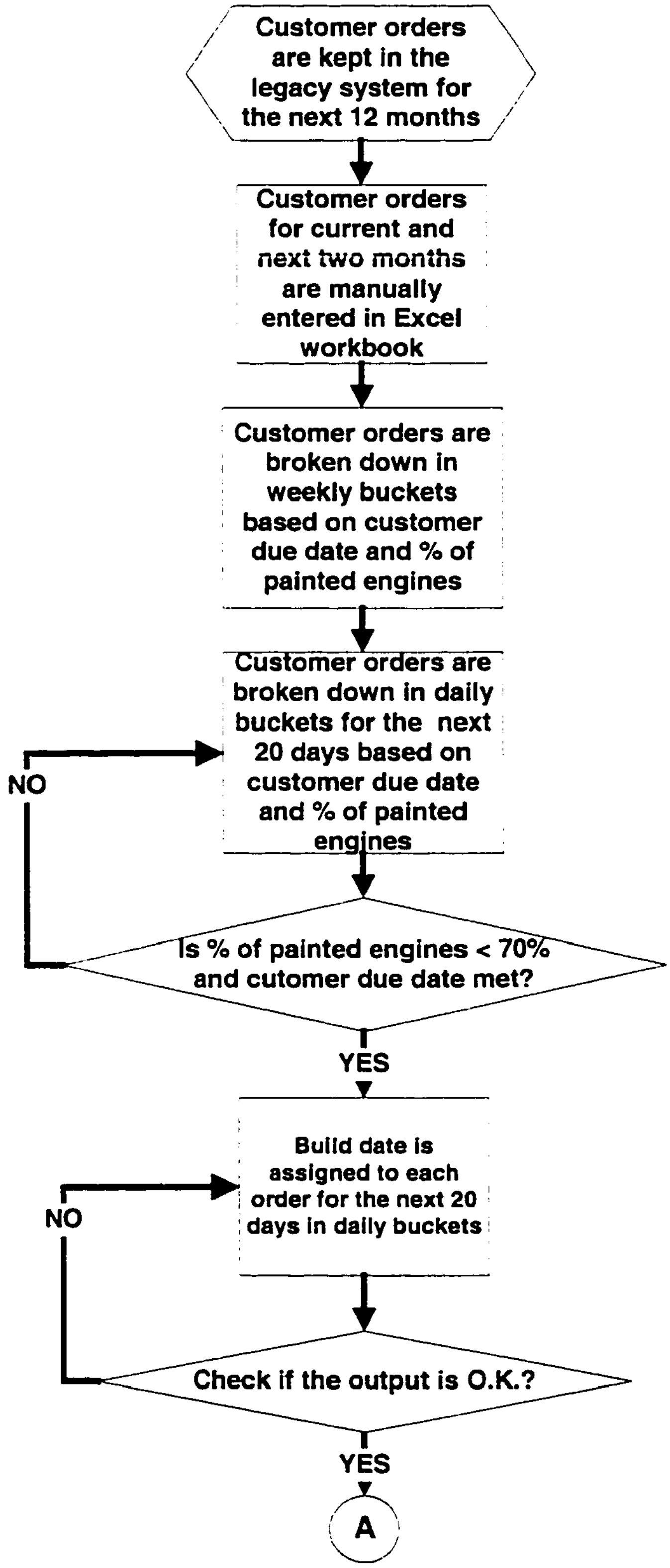
The process of master scheduling at EMP begins when an order is received from the customer with the required ship date. For interfactory customers, the common worldwide interfactory system (CWIS) is used; for various original equipment manufacturers (OEM), the complete goods order management and reporting system (COMAR) is utilized.

The master scheduler enters these orders into the master schedule system and accounts for the number of days it takes to build an engine (lead-time) for the next 12 months (Figure 6). Customer orders for the next two months are manually entered in an Excel workbook. These orders are broken down from monthly buckets into weekly buckets for these two months based on the customer due date and percentage of painted engines. An Excel file containing customer orders for the next four weeks is passed on to the line-up scheduler.

Customer orders for the next four weeks are broken down into daily buckets based on the customer due date and percentage of painted engines. A manual check is performed after the daily breakdown operation to confirm the percentage of painted engines is less than 70%. If the daily percentage of painted engines is less than 70% and customer due dates are met, a production build date is assigned to each customer order for the next 20 production days. If the daily percentage of painted engines is greater than 70%, assigned dates are adjusted manually and the schedule is frozen for the next production day. The next day's frozen schedule is manually sequenced in small batches.

The build schedule is generated and distributed on the shop floor for the next production day.

Flow chart for the control group was reviewed by the key expert in the area of production planning and control at EMP (D. Eck, personal communication, April 24, 2000), who confirmed that the flow chart is an actual representation of the current master scheduling and line-up process at EMP.


Customer orders are kept in the legacy system for the next 12 months
Customer orders for current and next two months are manually entered in Excel workbook
Customer orders are broken down in weekly buckets based on customer due date and % of painted engines
Customer orders are broken down in daily buckets for the next 20 days based on customer due date and % of painted engines
NO
Is % of painted engines < 70% and cutomer due date met?
YES
Build date is assigned to each order for the next 20 days in daily buckets
NO
Check if the output is O.K.?
YES
A

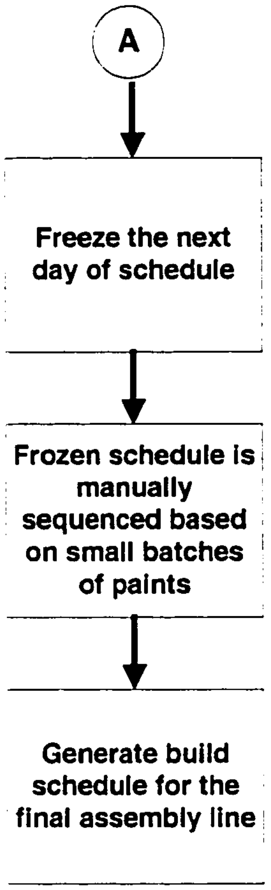
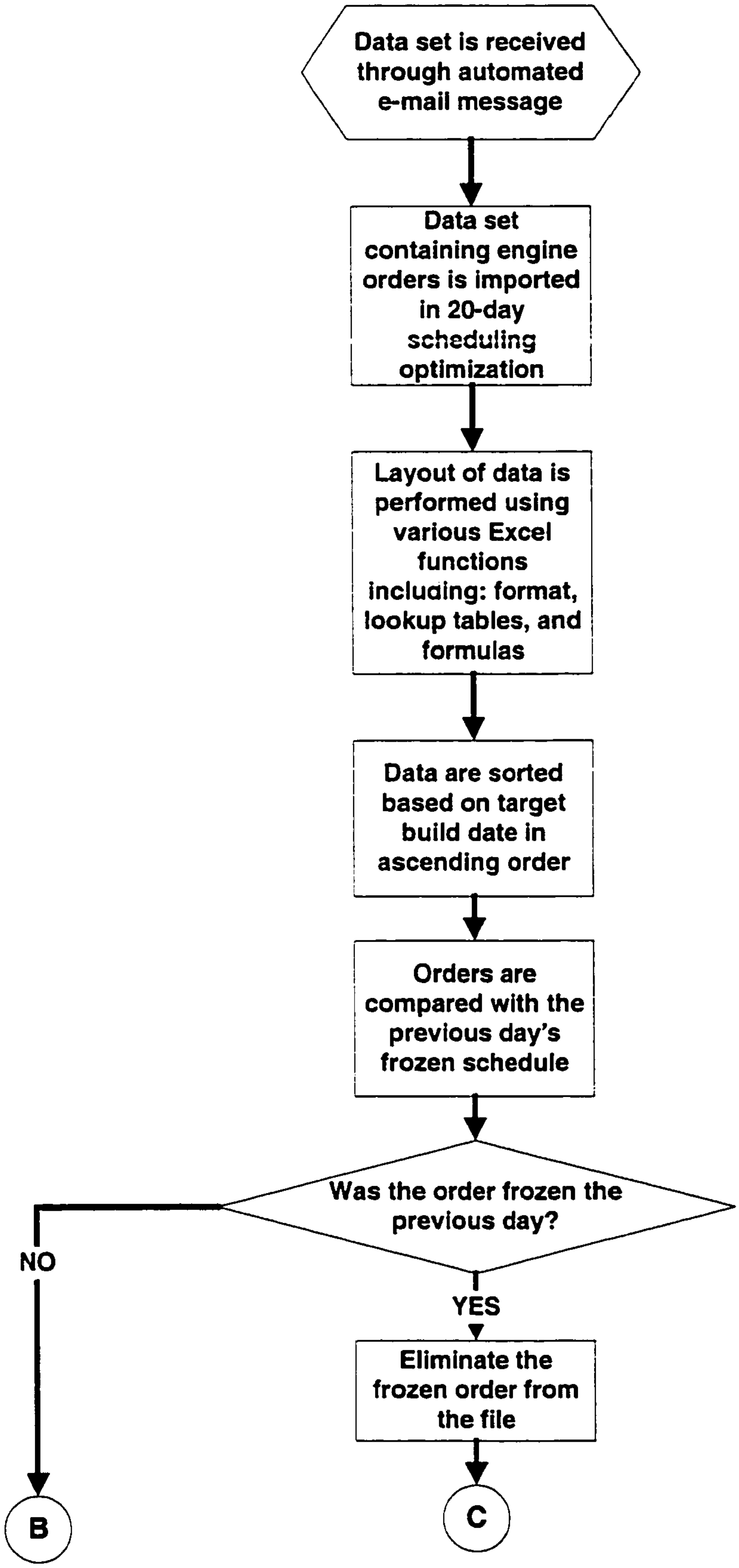


Figure 6. Control group flow chart for the master scheduling and line-up process.

Experimental Group

A flow chart for the experimental group is illustrated in Figure 7. This flow chart was also reviewed by the key expert in the area of production planning and control at EMP (D. Eck, personal communication, April 24, 2000). Detailed discussion about the new master scheduling and line-up process is presented in the next section. Snapshots of each Excel worksheet are described with the various Excel functions that were used for the development of the scheduling and sequencing model in Excel.


Data set is received through automated e-mail message
Data set containing engine orders is imported in 20-day scheduling optimization
Layout of data is performed using various Excel functions including: format, lookup tables, and formulas
Data are sorted based on target build date in ascending order
Orders are compared with the previous day's frozen schedule
Was the order frozen the previous day?
NO
YES
Eliminate the frozen order from the file
B
C

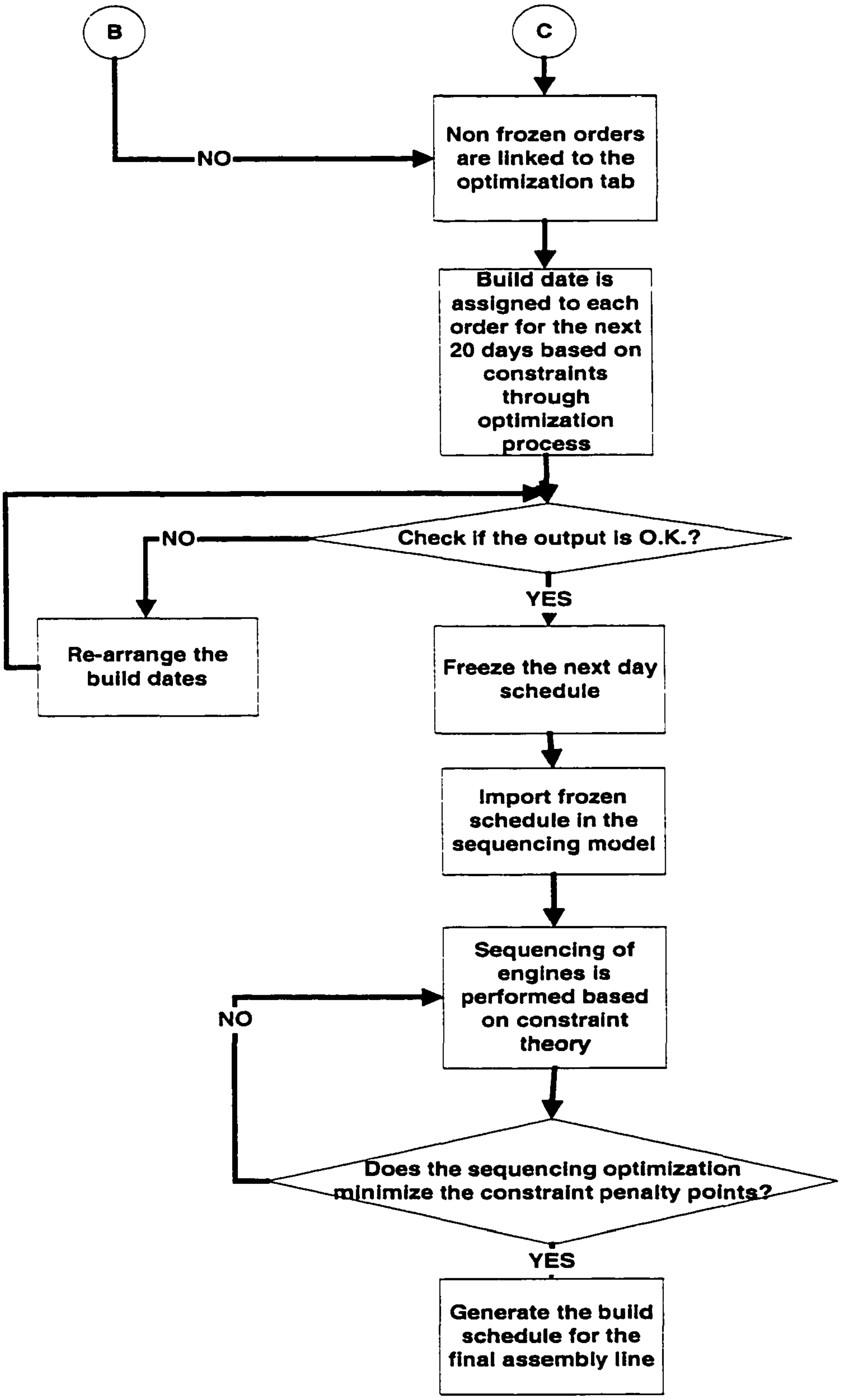


Figure 7. Experimental group flow chart for the master scheduling and line-up process.

Lack of time and capital resources limited the complete implementation of constraints management five focusing steps of: (a) identify the constraint, (b) exploit the constraint, (c) subordinate all other operations to the constraint, (d) elevate the constraint, and (e) avoid inertia. Three of the five focusing steps were used to develop the proposed scheduling and sequencing model at EMP; (a) identify the constraint, (b) exploit the constraint, and (c) subordinate all other operations to the constraint. Scheduling and sequencing methods used for the proposed model were based on drum-buffer-rope (DBR), which " is the core of the scheduling procedure under TOC" (Duclos & Spencer, 1995, p. 176). Figure 8 presents a generic version of the model used.

The paint operation was identified as the constraint at EMP, as indicated in step 1 of the focusing steps of constraints management. The paint operation dictates the launch schedule of engines at the final assembly line, thus fulfilling the definition of "drum" according to the APICS Dictionary: "the drum is the rate or pace of production set by the system's constraint" (p. 25). According to the Schragenheim and Ronen (1990), "drum is the exploitation of the constraint of the system." Using the drum to determine the pace of the system and its capacity accomplishes step 2 (exploit the constraint). A constraint buffer, which provides time to protect constraint from disruptions, was established after the custom trim operation. In the DBR method, the rope is a communication process from the constraint (paint operation) to the gating operation (final assembly line) that checks or limits material released into the system to support the constraint.

The flow of engines is depicted in Figure 9. After the engines leaves the final assembly line, a decision is made on space availability in test cells. If space is available, an engine is moved into a test cell; if not, the engine goes to temporary storage location.

After the engines are tested, they need to go through head torque operation. Once they pass this point, a decision is made on the routing of engines. Engines that are to be painted need to proceed first through custom trim, then paint and final trim areas. Non-paint engines go directly to final trim before they are warehoused. If both the custom trim and final trim queues are full, the head torque operation is shut down and the operator helps the test cell operators.

Figure 10 shows the time needed at each operation for the process of engines. A buffer of seven hours was created before the paint operation to protect the constraint from disruptions. The size of the constraint buffer was determined by managerial evaluation including operators in the paint operation and their supervisor opinions.

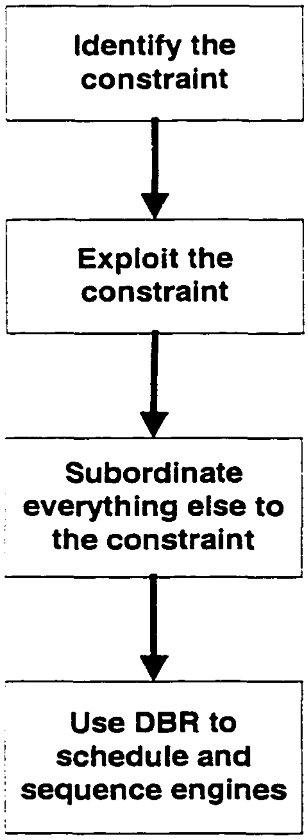


Figure 8. The application of CM at EMP.

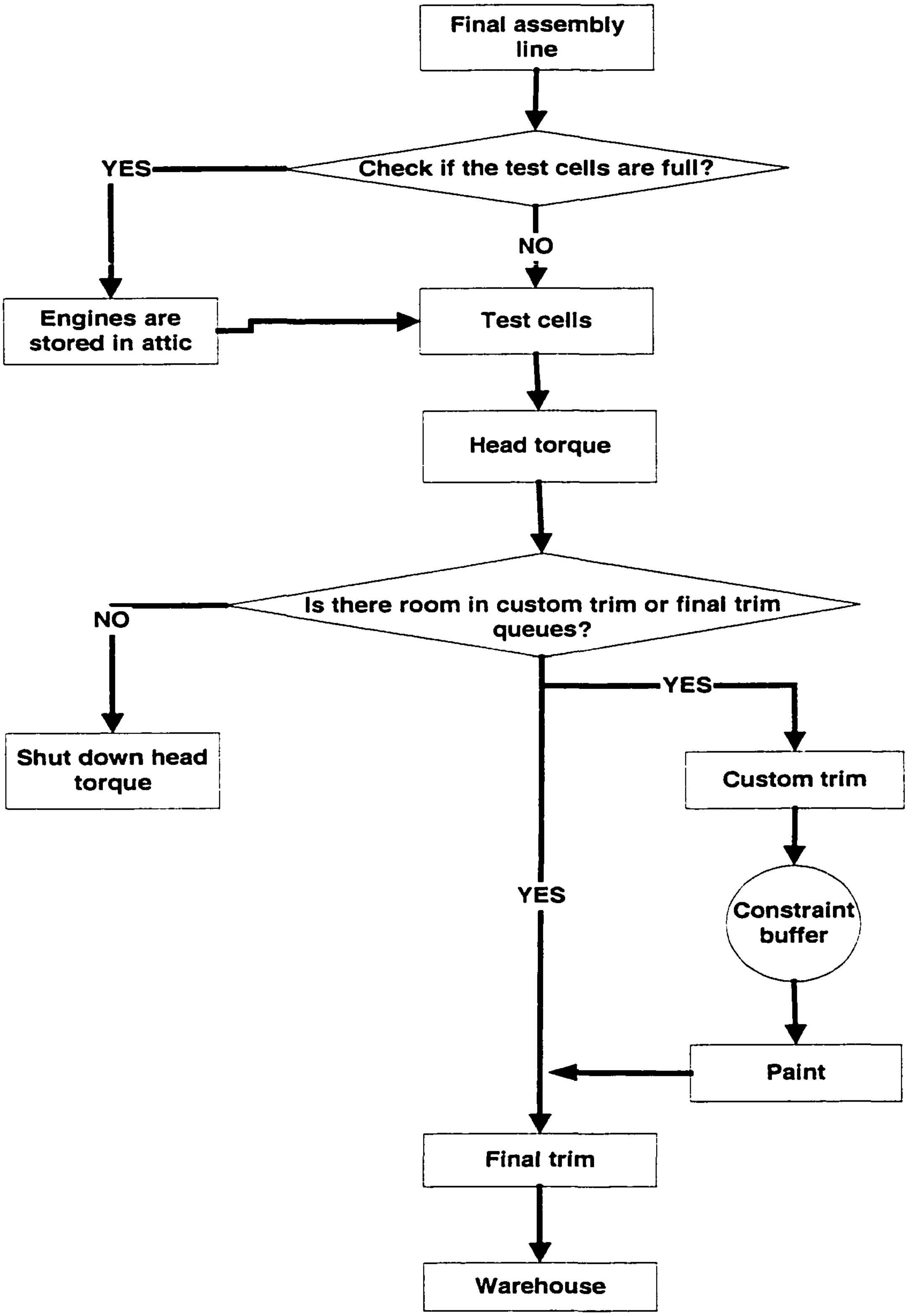


Figure 9. Flow of engines at EMP.

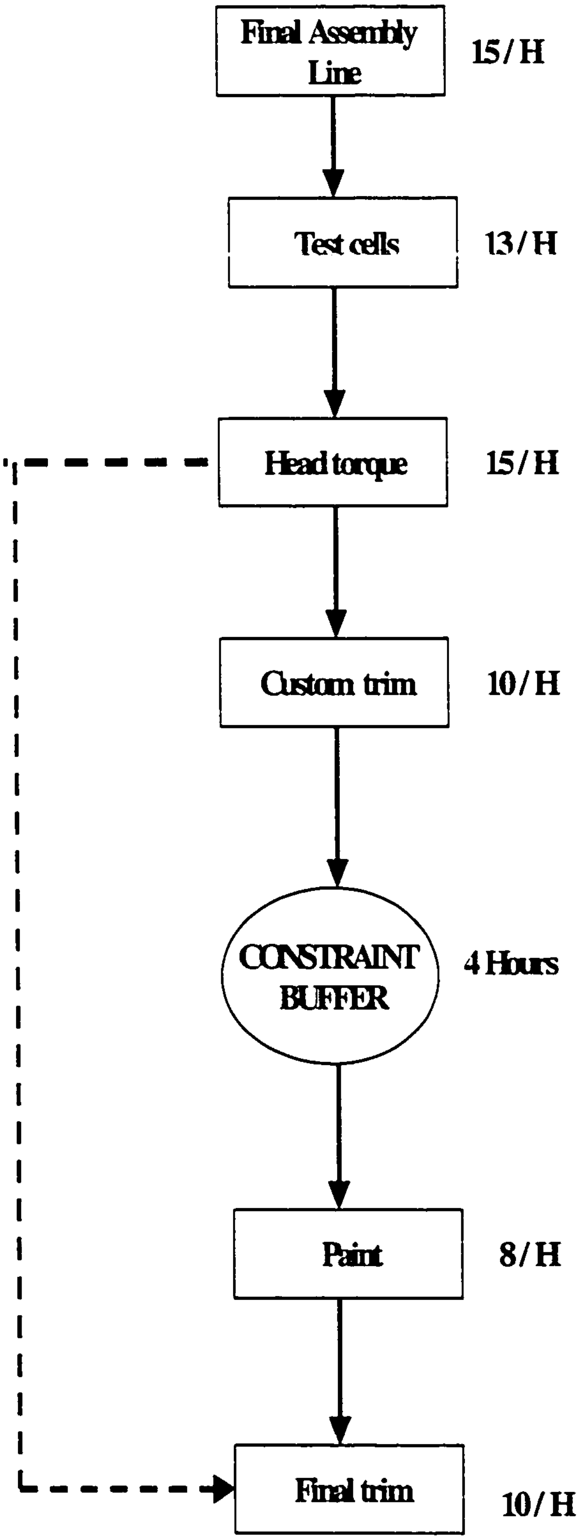


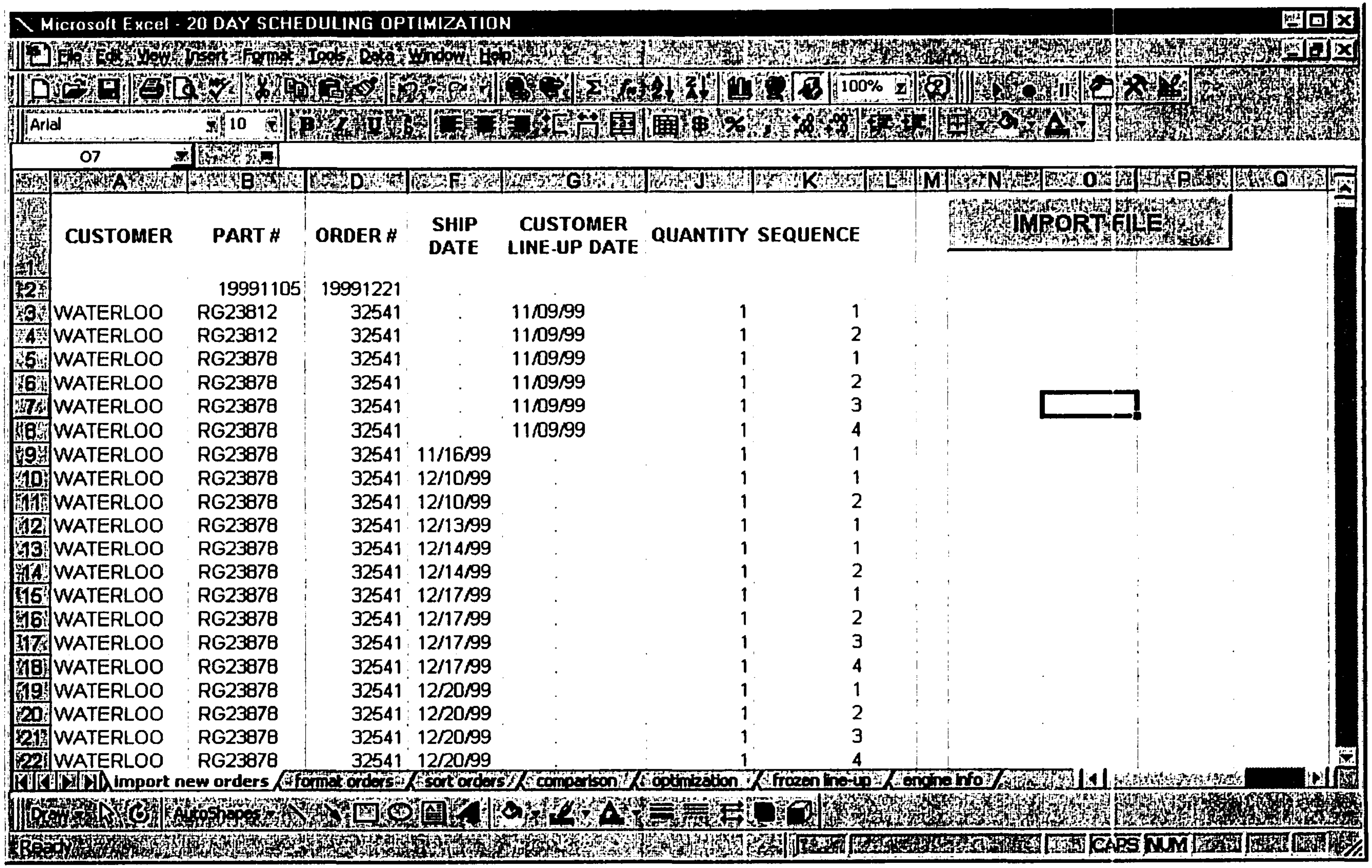
Figure 10. Flow rate of engines at EMP.

Scheduling model. Two-part model was developed in Excel, one part for scheduling and the other part for sequencing engines in order to utilize CM methods. In the scheduling part of the model, engine orders are assigned a date to be built based on the following constraint criteria:

1. Customer due date
2. Available capacity in final assembly line
3. Available capacity in the test department
4. Available capacity in the customer trim area
5. Available capacity in the paint area
6. Available capacity in the final trim area

Each day the scheduling model generated a daily build schedule for engines for the next 20 days. The build schedule was frozen for the first day of production and was adjusted daily for each of the remaining 19 days. Customer due date is the only hard constraint (constraint that cannot be violated) in this model. Soft constraints can be violated, but there is a penalty for each violation. The constraints and penalty points for each constraint are discussed in detail later in this section.

Figure 11 illustrates the first sheet of the scheduling model titled "import new orders." A new file is downloaded everyday by clicking on the icon titled "IMPORT FILE." Each file is updated daily in a folder saved on the server by the systems department. A macro was recorded with Microsoft Visual Basic in Excel to perform the import function from the server to the 20-day scheduling file. Each row represents an

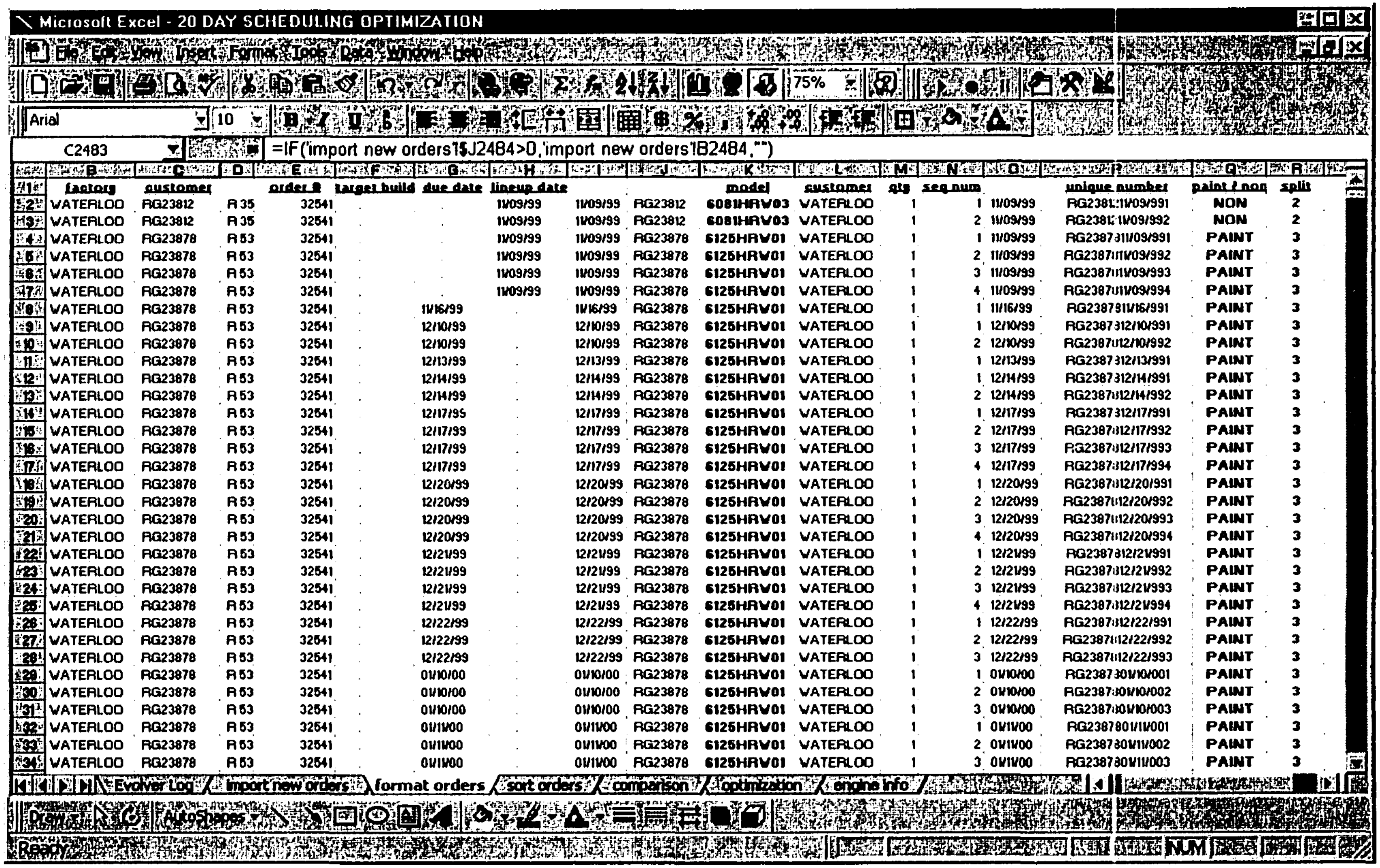


	CUSTOMER	PART #	ORDER #	SHIP DATE	CUSTOMER LINE-UP DATE	QUANTITY	SEQUENCE
2		19991105	19991221	.	.		
3	WATERLOO	RG23812	32541	.	11/09/99	1	1
4	WATERLOO	RG23812	32541	.	11/09/99	1	2
5	WATERLOO	RG23878	32541	.	11/09/99	1	1
6	WATERLOO	RG23878	32541	.	11/09/99	1	2
7	WATERLOO	RG23878	32541	.	11/09/99	1	3
8	WATERLOO	RG23878	32541	.	11/09/99	1	4
9	WATERLOO	RG23878	32541	11/16/99	.	1	1
10	WATERLOO	RG23878	32541	12/10/99	.	1	1
11	WATERLOO	RG23878	32541	12/10/99	.	1	2
12	WATERLOO	RG23878	32541	12/13/99	.	1	1
13	WATERLOO	RG23878	32541	12/14/99	.	1	1
14	WATERLOO	RG23878	32541	12/14/99	.	1	2
15	WATERLOO	RG23878	32541	12/17/99	.	1	1
16	WATERLOO	RG23878	32541	12/17/99	.	1	2
17	WATERLOO	RG23878	32541	12/17/99	.	1	3
18	WATERLOO	RG23878	32541	12/17/99	.	1	4
19	WATERLOO	RG23878	32541	12/20/99	.	1	1
20	WATERLOO	RG23878	32541	12/20/99	.	1	2
21	WATERLOO	RG23878	32541	12/20/99	.	1	3
22	WATERLOO	RG23878	32541	12/20/99	.	1	4

Figure 11. Import new orders.

order in this file. If a customer orders 10 engines for the same date, these 10 engines are represented in 10 continuous rows.

Data set received from legacy systems needs to be formatted before it can be utilized in a Windows-based application. Additional information is assembled using a function in Excel called Vlookup table. Numerous Excel formulas were used to clean the data and make it useable for the optimization. In the next sheet, "format orders," data are being filtered and cleaned. These formulas are visible in various figures in forthcoming sections. Figure 12 illustrates a snapshot of the "format orders" sheet, and Figure 13 illustrates the same sheet with the formulas in each cell visible. In the next sheet, "sort orders," shown in Figure 14, data are filtered again and sorted based on "target build date" criteria in ascending order. Customer orders that need to be built early on were moved to the top of the list. Figure 15 illustrates the same sheet with formulas visible in the cells.

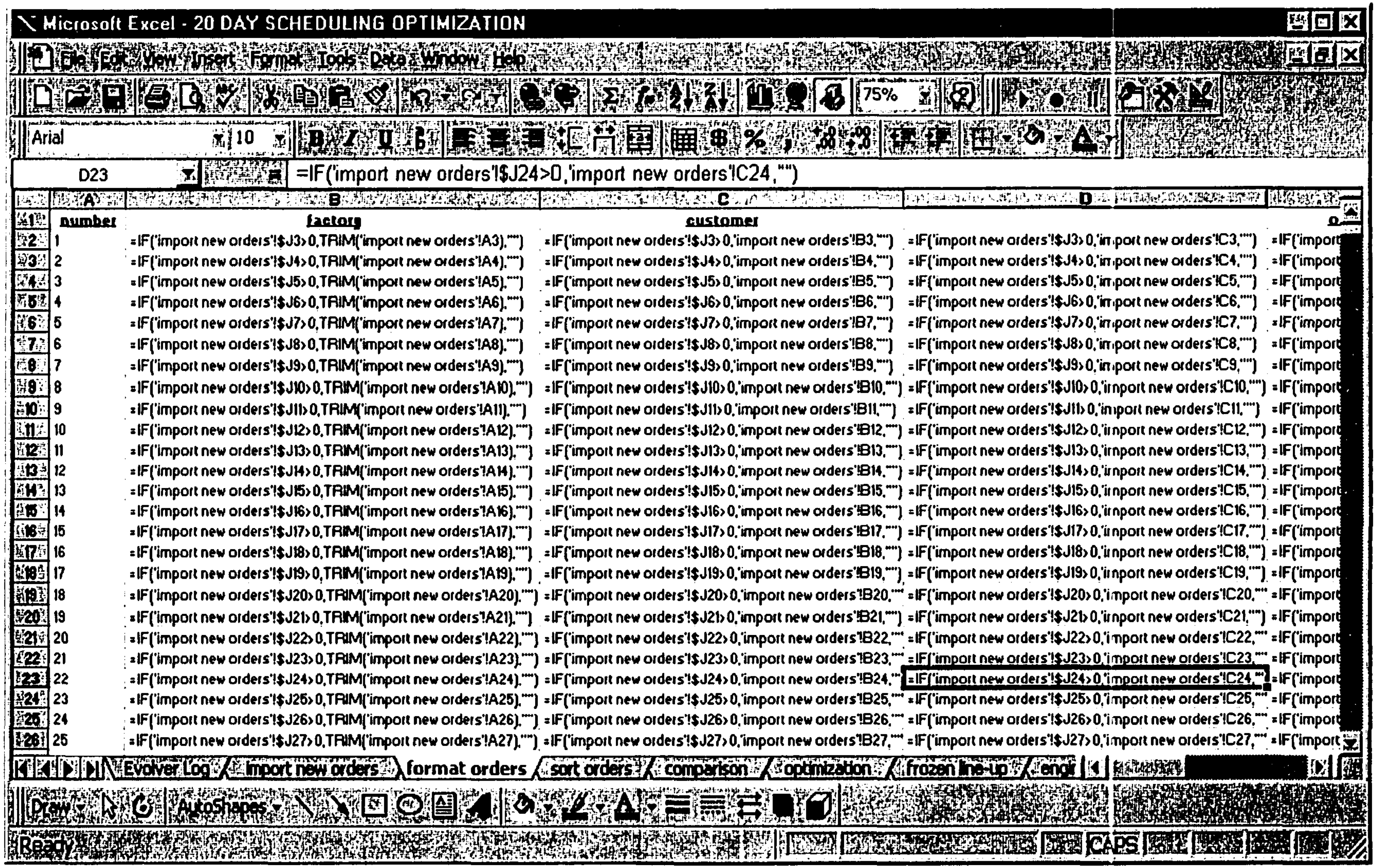


Microsoft Excel - 20 DAY SCHEDULING OPTIMIZATION

C2483 =IF('import new orders'!$J2484>0,'import new orders'!B2484,"")

	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	factory	customer		order #	target build	due date	lineup date			model	customer	qty	seq num		unique number	paint / non	split
2	WATERLOO	RG23812	R 35	32541			11/09/99	11/09/99	RG23812	6081HRV03	WATERLOO	1	1	11/09/99	RG2381211/09/991	NON	2
3	WATERLOO	RG23812	R 35	32541			11/09/99	11/09/99	RG23812	6081HRV03	WATERLOO	1	2	11/09/99	RG2381211/09/992	NON	2
4	WATERLOO	RG23878	R 53	32541			11/09/99	11/09/99	RG23878	6125HRV01	WATERLOO	1	1	11/09/99	RG2387811/09/991	PAINT	3
5	WATERLOO	RG23878	R 53	32541			11/09/99	11/09/99	RG23878	6125HRV01	WATERLOO	1	2	11/09/99	RG2387811/09/992	PAINT	3
6	WATERLOO	RG23878	R 53	32541			11/09/99	11/09/99	RG23878	6125HRV01	WATERLOO	1	3	11/09/99	RG2387811/09/993	PAINT	3
7	WATERLOO	RG23878	R 53	32541			11/09/99	11/09/99	RG23878	6125HRV01	WATERLOO	1	4	11/09/99	RG2387811/09/994	PAINT	3
8	WATERLOO	RG23878	R 53	32541		11/16/99		11/16/99	RG23878	6125HRV01	WATERLOO	1	1	11/16/99	RG2387811/16/991	PAINT	3
9	WATERLOO	RG23878	R 53	32541		12/10/99		12/10/99	RG23878	6125HRV01	WATERLOO	1	1	12/10/99	RG2387812/10/991	PAINT	3
10	WATERLOO	RG23878	R 53	32541		12/10/99		12/10/99	RG23878	6125HRV01	WATERLOO	1	2	12/10/99	RG2387812/10/992	PAINT	3
11	WATERLOO	RG23878	R 53	32541		12/13/99		12/13/99	RG23878	6125HRV01	WATERLOO	1	1	12/13/99	RG2387812/13/991	PAINT	3
12	WATERLOO	RG23878	R 53	32541		12/14/99		12/14/99	RG23878	6125HRV01	WATERLOO	1	1	12/14/99	RG2387812/14/991	PAINT	3
13	WATERLOO	RG23878	R 53	32541		12/14/99		12/14/99	RG23878	6125HRV01	WATERLOO	1	2	12/14/99	RG2387812/14/992	PAINT	3
14	WATERLOO	RG23878	R 53	32541		12/17/95		12/17/99	RG23878	6125HRV01	WATERLOO	1	1	12/17/99	RG2387812/17/991	PAINT	3
15	WATERLOO	RG23878	R 53	32541		12/17/99		12/17/99	RG23878	6125HRV01	WATERLOO	1	2	12/17/99	RG2387812/17/992	PAINT	3
16	WATERLOO	RG23878	R 53	32541		12/17/99		12/17/99	RG23878	6125HRV01	WATERLOO	1	3	12/17/99	RG2387812/17/993	PAINT	3
17	WATERLOO	RG23878	R 53	32541		12/17/99		12/17/99	RG23878	6125HRV01	WATERLOO	1	4	12/17/99	RG2387812/17/994	PAINT	3
18	WATERLOO	RG23878	R 53	32541		12/20/99		12/20/99	RG23878	6125HRV01	WATERLOO	1	1	12/20/99	RG2387812/20/991	PAINT	3
19	WATERLOO	RG23878	R 53	32541		12/20/99		12/20/99	RG23878	6125HRV01	WATERLOO	1	2	12/20/99	RG2387812/20/992	PAINT	3
20	WATERLOO	RG23878	R 53	32541		12/20/99		12/20/99	RG23878	6125HRV01	WATERLOO	1	3	12/20/99	RG2387812/20/993	PAINT	3
21	WATERLOO	RG23878	R 53	32541		12/20/99		12/20/99	RG23878	6125HRV01	WATERLOO	1	4	12/20/99	RG2387812/20/994	PAINT	3
22	WATERLOO	RG23878	R 53	32541		12/21/99		12/21/99	RG23878	6125HRV01	WATERLOO	1	1	12/21/99	RG2387812/21/991	PAINT	3
23	WATERLOO	RG23878	R 53	32541		12/21/99		12/21/99	RG23878	6125HRV01	WATERLOO	1	2	12/21/99	RG2387812/21/992	PAINT	3
24	WATERLOO	RG23878	R 53	32541		12/21/99		12/21/99	RG23878	6125HRV01	WATERLOO	1	3	12/21/99	RG2387812/21/993	PAINT	3
25	WATERLOO	RG23878	R 53	32541		12/21/99		12/21/99	RG23878	6125HRV01	WATERLOO	1	4	12/21/99	RG2387812/21/994	PAINT	3
26	WATERLOO	RG23878	R 53	32541		12/22/99		12/22/99	RG23878	6125HRV01	WATERLOO	1	1	12/22/99	RG2387812/22/991	PAINT	3
27	WATERLOO	RG23878	R 53	32541		12/22/99		12/22/99	RG23878	6125HRV01	WATERLOO	1	2	12/22/99	RG2387812/22/992	PAINT	3
28	WATERLOO	RG23878	R 53	32541		12/22/99		12/22/99	RG23878	6125HRV01	WATERLOO	1	3	12/22/99	RG2387812/22/993	PAINT	3
29	WATERLOO	RG23878	R 53	32541		01/10/00		01/10/00	RG23878	6125HRV01	WATERLOO	1	1	01/10/00	RG2387801/10/001	PAINT	3
30	WATERLOO	RG23878	R 53	32541		01/10/00		01/10/00	RG23878	6125HRV01	WATERLOO	1	2	01/10/00	RG2387801/10/002	PAINT	3
31	WATERLOO	RG23878	R 53	32541		01/10/00		01/10/00	RG23878	6125HRV01	WATERLOO	1	3	01/10/00	RG2387801/10/003	PAINT	3
32	WATERLOO	RG23878	R 53	32541		01/11/00		01/11/00	RG23878	6125HRV01	WATERLOO	1	1	01/11/00	RG2387801/11/001	PAINT	3
33	WATERLOO	RG23878	R 53	32541		01/11/00		01/11/00	RG23878	6125HRV01	WATERLOO	1	2	01/11/00	RG2387801/11/002	PAINT	3
34	WATERLOO	RG23878	R 53	32541		01/11/00		01/11/00	RG23878	6125HRV01	WATERLOO	1	3	01/11/00	RG2387801/11/003	PAINT	3

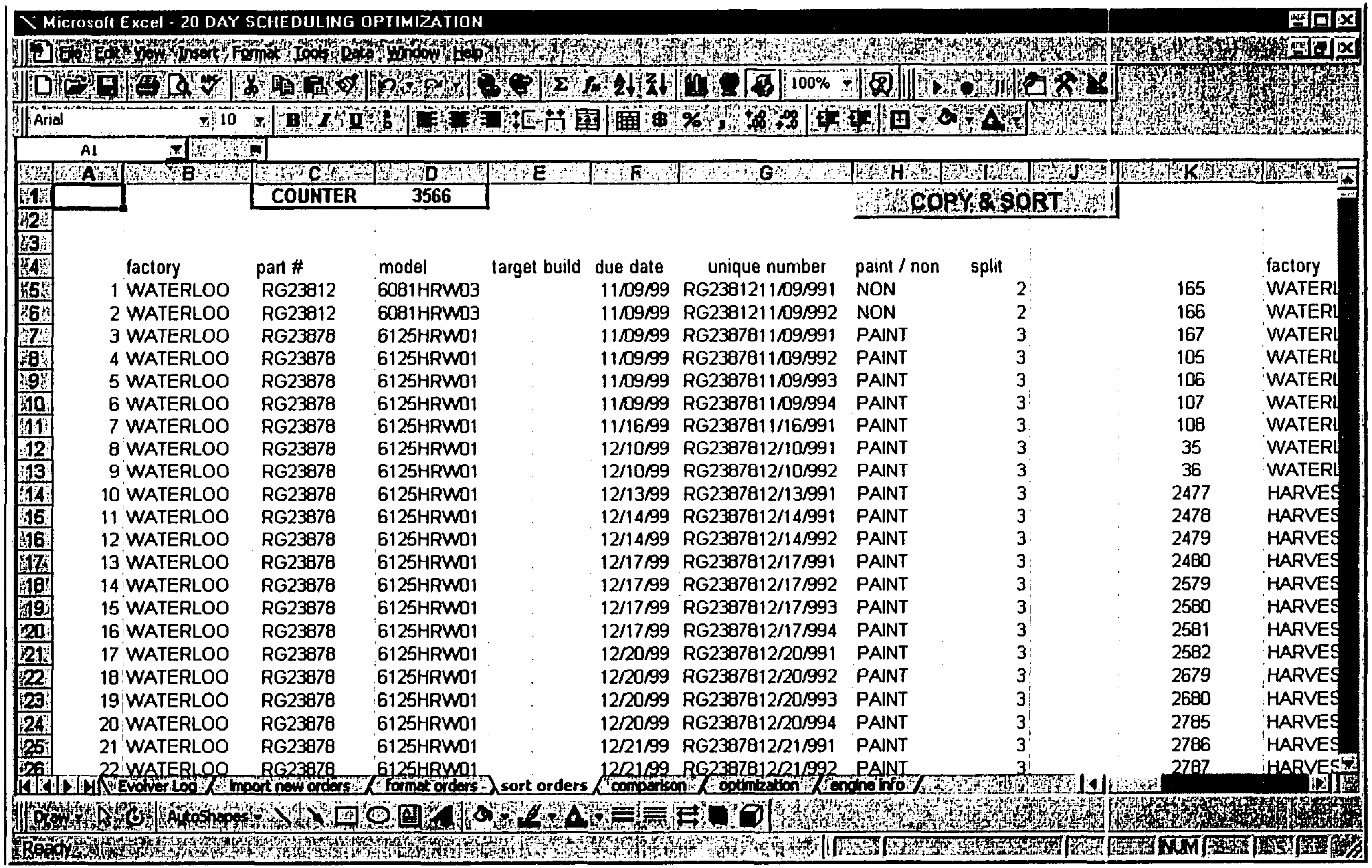
Figure 12. Format orders.



D23 =IF('import new orders'!$J24>0,'import new orders'!C24,"")

	A	B	C	D	
1	number	factory	customer		o
2	1	=IF('import new orders'!$J3>0,TRIM('import new orders'!A3),"")	=IF('import new orders'!$J3>0,'import new orders'!B3,"")	=IF('import new orders'!$J3>0,'import new orders'!C3,"")	=IF('import
3	2	=IF('import new orders'!$J4>0,TRIM('import new orders'!A4),"")	=IF('import new orders'!$J4>0,'import new orders'!B4,"")	=IF('import new orders'!$J4>0,'import new orders'!C4,"")	=IF('import
4	3	=IF('import new orders'!$J5>0,TRIM('import new orders'!A5),"")	=IF('import new orders'!$J5>0,'import new orders'!B5,"")	=IF('import new orders'!$J5>0,'import new orders'!C5,"")	=IF('import
5	4	=IF('import new orders'!$J6>0,TRIM('import new orders'!A6),"")	=IF('import new orders'!$J6>0,'import new orders'!B6,"")	=IF('import new orders'!$J6>0,'import new orders'!C6,"")	=IF('import
6	5	=IF('import new orders'!$J7>0,TRIM('import new orders'!A7),"")	=IF('import new orders'!$J7>0,'import new orders'!B7,"")	=IF('import new orders'!$J7>0,'import new orders'!C7,"")	=IF('import
7	6	=IF('import new orders'!$J8>0,TRIM('import new orders'!A8),"")	=IF('import new orders'!$J8>0,'import new orders'!B8,"")	=IF('import new orders'!$J8>0,'import new orders'!C8,"")	=IF('import
8	7	=IF('import new orders'!$J9>0,TRIM('import new orders'!A9),"")	=IF('import new orders'!$J9>0,'import new orders'!B9,"")	=IF('import new orders'!$J9>0,'import new orders'!C9,"")	=IF('import
9	8	=IF('import new orders'!$J10>0,TRIM('import new orders'!A10),"")	=IF('import new orders'!$J10>0,'import new orders'!B10,"")	=IF('import new orders'!$J10>0,'import new orders'!C10,"")	=IF('import
10	9	=IF('import new orders'!$J11>0,TRIM('import new orders'!A11),"")	=IF('import new orders'!$J11>0,'import new orders'!B11,"")	=IF('import new orders'!$J11>0,'import new orders'!C11,"")	=IF('import
11	10	=IF('import new orders'!$J12>0,TRIM('import new orders'!A12),"")	=IF('import new orders'!$J12>0,'import new orders'!B12,"")	=IF('import new orders'!$J12>0,'import new orders'!C12,"")	=IF('import
12	11	=IF('import new orders'!$J13>0,TRIM('import new orders'!A13),"")	=IF('import new orders'!$J13>0,'import new orders'!B13,"")	=IF('import new orders'!$J13>0,'import new orders'!C13,"")	=IF('import
13	12	=IF('import new orders'!$J14>0,TRIM('import new orders'!A14),"")	=IF('import new orders'!$J14>0,'import new orders'!B14,"")	=IF('import new orders'!$J14>0,'import new orders'!C14,"")	=IF('import
14	13	=IF('import new orders'!$J15>0,TRIM('import new orders'!A15),"")	=IF('import new orders'!$J15>0,'import new orders'!B15,"")	=IF('import new orders'!$J15>0,'import new orders'!C15,"")	=IF('import
15	14	=IF('import new orders'!$J16>0,TRIM('import new orders'!A16),"")	=IF('import new orders'!$J16>0,'import new orders'!B16,"")	=IF('import new orders'!$J16>0,'import new orders'!C16,"")	=IF('import
16	15	=IF('import new orders'!$J17>0,TRIM('import new orders'!A17),"")	=IF('import new orders'!$J17>0,'import new orders'!B17,"")	=IF('import new orders'!$J17>0,'import new orders'!C17,"")	=IF('import
17	16	=IF('import new orders'!$J18>0,TRIM('import new orders'!A18),"")	=IF('import new orders'!$J18>0,'import new orders'!B18,"")	=IF('import new orders'!$J18>0,'import new orders'!C18,"")	=IF('import
18	17	=IF('import new orders'!$J19>0,TRIM('import new orders'!A19),"")	=IF('import new orders'!$J19>0,'import new orders'!B19,"")	=IF('import new orders'!$J19>0,'import new orders'!C19,"")	=IF('import
19	18	=IF('import new orders'!$J20>0,TRIM('import new orders'!A20),"")	=IF('import new orders'!$J20>0,'import new orders'!B20,""	=IF('import new orders'!$J20>0,'import new orders'!C20,""	=IF('import
20	19	=IF('import new orders'!$J21>0,TRIM('import new orders'!A21),"")	=IF('import new orders'!$J21>0,'import new orders'!B21,"")	=IF('import new orders'!$J21>0,'import new orders'!C21,"")	=IF('import
21	20	=IF('import new orders'!$J22>0,TRIM('import new orders'!A22),"")	=IF('import new orders'!$J22>0,'import new orders'!B22,""	=IF('import new orders'!$J22>0,'import new orders'!C22,""	=IF('import
22	21	=IF('import new orders'!$J23>0,TRIM('import new orders'!A23),"")	=IF('import new orders'!$J23>0,'import new orders'!B23,""	=IF('import new orders'!$J23>0,'import new orders'!C23,""	=IF('import
23	22	=IF('import new orders'!$J24>0,TRIM('import new orders'!A24),"")	=IF('import new orders'!$J24>0,'import new orders'!B24,""	=IF('import new orders'!$J24>0,'import new orders'!C24,""	=IF('import
24	23	=IF('import new orders'!$J25>0,TRIM('import new orders'!A25),"")	=IF('import new orders'!$J25>0,'import new orders'!B25,""	=IF('import new orders'!$J25>0,'import new orders'!C25,""	=IF('import
25	24	=IF('import new orders'!$J26>0,TRIM('import new orders'!A26),"")	=IF('import new orders'!$J26>0,'import new orders'!B26,""	=IF('import new orders'!$J26>0,'import new orders'!C26,""	=IF('import
26	25	=IF('import new orders'!$J27>0,TRIM('import new orders'!A27),"")	=IF('import new orders'!$J27>0,'import new orders'!B27,""	=IF('import new orders'!$J27>0,'import new orders'!C27,""	=IF('import

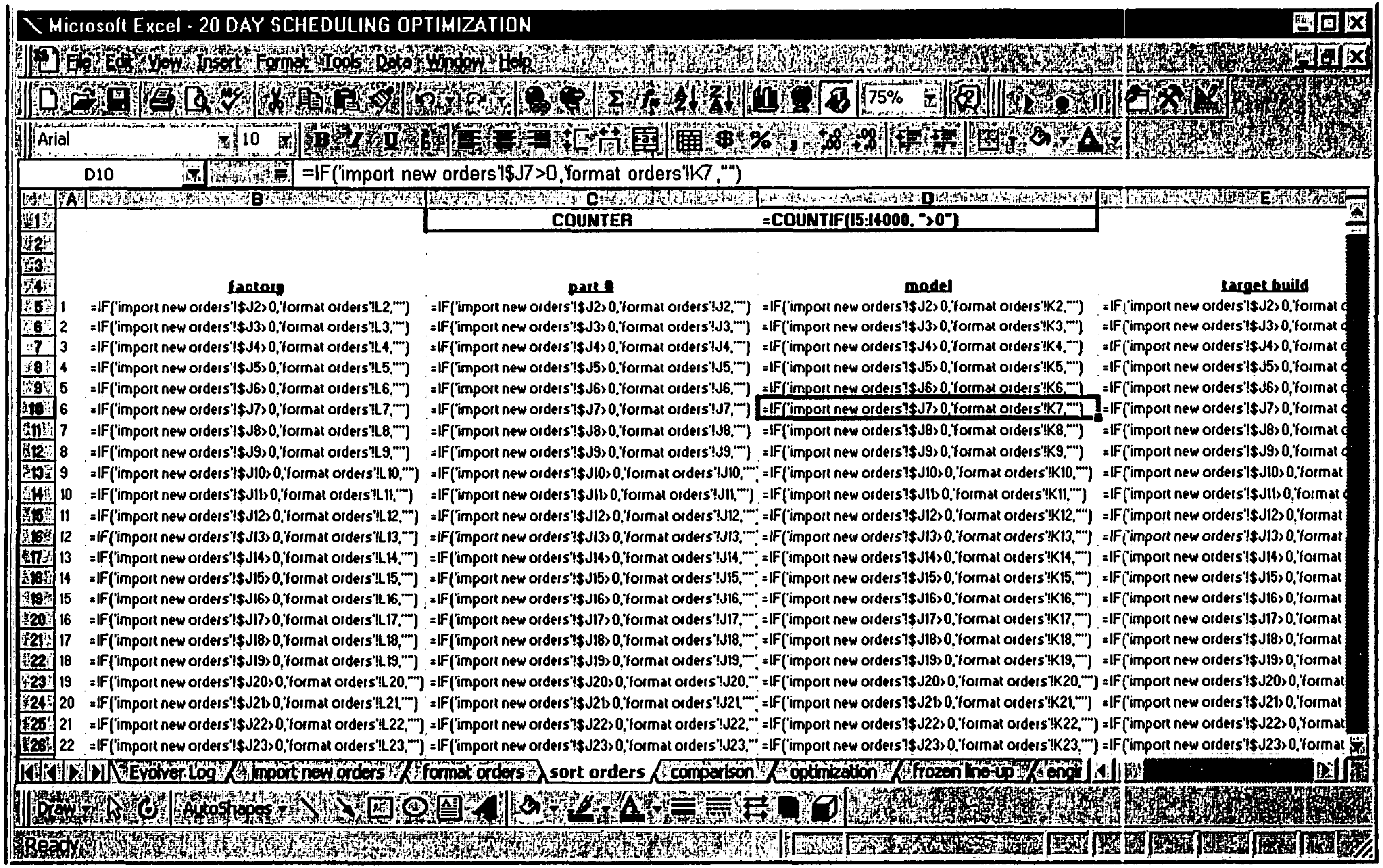
Figure 13. Format orders with formulas visible in the cells.



COUNTER 3566

	factory	part #	model	target build	due date	unique number	paint / non	split		factory
1	WATERLOO	RG23812	6081HRW03		11/09/99	RG2381211/09/991	NON	2	165	WATERL
2	WATERLOO	RG23812	6081HRW03		11/09/99	RG2381211/09/992	NON	2	166	WATERL
3	WATERLOO	RG23878	6125HRW01		11/09/99	RG2387811/09/991	PAINT	3	167	WATERL
4	WATERLOO	RG23878	6125HRW01		11/09/99	RG2387811/09/992	PAINT	3	105	WATERL
5	WATERLOO	RG23878	6125HRW01		11/09/99	RG2387811/09/993	PAINT	3	106	WATERL
6	WATERLOO	RG23878	6125HRW01		11/09/99	RG2387811/09/994	PAINT	3	107	WATERL
7	WATERLOO	RG23878	6125HRW01		11/16/99	RG2387811/16/991	PAINT	3	108	WATERL
8	WATERLOO	RG23878	6125HRW01		12/10/99	RG2387812/10/991	PAINT	3	35	WATERL
9	WATERLOO	RG23878	6125HRW01		12/10/99	RG2387812/10/992	PAINT	3	36	WATERL
10	WATERLOO	RG23878	6125HRW01		12/13/99	RG2387812/13/991	PAINT	3	2477	HARVES
11	WATERLOO	RG23878	6125HRW01		12/14/99	RG2387812/14/991	PAINT	3	2478	HARVES
12	WATERLOO	RG23878	6125HRW01		12/14/99	RG2387812/14/992	PAINT	3	2479	HARVES
13	WATERLOO	RG23878	6125HRW01		12/17/99	RG2387812/17/991	PAINT	3	2480	HARVES
14	WATERLOO	RG23878	6125HRW01		12/17/99	RG2387812/17/992	PAINT	3	2579	HARVES
15	WATERLOO	RG23878	6125HRW01		12/17/99	RG2387812/17/993	PAINT	3	2580	HARVES
16	WATERLOO	RG23878	6125HRW01		12/17/99	RG2387812/17/994	PAINT	3	2581	HARVES
17	WATERLOO	RG23878	6125HRW01		12/20/99	RG2387812/20/991	PAINT	3	2582	HARVES
18	WATERLOO	RG23878	6125HRW01		12/20/99	RG2387812/20/992	PAINT	3	2679	HARVES
19	WATERLOO	RG23878	6125HRW01		12/20/99	RG2387812/20/993	PAINT	3	2680	HARVES
20	WATERLOO	RG23878	6125HRW01		12/20/99	RG2387812/20/994	PAINT	3	2785	HARVES
21	WATERLOO	RG23878	6125HRW01		12/21/99	RG2387812/21/991	PAINT	3	2786	HARVES
22	WATERLOO	RG23878	6125HRW01		12/21/99	RG2387812/21/992	PAINT	3	2787	HARVES

Figure 14. Sort orders.



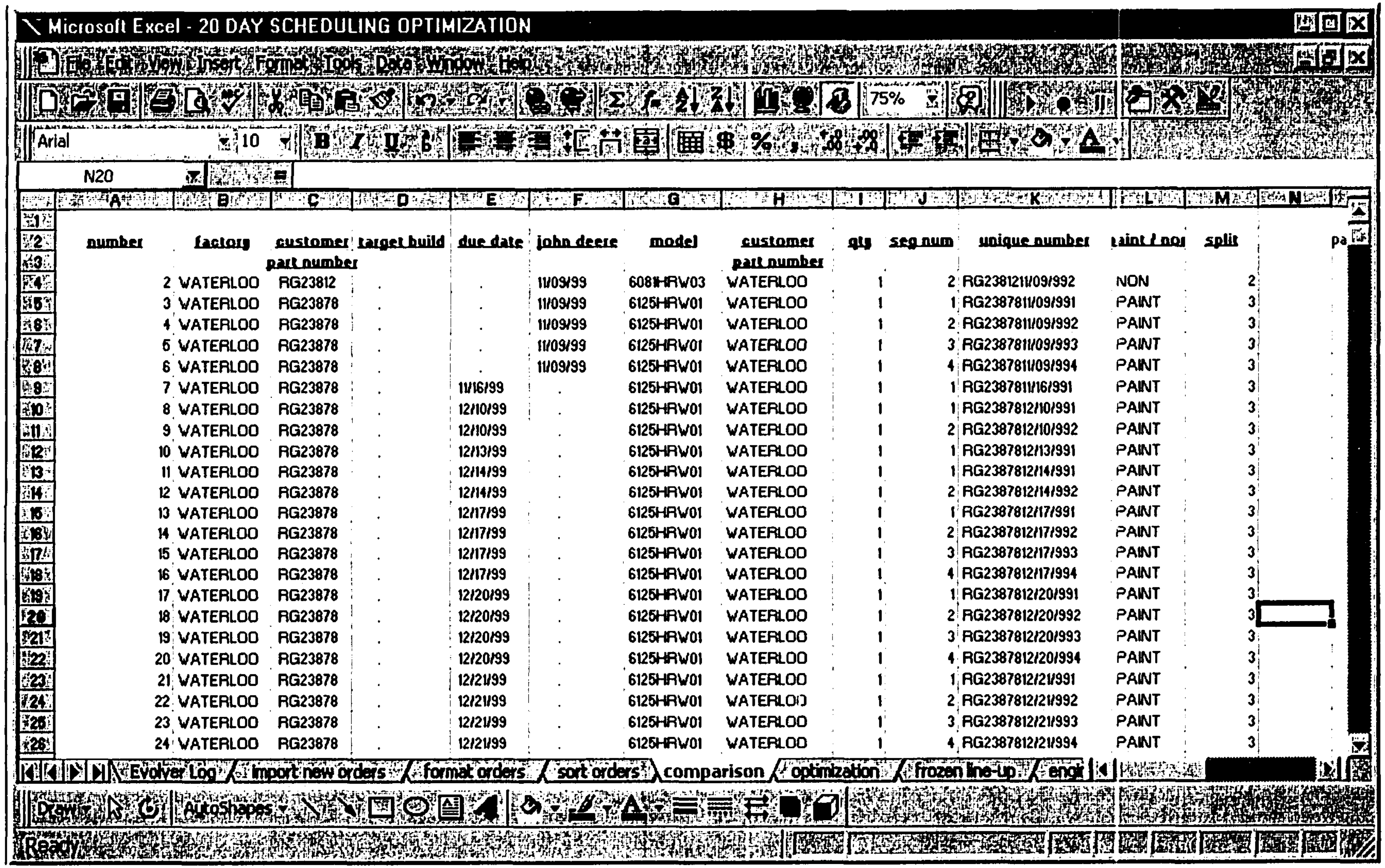
D10 =IF('import new orders'!$J7>0,'format orders'!K7,"")

	A	B	C	D	E
1			COUNTER	=COUNTIF(I5:I4000, ">0")	
2					
3					
4		factory	part #	model	target build
5	1	=IF('import new orders'!$J2>0,'format orders'!L2,"")	=IF('import new orders'!$J2>0,'format orders'!J2,"")	=IF('import new orders'!$J2>0,'format orders'!K2,"")	=IF('import new orders'!$J2>0,'format
6	2	=IF('import new orders'!$J3>0,'format orders'!L3,"")	=IF('import new orders'!$J3>0,'format orders'!J3,"")	=IF('import new orders'!$J3>0,'format orders'!K3,"")	=IF('import new orders'!$J3>0,'format
7	3	=IF('import new orders'!$J4>0,'format orders'!L4,"")	=IF('import new orders'!$J4>0,'format orders'!J4,"")	=IF('import new orders'!$J4>0,'format orders'!K4,"")	=IF('import new orders'!$J4>0,'format
8	4	=IF('import new orders'!$J5>0,'format orders'!L5,"")	=IF('import new orders'!$J5>0,'format orders'!J5,"")	=IF('import new orders'!$J5>0,'format orders'!K5,"")	=IF('import new orders'!$J5>0,'format
9	5	=IF('import new orders'!$J6>0,'format orders'!L6,"")	=IF('import new orders'!$J6>0,'format orders'!J6,"")	=IF('import new orders'!$J6>0,'format orders'!K6,"")	=IF('import new orders'!$J6>0,'format
10	6	=IF('import new orders'!$J7>0,'format orders'!L7,"")	=IF('import new orders'!$J7>0,'format orders'!J7,"")	=IF('import new orders'!$J7>0,'format orders'!K7,"")	=IF('import new orders'!$J7>0,'format
11	7	=IF('import new orders'!$J8>0,'format orders'!L8,"")	=IF('import new orders'!$J8>0,'format orders'!J8,"")	=IF('import new orders'!$J8>0,'format orders'!K8,"")	=IF('import new orders'!$J8>0,'format
12	8	=IF('import new orders'!$J9>0,'format orders'!L9,"")	=IF('import new orders'!$J9>0,'format orders'!J9,"")	=IF('import new orders'!$J9>0,'format orders'!K9,"")	=IF('import new orders'!$J9>0,'format
13	9	=IF('import new orders'!$J10>0,'format orders'!L10,"")	=IF('import new orders'!$J10>0,'format orders'!J10,""	=IF('import new orders'!$J10>0,'format orders'!K10,"")	=IF('import new orders'!$J10>0,'format
14	10	=IF('import new orders'!$J11>0,'format orders'!L11,"")	=IF('import new orders'!$J11>0,'format orders'!J11,"")	=IF('import new orders'!$J11>0,'format orders'!K11,"")	=IF('import new orders'!$J11>0,'format
15	11	=IF('import new orders'!$J12>0,'format orders'!L12,"")	=IF('import new orders'!$J12>0,'format orders'!J12,""	=IF('import new orders'!$J12>0,'format orders'!K12,"")	=IF('import new orders'!$J12>0,'format
16	12	=IF('import new orders'!$J13>0,'format orders'!L13,"")	=IF('import new orders'!$J13>0,'format orders'!J13,""	=IF('import new orders'!$J13>0,'format orders'!K13,"")	=IF('import new orders'!$J13>0,'format
17	13	=IF('import new orders'!$J14>0,'format orders'!L14,"")	=IF('import new orders'!$J14>0,'format orders'!J14,""	=IF('import new orders'!$J14>0,'format orders'!K14,"")	=IF('import new orders'!$J14>0,'format
18	14	=IF('import new orders'!$J15>0,'format orders'!L15,"")	=IF('import new orders'!$J15>0,'format orders'!J15,""	=IF('import new orders'!$J15>0,'format orders'!K15,"")	=IF('import new orders'!$J15>0,'format
19	15	=IF('import new orders'!$J16>0,'format orders'!L16,"")	=IF('import new orders'!$J16>0,'format orders'!J16,""	=IF('import new orders'!$J16>0,'format orders'!K16,"")	=IF('import new orders'!$J16>0,'format
20	16	=IF('import new orders'!$J17>0,'format orders'!L17,"")	=IF('import new orders'!$J17>0,'format orders'!J17,""	=IF('import new orders'!$J17>0,'format orders'!K17,"")	=IF('import new orders'!$J17>0,'format
21	17	=IF('import new orders'!$J18>0,'format orders'!L18,"")	=IF('import new orders'!$J18>0,'format orders'!J18,""	=IF('import new orders'!$J18>0,'format orders'!K18,"")	=IF('import new orders'!$J18>0,'format
22	18	=IF('import new orders'!$J19>0,'format orders'!L19,"")	=IF('import new orders'!$J19>0,'format orders'!J19,""	=IF('import new orders'!$J19>0,'format orders'!K19,"")	=IF('import new orders'!$J19>0,'format
23	19	=IF('import new orders'!$J20>0,'format orders'!L20,"")	=IF('import new orders'!$J20>0,'format orders'!J20,""	=IF('import new orders'!$J20>0,'format orders'!K20,"")	=IF('import new orders'!$J20>0,'format
24	20	=IF('import new orders'!$J21>0,'format orders'!L21,"")	=IF('import new orders'!$J21>0,'format orders'!J21,""	=IF('import new orders'!$J21>0,'format orders'!K21,"")	=IF('import new orders'!$J21>0,'format
25	21	=IF('import new orders'!$J22>0,'format orders'!L22,"")	=IF('import new orders'!$J22>0,'format orders'!J22,""	=IF('import new orders'!$J22>0,'format orders'!K22,"")	=IF('import new orders'!$J22>0,'format
26	22	=IF('import new orders'!$J23>0,'format orders'!L23,"")	=IF('import new orders'!$J23>0,'format orders'!J23,""	=IF('import new orders'!$J23>0,'format orders'!K23,"")	=IF('import new orders'!$J23>0,'format

Figure 15. Sort orders with formulas visible in the cells.

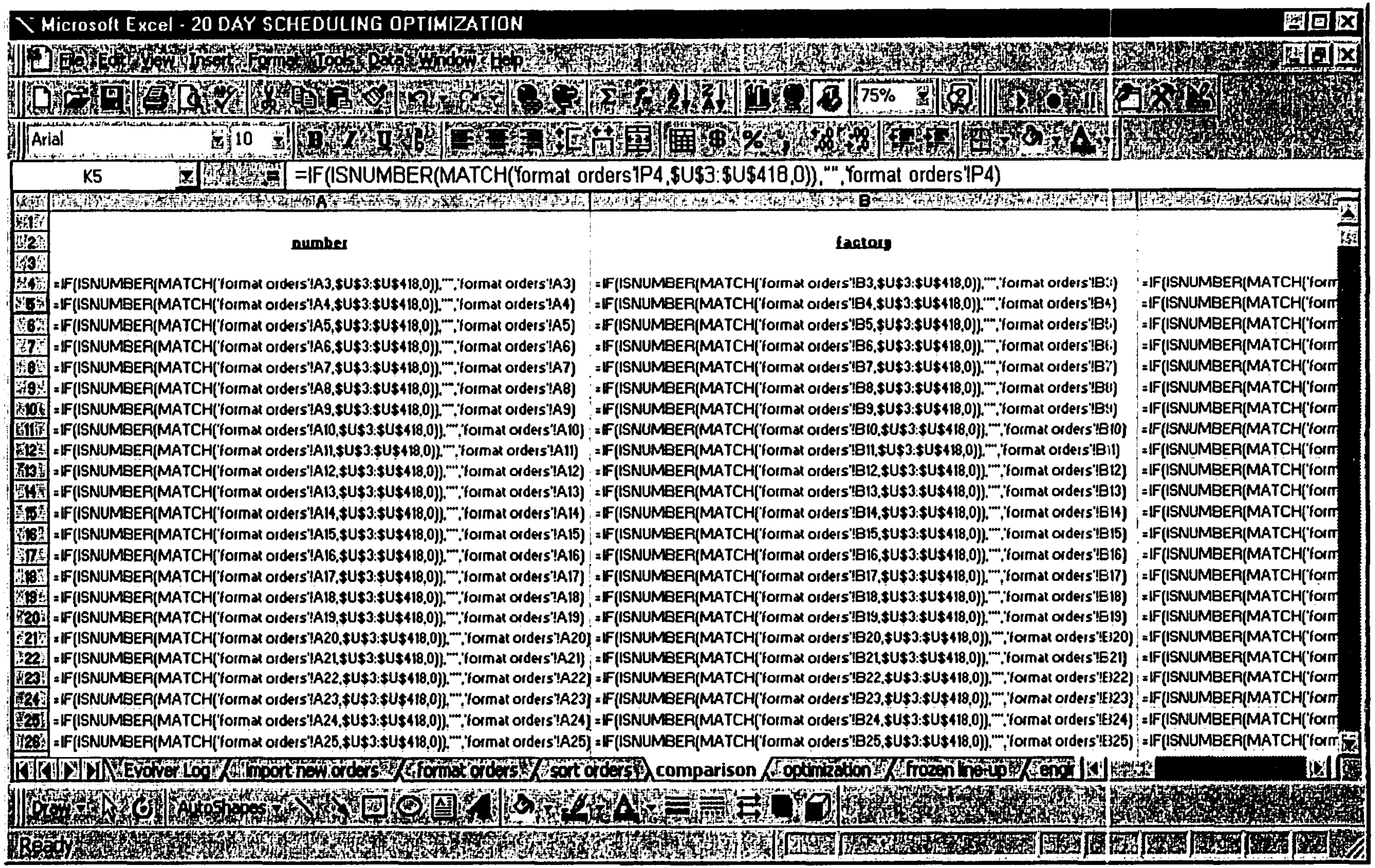
Before the orders are linked to the "optimization" sheet, they are compared with the previous day's frozen line-up. This step was necessary to avoid orders being duplicated. If an order is already frozen the previous day, that order will not be linked to the "optimization" sheet and thus will not be used for optimization. Figure 16 presents a snapshot of a "comparison" sheet, and Figure 17 depicts the same sheet with formulas visible in each cell.

Figures 18–23 illustrate various sections of the optimization sheet, the next step in the scheduling model. Figures 18 and 19 display the section in which available capacity in standard minutes is calculated for the j-hook capacity (final assembly line), test (engine test cells), custom trim (painted engines are trimmed before paint operation), final trim (painted engines are trimmed again after paint), and paint operations. Figures 20 and 21 illustrate the required capacity in standard minutes for the same processes. A calculation for the difference in available and required capacity for each process is also performed here. Figures 22 and 23 present the optimization sheet displaying scheduled orders with regard to customer ship dates. If an order is scheduled late, the date field is highlighted in red, making it readily visible for the master scheduler to adjust the schedule.



number	factory	customer part number	target build	due date	john deere	model	customer part number	qty	seg num	unique number	aint / no	split	pa
2	WATERLOO	RG23812	.	.	11/09/99	608HRW03	WATERLOO	1	2	RG2381211/09/992	NON	2	
3	WATERLOO	RG23878	.	.	11/09/99	6125HRW01	WATERLOO	1	1	RG2387811/09/991	PAINT	3	
4	WATERLOO	RG23878	.	.	11/09/99	6125HRW01	WATERLOO	1	2	RG2387811/09/992	PAINT	3	
5	WATERLOO	RG23878	.	.	11/09/99	6125HRW01	WATERLOO	1	3	RG2387811/09/993	PAINT	3	
6	WATERLOO	RG23878	.	.	11/09/99	6125HRW01	WATERLOO	1	4	RG2387811/09/994	PAINT	3	
7	WATERLOO	RG23878	.	11/16/99	.	6125HRW01	WATERLOO	1	1	RG2387811/16/991	PAINT	3	
8	WATERLOO	RG23878	.	12/10/99	.	6125HRW01	WATERLOO	1	1	RG2387812/10/991	PAINT	3	
9	WATERLOO	RG23878	.	12/10/99	.	6125HRW01	WATERLOO	1	2	RG2387812/10/992	PAINT	3	
10	WATERLOO	RG23878	.	12/13/99	.	6125HRW01	WATERLOO	1	1	RG2387812/13/991	PAINT	3	
11	WATERLOO	RG23878	.	12/14/99	.	6125HRW01	WATERLOO	1	1	RG2387812/14/991	PAINT	3	
12	WATERLOO	RG23878	.	12/14/99	.	6125HRW01	WATERLOO	1	2	RG2387812/14/992	PAINT	3	
13	WATERLOO	RG23878	.	12/17/99	.	6125HRW01	WATERLOO	1	1	RG2387812/17/991	PAINT	3	
14	WATERLOO	RG23878	.	12/17/99	.	6125HRW01	WATERLOO	1	2	RG2387812/17/992	PAINT	3	
15	WATERLOO	RG23878	.	12/17/99	.	6125HRW01	WATERLOO	1	3	RG2387812/17/993	PAINT	3	
16	WATERLOO	RG23878	.	12/17/99	.	6125HRW01	WATERLOO	1	4	RG2387812/17/994	PAINT	3	
17	WATERLOO	RG23878	.	12/20/99	.	6125HRW01	WATERLOO	1	1	RG2387812/20/991	PAINT	3	
18	WATERLOO	RG23878	.	12/20/99	.	6125HRW01	WATERLOO	1	2	RG2387812/20/992	PAINT	3	
19	WATERLOO	RG23878	.	12/20/99	.	6125HRW01	WATERLOO	1	3	RG2387812/20/993	PAINT	3	
20	WATERLOO	RG23878	.	12/20/99	.	6125HRW01	WATERLOO	1	4	RG2387812/20/994	PAINT	3	
21	WATERLOO	RG23878	.	12/21/99	.	6125HRW01	WATERLOO	1	1	RG2387812/21/991	PAINT	3	
22	WATERLOO	RG23878	.	12/21/99	.	6125HRW01	WATERLOO	1	2	RG2387812/21/992	PAINT	3	
23	WATERLOO	RG23878	.	12/21/99	.	6125HRW01	WATERLOO	1	3	RG2387812/21/993	PAINT	3	
24	WATERLOO	RG23878	.	12/21/99	.	6125HRW01	WATERLOO	1	4	RG2387812/21/994	PAINT	3	

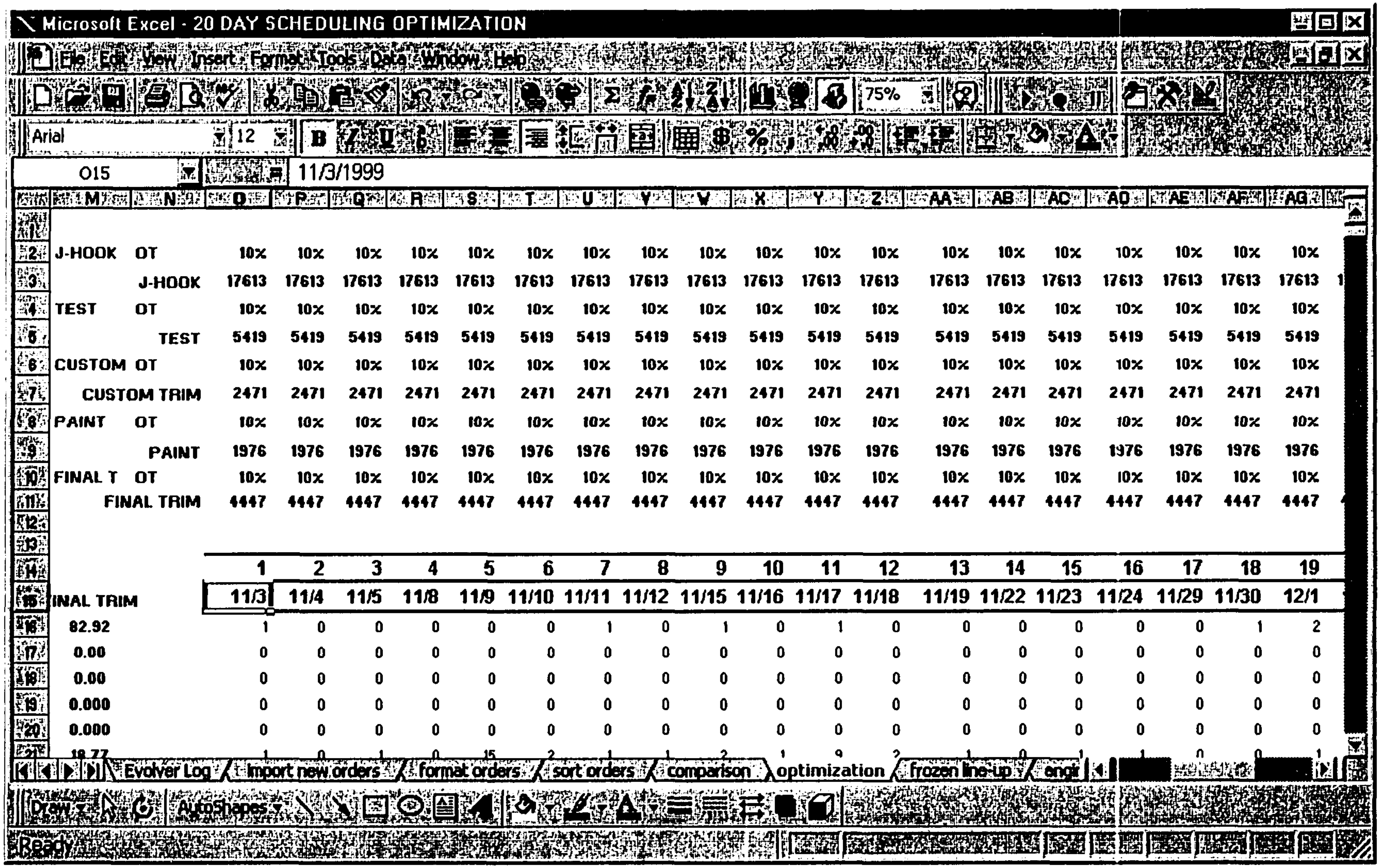
Figure 16. Comparison.



K5 =IF(ISNUMBER(MATCH('format orders'!P4,$U$3:$U$418,0)),"",'format orders'!P4)

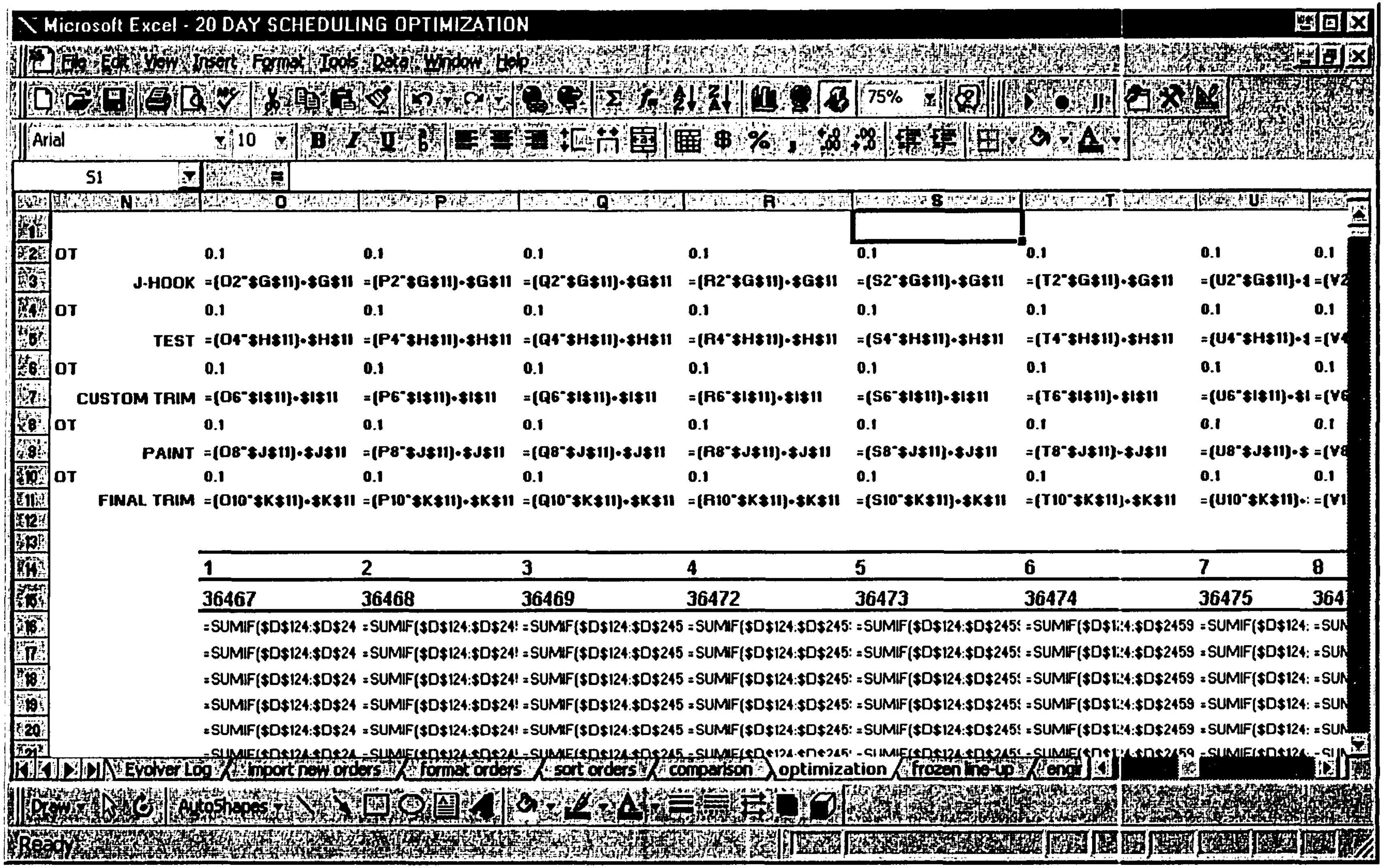
	A	B	C
1			
2	**number**	**factors**	
3			
4	=IF(ISNUMBER(MATCH('format orders'!A3,$U$3:$U$418,0)),"",'format orders'!A3)	=IF(ISNUMBER(MATCH('format orders'!B3,$U$3:$U$418,0)),"",'format orders'!B3)	=IF(ISNUMBER(MATCH('form
5	=IF(ISNUMBER(MATCH('format orders'!A4,$U$3:$U$418,0)),"",'format orders'!A4)	=IF(ISNUMBER(MATCH('format orders'!B4,$U$3:$U$418,0)),"",'format orders'!B4)	=IF(ISNUMBER(MATCH('form
6	=IF(ISNUMBER(MATCH('format orders'!A5,$U$3:$U$418,0)),"",'format orders'!A5)	=IF(ISNUMBER(MATCH('format orders'!B5,$U$3:$U$418,0)),"",'format orders'!B5)	=IF(ISNUMBER(MATCH('form
7	=IF(ISNUMBER(MATCH('format orders'!A6,$U$3:$U$418,0)),"",'format orders'!A6)	=IF(ISNUMBER(MATCH('format orders'!B6,$U$3:$U$418,0)),"",'format orders'!B6)	=IF(ISNUMBER(MATCH('form
8	=IF(ISNUMBER(MATCH('format orders'!A7,$U$3:$U$418,0)),"",'format orders'!A7)	=IF(ISNUMBER(MATCH('format orders'!B7,$U$3:$U$418,0)),"",'format orders'!B7)	=IF(ISNUMBER(MATCH('form
9	=IF(ISNUMBER(MATCH('format orders'!A8,$U$3:$U$418,0)),"",'format orders'!A8)	=IF(ISNUMBER(MATCH('format orders'!B8,$U$3:$U$418,0)),"",'format orders'!B8)	=IF(ISNUMBER(MATCH('form
10	=IF(ISNUMBER(MATCH('format orders'!A9,$U$3:$U$418,0)),"",'format orders'!A9)	=IF(ISNUMBER(MATCH('format orders'!B9,$U$3:$U$418,0)),"",'format orders'!B9)	=IF(ISNUMBER(MATCH('form
11	=IF(ISNUMBER(MATCH('format orders'!A10,$U$3:$U$418,0)),"",'format orders'!A10)	=IF(ISNUMBER(MATCH('format orders'!B10,$U$3:$U$418,0)),"",'format orders'!B10)	=IF(ISNUMBER(MATCH('form
12	=IF(ISNUMBER(MATCH('format orders'!A11,$U$3:$U$418,0)),"",'format orders'!A11)	=IF(ISNUMBER(MATCH('format orders'!B11,$U$3:$U$418,0)),"",'format orders'!B11)	=IF(ISNUMBER(MATCH('form
13	=IF(ISNUMBER(MATCH('format orders'!A12,$U$3:$U$418,0)),"",'format orders'!A12)	=IF(ISNUMBER(MATCH('format orders'!B12,$U$3:$U$418,0)),"",'format orders'!B12)	=IF(ISNUMBER(MATCH('form
14	=IF(ISNUMBER(MATCH('format orders'!A13,$U$3:$U$418,0)),"",'format orders'!A13)	=IF(ISNUMBER(MATCH('format orders'!B13,$U$3:$U$418,0)),"",'format orders'!B13)	=IF(ISNUMBER(MATCH('form
15	=IF(ISNUMBER(MATCH('format orders'!A14,$U$3:$U$418,0)),"",'format orders'!A14)	=IF(ISNUMBER(MATCH('format orders'!B14,$U$3:$U$418,0)),"",'format orders'!B14)	=IF(ISNUMBER(MATCH('form
16	=IF(ISNUMBER(MATCH('format orders'!A15,$U$3:$U$418,0)),"",'format orders'!A15)	=IF(ISNUMBER(MATCH('format orders'!B15,$U$3:$U$418,0)),"",'format orders'!B15)	=IF(ISNUMBER(MATCH('form
17	=IF(ISNUMBER(MATCH('format orders'!A16,$U$3:$U$418,0)),"",'format orders'!A16)	=IF(ISNUMBER(MATCH('format orders'!B16,$U$3:$U$418,0)),"",'format orders'!B16)	=IF(ISNUMBER(MATCH('form
18	=IF(ISNUMBER(MATCH('format orders'!A17,$U$3:$U$418,0)),"",'format orders'!A17)	=IF(ISNUMBER(MATCH('format orders'!B17,$U$3:$U$418,0)),"",'format orders'!B17)	=IF(ISNUMBER(MATCH('form
19	=IF(ISNUMBER(MATCH('format orders'!A18,$U$3:$U$418,0)),"",'format orders'!A18)	=IF(ISNUMBER(MATCH('format orders'!B18,$U$3:$U$418,0)),"",'format orders'!B18)	=IF(ISNUMBER(MATCH('form
20	=IF(ISNUMBER(MATCH('format orders'!A19,$U$3:$U$418,0)),"",'format orders'!A19)	=IF(ISNUMBER(MATCH('format orders'!B19,$U$3:$U$418,0)),"",'format orders'!B19)	=IF(ISNUMBER(MATCH('form
21	=IF(ISNUMBER(MATCH('format orders'!A20,$U$3:$U$418,0)),"",'format orders'!A20)	=IF(ISNUMBER(MATCH('format orders'!B20,$U$3:$U$418,0)),"",'format orders'!B20)	=IF(ISNUMBER(MATCH('form
22	=IF(ISNUMBER(MATCH('format orders'!A21,$U$3:$U$418,0)),"",'format orders'!A21)	=IF(ISNUMBER(MATCH('format orders'!B21,$U$3:$U$418,0)),"",'format orders'!B21)	=IF(ISNUMBER(MATCH('form
23	=IF(ISNUMBER(MATCH('format orders'!A22,$U$3:$U$418,0)),"",'format orders'!A22)	=IF(ISNUMBER(MATCH('format orders'!B22,$U$3:$U$418,0)),"",'format orders'!B22)	=IF(ISNUMBER(MATCH('form
24	=IF(ISNUMBER(MATCH('format orders'!A23,$U$3:$U$418,0)),"",'format orders'!A23)	=IF(ISNUMBER(MATCH('format orders'!B23,$U$3:$U$418,0)),"",'format orders'!B23)	=IF(ISNUMBER(MATCH('form
25	=IF(ISNUMBER(MATCH('format orders'!A24,$U$3:$U$418,0)),"",'format orders'!A24)	=IF(ISNUMBER(MATCH('format orders'!B24,$U$3:$U$418,0)),"",'format orders'!B24)	=IF(ISNUMBER(MATCH('form
26	=IF(ISNUMBER(MATCH('format orders'!A25,$U$3:$U$418,0)),"",'format orders'!A25)	=IF(ISNUMBER(MATCH('format orders'!B25,$U$3:$U$418,0)),"",'format orders'!B25)	=IF(ISNUMBER(MATCH('form

Figure 17. Comparison with formulas visible in the cells.



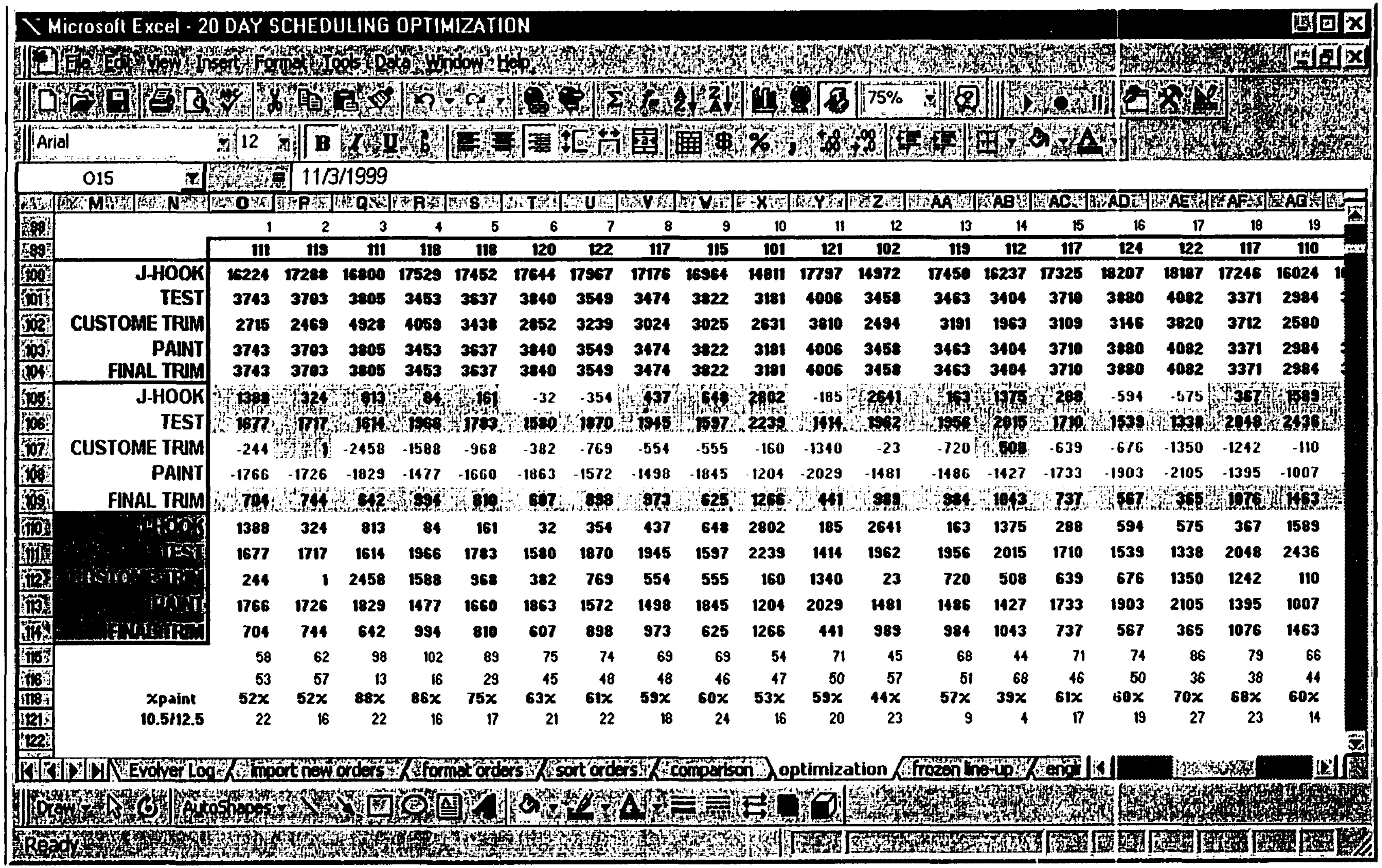
	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG
1																					
2	J-HOOK	OT	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%
3		J-HOOK	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613	17613
4	TEST	OT	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%
5		TEST	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419	5419
6	CUSTOM	OT	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%
7		CUSTOM TRIM	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471	2471
8	PAINT	OT	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%
9		PAINT	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976	1976
10	FINAL T	OT	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%	10%
11		FINAL TRIM	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447	4447
12																					
13																					
14			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
15	INAL TRIM		11/3	11/4	11/5	11/8	11/9	11/10	11/11	11/12	11/15	11/16	11/17	11/18	11/19	11/22	11/23	11/24	11/29	11/30	12/1
16	82.92		1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	1	2
17	0.00		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0.00		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0.000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0.000		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 18. Optimization tab, available capacity section.



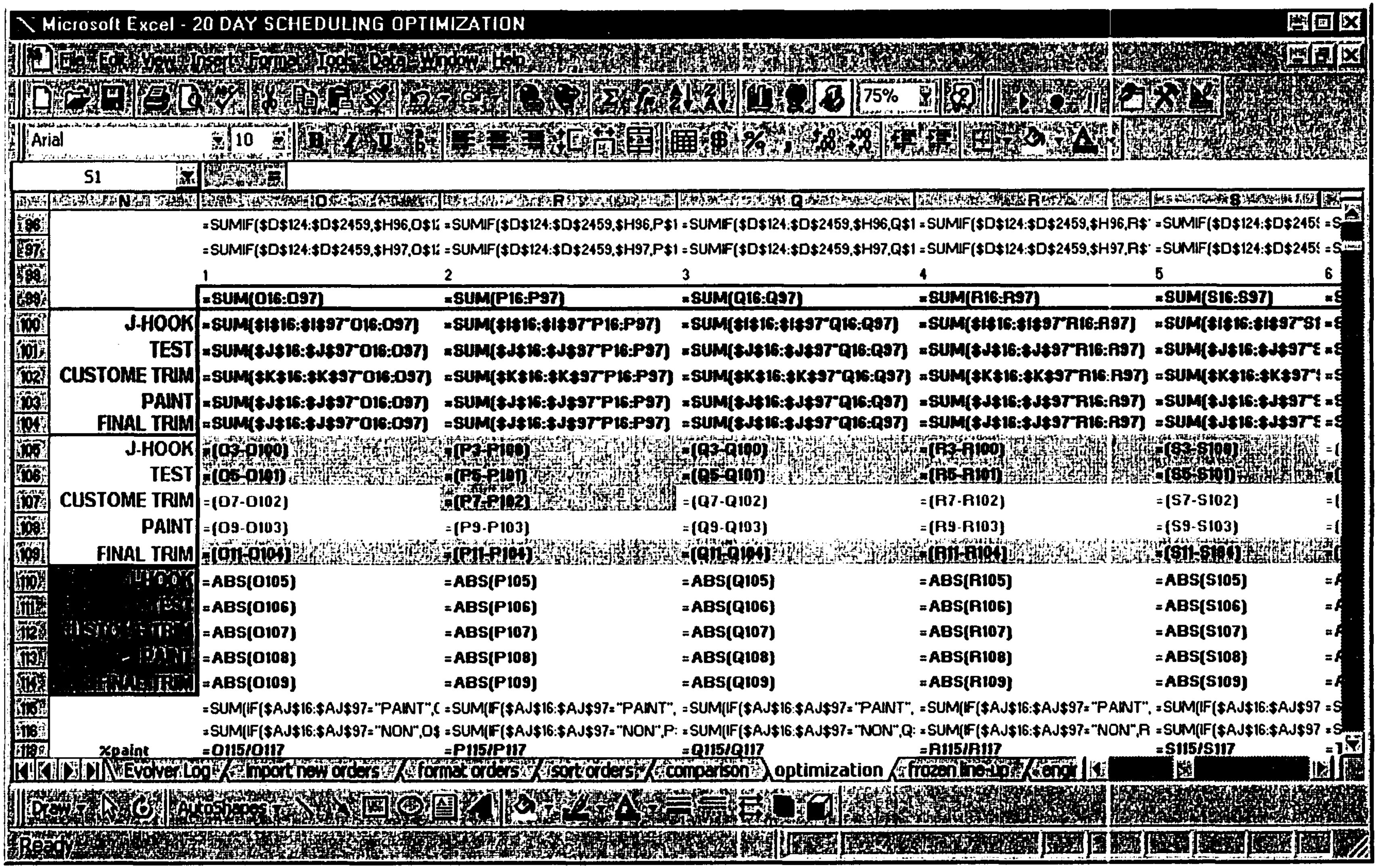
	N	O	P	Q	R	S	T	U	
1									
2	OT	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
3	J-HOOK	=(O2*$G$11)+$G$11	=(P2*$G$11)+$G$11	=(Q2*$G$11)+$G$11	=(R2*$G$11)+$G$11	=(S2*$G$11)+$G$11	=(T2*$G$11)+$G$11	=(U2*$G$11)+$	=(V2
4	OT	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
5	TEST	=(O4*$H$11)+$H$11	=(P4*$H$11)+$H$11	=(Q4*$H$11)+$H$11	=(R4*$H$11)+$H$11	=(S4*$H$11)+$H$11	=(T4*$H$11)+$H$11	=(U4*$H$11)+$	=(V4
6	OT	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
7	CUSTOM TRIM	=(O6*$I$11)+$I$11	=(P6*$I$11)+$I$11	=(Q6*$I$11)+$I$11	=(R6*$I$11)+$I$11	=(S6*$I$11)+$I$11	=(T6*$I$11)+$I$11	=(U6*$I$11)+$I	=(V6
8	OT	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
9	PAINT	=(O8*$J$11)+$J$11	=(P8*$J$11)+$J$11	=(Q8*$J$11)+$J$11	=(R8*$J$11)+$J$11	=(S8*$J$11)+$J$11	=(T8*$J$11)-$J$11	=(U8*$J$11)+$	=(V8
10	OT	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
11	FINAL TRIM	=(O10*$K$11)+$K$11	=(P10*$K$11)+$K$11	=(Q10*$K$11)+$K$11	=(R10*$K$11)+$K$11	=(S10*$K$11)+$K$11	=(T10*$K$11)+$K$11	=(U10*$K$11)+:	=(V1
12									
13									
14		1	2	3	4	5	6	7	8
15		36467	36468	36469	36472	36473	36474	36475	364
16		=SUMIF($D$124:$D$24	=SUMIF($D$124:$D$24!	=SUMIF($D$124:$D$245	=SUMIF($D$124:$D$245:	=SUMIF($D$124:$D$245!	=SUMIF($D$1:4:$D$2459	=SUMIF($D$124:	=SUN
17		=SUMIF($D$124:$D$24	=SUMIF($D$124:$D$24!	=SUMIF($D$124:$D$245	=SUMIF($D$124:$D$245:	=SUMIF($D$124:$D$245!	=SUMIF($D$1:4:$D$2459	=SUMIF($D$124:	=SUN
18		=SUMIF($D$124:$D$24	=SUMIF($D$124:$D$24!	=SUMIF($D$124:$D$245	=SUMIF($D$124:$D$245:	=SUMIF($D$124:$D$245!	=SUMIF($D$1:4:$D$2459	=SUMIF($D$124:	=SUN
19		=SUMIF($D$124:$D$24	=SUMIF($D$124:$D$24!	=SUMIF($D$124:$D$245	=SUMIF($D$124:$D$245:	=SUMIF($D$124:$D$245!	=SUMIF($D$1:4:$D$2459	=SUMIF($D$124:	=SUN
20		=SUMIF($D$124:$D$24	=SUMIF($D$124:$D$24!	=SUMIF($D$124:$D$245	=SUMIF($D$124:$D$245:	=SUMIF($D$124:$D$245!	=SUMIF($D$1:4:$D$2459	=SUMIF($D$124:	=SUN

Figure 19. Optimization tab, available capacity section, with formulas visible in the cells.



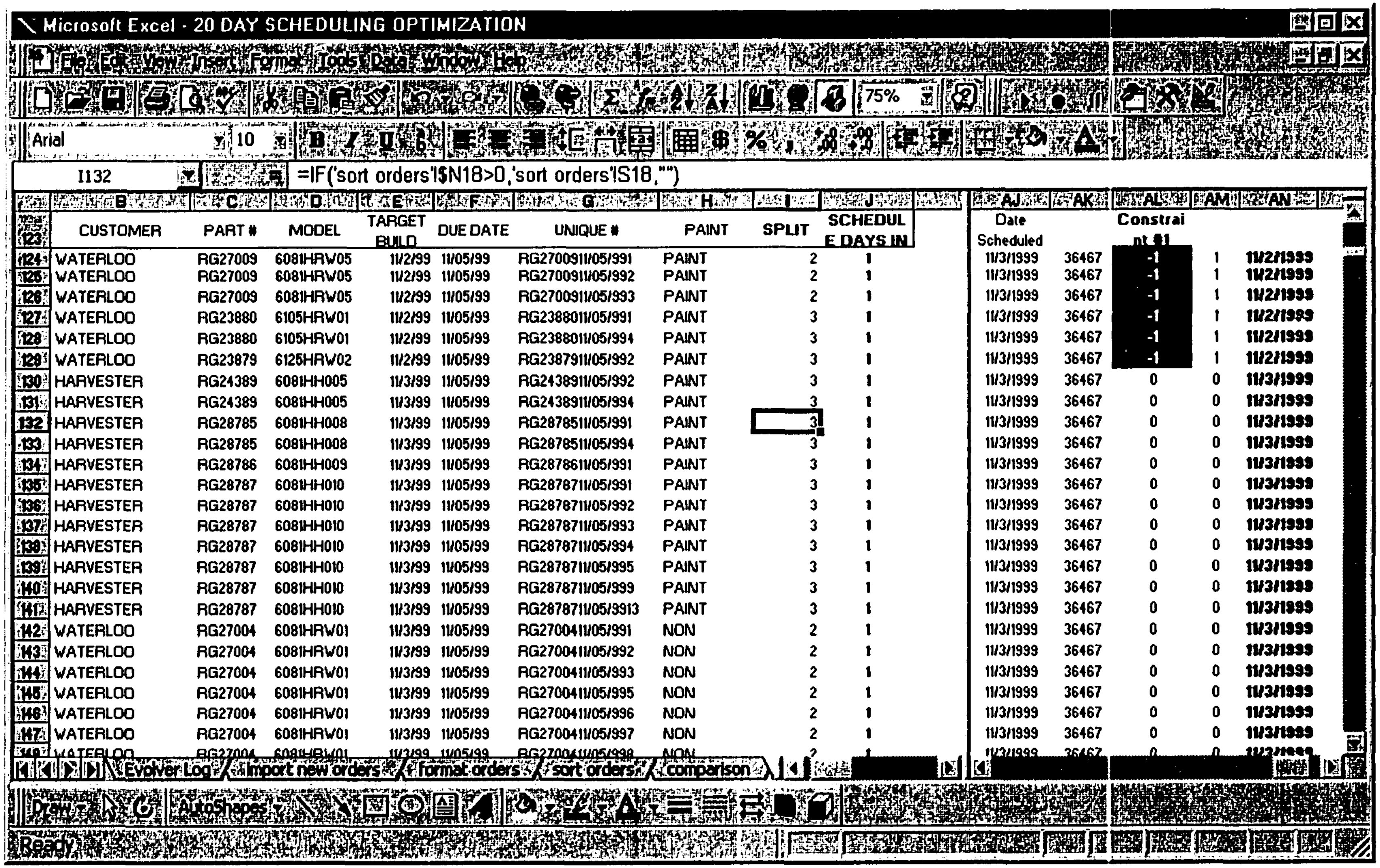
	M	N	O	P	Q	R	S	T	U	V	V	X	Y	Z	AA	AB	AC	AD	AE	AF	AG
98			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
99			111	119	111	118	118	120	122	117	115	101	121	102	119	112	117	124	122	117	110
100		J-HOOK	16224	17288	16800	17529	17452	17644	17967	17176	16964	14811	17797	14972	17450	16237	17325	18207	18187	17246	16024
101		TEST	3743	3703	3805	3453	3637	3840	3549	3474	3822	3181	4006	3458	3463	3404	3710	3880	4082	3371	2984
102		CUSTOME TRIM	2715	2469	4928	4059	3438	2852	3239	3024	3025	2631	3810	2494	3191	1963	3109	3146	3820	3712	2580
103		PAINT	3743	3703	3805	3453	3637	3840	3549	3474	3822	3181	4006	3458	3463	3404	3710	3880	4082	3371	2984
104		FINAL TRIM	3743	3703	3805	3453	3637	3840	3549	3474	3822	3181	4006	3458	3463	3404	3710	3880	4082	3371	2984
105		J-HOOK	1388	324	813	84	161	-32	-354	437	648	2802	-185	2641	163	1375	288	-594	-575	367	1589
106		TEST	1677	1717	1614	1966	1783	1580	1870	1945	1597	2239	1414	1962	1956	2015	1710	1539	1338	2048	2436
107		CUSTOME TRIM	-244	1	-2458	-1588	-968	-382	-769	-554	-555	-160	-1340	-23	-720	508	-639	-676	-1350	-1242	-110
108		PAINT	-1766	-1726	-1829	-1477	-1660	-1863	-1572	-1498	-1845	-1204	-2029	-1481	-1486	-1427	-1733	-1903	-2105	-1395	-1007
109		FINAL TRIM	704	744	642	994	810	607	898	973	625	1266	441	989	984	1043	737	567	365	1076	1463
110		J-HOOK	1388	324	813	84	161	32	354	437	648	2802	185	2641	163	1375	288	594	575	367	1589
111		TEST	1677	1717	1614	1966	1783	1580	1870	1945	1597	2239	1414	1962	1956	2015	1710	1539	1338	2048	2436
112		CUSTOME TRIM	244	1	2458	1588	968	382	769	554	555	160	1340	23	720	508	639	676	1350	1242	110
113		PAINT	1766	1726	1829	1477	1660	1863	1572	1498	1845	1204	2029	1481	1486	1427	1733	1903	2105	1395	1007
114		FINAL TRIM	704	744	642	994	810	607	898	973	625	1266	441	989	984	1043	737	567	365	1076	1463
115			58	62	98	102	89	75	74	69	69	54	71	45	68	44	71	74	86	79	66
116			53	57	13	16	29	45	48	48	46	47	50	57	51	68	46	50	36	38	44
118		xpaint	52x	52x	88x	86x	75x	63x	61x	59x	60x	53x	59x	44x	57x	39x	61x	60x	70x	68x	60x
121		10.5/12.5	22	16	22	16	17	21	22	18	24	16	20	23	9	4	17	19	27	23	14
122																					

Figure 20. Optimization tab, required capacity section.



	N	O	P	Q	R	S	
96		=SUMIF($D$124:$D$2459,$H96,O$1	=SUMIF($D$124:$D$2459,$H96,P$1	=SUMIF($D$124:$D$2459,$H96,Q$1	=SUMIF($D$124:$D$2459,$H96,R$	=SUMIF($D$124:$D$245	=S
97		=SUMIF($D$124:$D$2459,$H97,O$1	=SUMIF($D$124:$D$2459,$H97,P$1	=SUMIF($D$124:$D$2459,$H97,Q$1	=SUMIF($D$124:$D$2459,$H97,R$	=SUMIF($D$124:$D$245	=S
98		1	2	3	4	5	6
99		=SUM(O16:O97)	=SUM(P16:P97)	=SUM(Q16:Q97)	=SUM(R16:R97)	=SUM(S16:S97)	=S
100	J-HOOK	=SUM($I$16:$I$97*O16:O97)	=SUM($I$16:$I$97*P16:P97)	=SUM($I$16:$I$97*Q16:Q97)	=SUM($I$16:$I$97*R16:R97)	=SUM($I$16:$I$97*S1	=S
101	TEST	=SUM($J$16:$J$97*O16:O97)	=SUM($J$16:$J$97*P16:P97)	=SUM($J$16:$J$97*Q16:Q97)	=SUM($J$16:$J$97*R16:R97)	=SUM($J$16:$J$97*S	=S
102	CUSTOME TRIM	=SUM($K$16:$K$97*O16:O97)	=SUM($K$16:$K$97*P16:P97)	=SUM($K$16:$K$97*Q16:Q97)	=SUM($K$16:$K$97*R16:R97)	=SUM($K$16:$K$97*S	=S
103	PAINT	=SUM($J$16:$J$97*O16:O97)	=SUM($J$16:$J$97*P16:P97)	=SUM($J$16:$J$97*Q16:Q97)	=SUM($J$16:$J$97*R16:R97)	=SUM($J$16:$J$97*S	=S
104	FINAL TRIM	=SUM($J$16:$J$97*O16:O97)	=SUM($J$16:$J$97*P16:P97)	=SUM($J$16:$J$97*Q16:Q97)	=SUM($J$16:$J$97*R16:R97)	=SUM($J$16:$J$97*S	=S
105	J-HOOK	=(O3-O100)	=(P3-P100)	=(Q3-Q100)	=(R3-R100)	=(S3-S100)	=(
106	TEST	=(O5-O101)	=(P5-P101)	=(Q5-Q101)	=(R5-R101)	=(S5-S101)	=(
107	CUSTOME TRIM	=(O7-O102)	=(P7-P102)	=(Q7-Q102)	=(R7-R102)	=(S7-S102)	=(
108	PAINT	=(O9-O103)	=(P9-P103)	=(Q9-Q103)	=(R9-R103)	=(S9-S103)	=(
109	FINAL TRIM	=(O11-O104)	=(P11-P104)	=(Q11-Q104)	=(R11-R104)	=(S11-S104)	=(
110	J-HOOK	=ABS(O105)	=ABS(P105)	=ABS(Q105)	=ABS(R105)	=ABS(S105)	=A
111	TEST	=ABS(O106)	=ABS(P106)	=ABS(Q106)	=ABS(R106)	=ABS(S106)	=A
112	CUSTOME TRIM	=ABS(O107)	=ABS(P107)	=ABS(Q107)	=ABS(R107)	=ABS(S107)	=A
113	PAINT	=ABS(O108)	=ABS(P108)	=ABS(Q108)	=ABS(R108)	=ABS(S108)	=A
114	FINAL TRIM	=ABS(O109)	=ABS(P109)	=ABS(Q109)	=ABS(R109)	=ABS(S109)	=A
115		=SUM(IF($AJ$16:$AJ$97="PAINT",(	=SUM(IF($AJ$16:$AJ$97="PAINT",	=SUM(IF($AJ$16:$AJ$97="PAINT",	=SUM(IF($AJ$16:$AJ$97="PAINT",	=SUM(IF($AJ$16:$AJ$97	=S
116		=SUM(IF($AJ$16:$AJ$97="NON",O$	=SUM(IF($AJ$16:$AJ$97="NON",P:	=SUM(IF($AJ$16:$AJ$97="NON",Q:	=SUM(IF($AJ$16:$AJ$97="NON",R	=SUM(IF($AJ$16:$AJ$97	=S
118	%paint	=O115/O117	=P115/P117	=Q115/Q117	=R115/R117	=S115/S117	=1

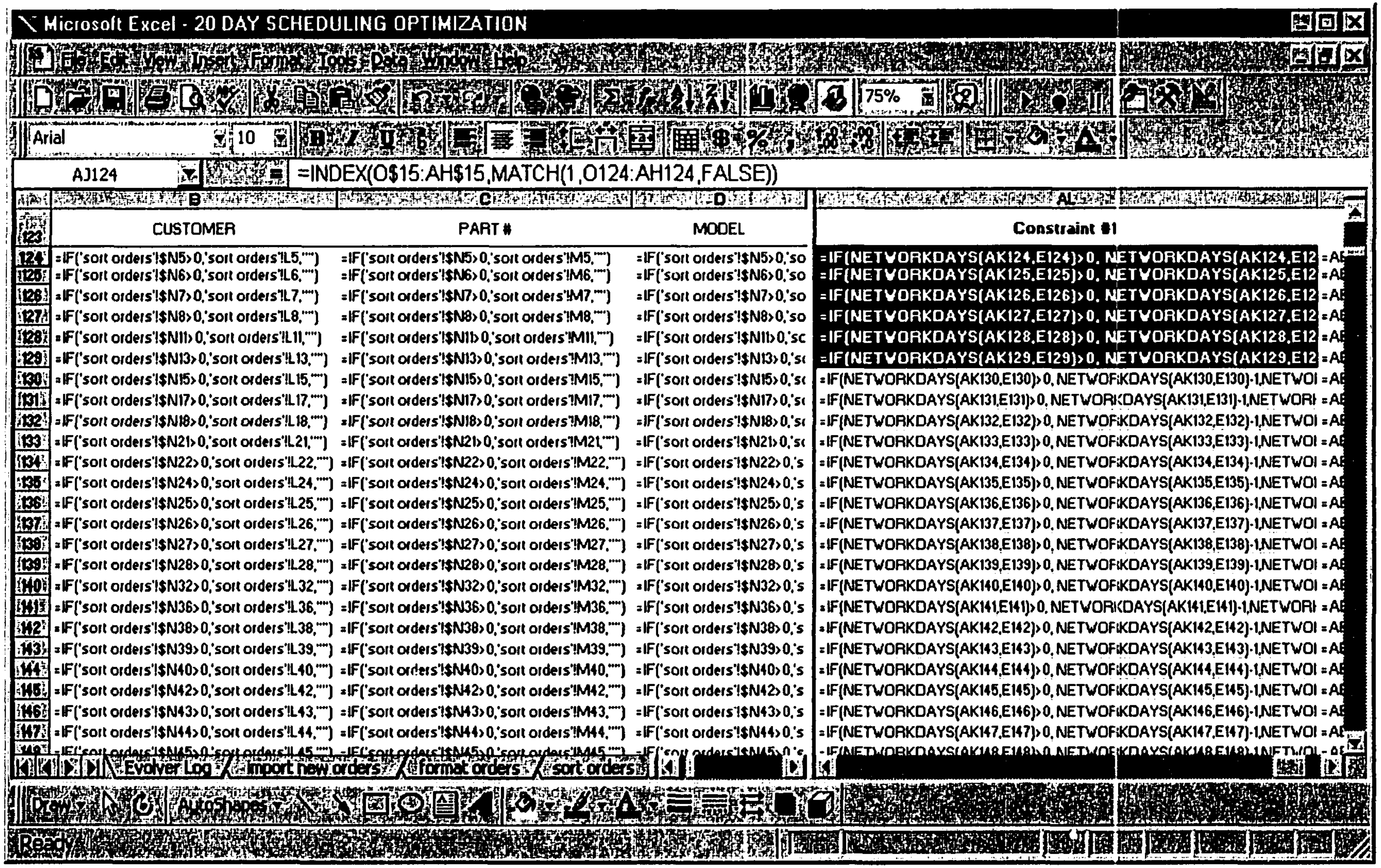
Figure 21. Optimization tab, required capacity section, with formulas visible in the cells.



I132 =IF('sort orders'!$N18>0,'sort orders'!S18,"")

	B	C	D	E	F	G	H	I	J	AJ	AK	AL	AM	AN
123	CUSTOMER	PART #	MODEL	TARGET BUILD	DUE DATE	UNIQUE #	PAINT	SPLIT	SCHEDULE DAYS IN	Date Scheduled		Constraint #1		
124	WATERLOO	RG27009	6081HRW05	11/2/99	11/05/99	RG2700911/05/991	PAINT	2	1	11/3/1999	36467	-1	1	11/2/1999
125	WATERLOO	RG27009	6081HRW05	11/2/99	11/05/99	RG2700911/05/992	PAINT	2	1	11/3/1999	36467	-1	1	11/2/1999
126	WATERLOO	RG27009	6081HRW05	11/2/99	11/05/99	RG2700911/05/993	PAINT	2	1	11/3/1999	36467	-1	1	11/2/1999
127	WATERLOO	RG23880	6105HRW01	11/2/99	11/05/99	RG2388011/05/991	PAINT	3	1	11/3/1999	36467	-1	1	11/2/1999
128	WATERLOO	RG23880	6105HRW01	11/2/99	11/05/99	RG2388011/05/994	PAINT	3	1	11/3/1999	36467	-1	1	11/2/1999
129	WATERLOO	RG23879	6125HRW02	11/2/99	11/05/99	RG2387911/05/992	PAINT	3	1	11/3/1999	36467	-1	1	11/2/1999
130	HARVESTER	RG24389	6081HH005	11/3/99	11/05/99	RG2438911/05/992	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
131	HARVESTER	RG24389	6081HH005	11/3/99	11/05/99	RG2438911/05/994	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
132	HARVESTER	RG28785	6081HH008	11/3/99	11/05/99	RG2878511/05/991	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
133	HARVESTER	RG28785	6081HH008	11/3/99	11/05/99	RG2878511/05/994	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
134	HARVESTER	RG28786	6081HH009	11/3/99	11/05/99	RG2878611/05/991	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
135	HARVESTER	RG28787	6081HH010	11/3/99	11/05/99	RG2878711/05/991	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
136	HARVESTER	RG28787	6081HH010	11/3/99	11/05/99	RG2878711/05/992	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
137	HARVESTER	RG28787	6081HH010	11/3/99	11/05/99	RG2878711/05/993	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
138	HARVESTER	RG28787	6081HH010	11/3/99	11/05/99	RG2878711/05/994	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
139	HARVESTER	RG28787	6081HH010	11/3/99	11/05/99	RG2878711/05/995	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
140	HARVESTER	RG28787	6081HH010	11/3/99	11/05/99	RG2878711/05/999	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
141	HARVESTER	RG28787	6081HH010	11/3/99	11/05/99	RG2878711/05/9913	PAINT	3	1	11/3/1999	36467	0	0	11/3/1999
142	WATERLOO	RG27004	6081HRW01	11/3/99	11/05/99	RG2700411/05/991	NON	2	1	11/3/1999	36467	0	0	11/3/1999
143	WATERLOO	RG27004	6081HRW01	11/3/99	11/05/99	RG2700411/05/992	NON	2	1	11/3/1999	36467	0	0	11/3/1999
144	WATERLOO	RG27004	6081HRW01	11/3/99	11/05/99	RG2700411/05/993	NON	2	1	11/3/1999	36467	0	0	11/3/1999
145	WATERLOO	RG27004	6081HRW01	11/3/99	11/05/99	RG2700411/05/995	NON	2	1	11/3/1999	36467	0	0	11/3/1999
146	WATERLOO	RG27004	6081HRW01	11/3/99	11/05/99	RG2700411/05/996	NON	2	1	11/3/1999	36467	0	0	11/3/1999
147	WATERLOO	RG27004	6081HRW01	11/3/99	11/05/99	RG2700411/05/997	NON	2	1	11/3/1999	36467	0	0	11/3/1999
148	WATERLOO	RG27004	6081HRW01	11/3/99	11/05/99	RG2700411/05/998	NON	2	1	11/3/1999	36467	0	0	11/3/1999

Figure 22. Optimization tab, scheduled orders with ship date relative to each customer.

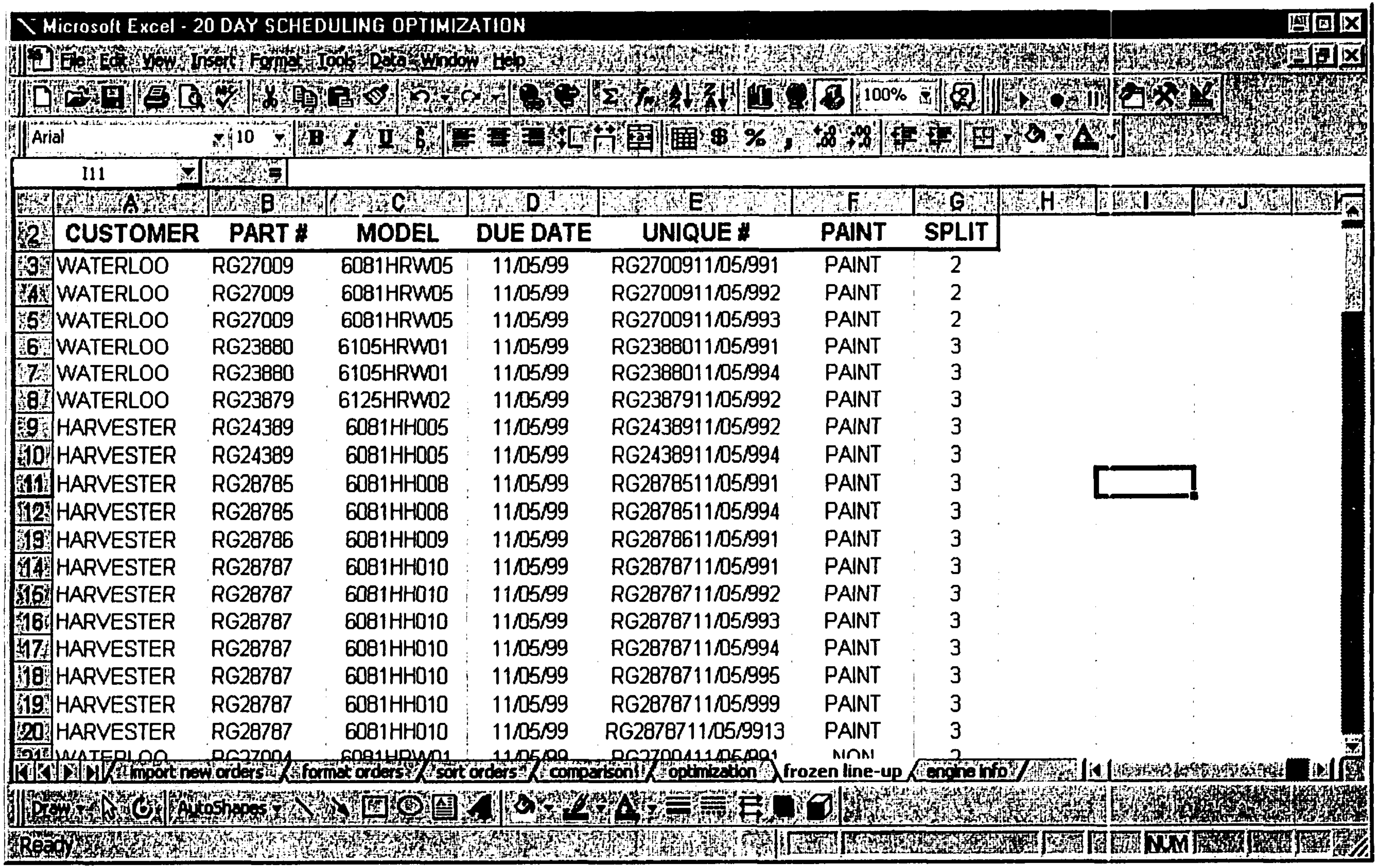


AJ124 =INDEX(O$15:AH$15,MATCH(1,O124:AH124,FALSE))

123	CUSTOMER	PART #	MODEL	Constraint #1	
124	=IF('sort orders'!$N5>0,'sort orders'!L5,"")	=IF('sort orders'!$N5>0,'sort orders'!M5,"")	=IF('sort orders'!$N5>0,'so	=IF(NETWORKDAYS(AK124,E124)>0, NETWORKDAYS(AK124,E12	=A
125	=IF('sort orders'!$N6>0,'sort orders'!L6,"")	=IF('sort orders'!$N6>0,'sort orders'!M6,"")	=IF('sort orders'!$N6>0,'so	=IF(NETWORKDAYS(AK125,E125)>0, NETWORKDAYS(AK125,E12	=A
126	=IF('sort orders'!$N7>0,'sort orders'!L7,"")	=IF('sort orders'!$N7>0,'sort orders'!M7,"")	=IF('sort orders'!$N7>0,'so	=IF(NETWORKDAYS(AK126,E126)>0, NETWORKDAYS(AK126,E12	=A
127	=IF('sort orders'!$N8>0,'sort orders'!L8,"")	=IF('sort orders'!$N8>0,'sort orders'!M8,"")	=IF('sort orders'!$N8>0,'so	=IF(NETWORKDAYS(AK127,E127)>0, NETWORKDAYS(AK127,E12	=A
128	=IF('sort orders'!$N11>0,'sort orders'!L11,"")	=IF('sort orders'!$N11>0,'sort orders'!M11,"")	=IF('sort orders'!$N11>0,'sc	=IF(NETWORKDAYS(AK128,E128)>0, NETWORKDAYS(AK128,E12	=A
129	=IF('sort orders'!$N13>0,'sort orders'!L13,"")	=IF('sort orders'!$N13>0,'sort orders'!M13,"")	=IF('sort orders'!$N13>0,'s	=IF(NETWORKDAYS(AK129,E129)>0, NETWORKDAYS(AK129,E12	=A
130	=IF('sort orders'!$N15>0,'sort orders'!L15,"")	=IF('sort orders'!$N15>0,'sort orders'!M15,"")	=IF('sort orders'!$N15>0,'s	=IF(NETWORKDAYS(AK130,E130)>0, NETWORKDAYS(AK130,E130)-1,NETWO	=A
131	=IF('sort orders'!$N17>0,'sort orders'!L17,"")	=IF('sort orders'!$N17>0,'sort orders'!M17,"")	=IF('sort orders'!$N17>0,'s	=IF(NETWORKDAYS(AK131,E131)>0, NETWORKDAYS(AK131,E131)-1,NETWOR	=A
132	=IF('sort orders'!$N18>0,'sort orders'!L18,"")	=IF('sort orders'!$N18>0,'sort orders'!M18,"")	=IF('sort orders'!$N18>0,'s	=IF(NETWORKDAYS(AK132,E132)>0, NETWORKDAYS(AK132,E132)-1,NETWO	=A
133	=IF('sort orders'!$N21>0,'sort orders'!L21,"")	=IF('sort orders'!$N21>0,'sort orders'!M21,"")	=IF('sort orders'!$N21>0,'s	=IF(NETWORKDAYS(AK133,E133)>0, NETWORKDAYS(AK133,E133)-1,NETWO	=A
134	=IF('sort orders'!$N22>0,'sort orders'!L22,"")	=IF('sort orders'!$N22>0,'sort orders'!M22,"")	=IF('sort orders'!$N22>0,'s	=IF(NETWORKDAYS(AK134,E134)>0, NETWORKDAYS(AK134,E134)-1,NETWO	=A
135	=IF('sort orders'!$N24>0,'sort orders'!L24,"")	=IF('sort orders'!$N24>0,'sort orders'!M24,"")	=IF('sort orders'!$N24>0,'s	=IF(NETWORKDAYS(AK135,E135)>0, NETWORKDAYS(AK135,E135)-1,NETWO	=A
136	=IF('sort orders'!$N25>0,'sort orders'!L25,"")	=IF('sort orders'!$N25>0,'sort orders'!M25,"")	=IF('sort orders'!$N25>0,'s	=IF(NETWORKDAYS(AK136,E136)>0, NETWORKDAYS(AK136,E136)-1,NETWO	=A
137	=IF('sort orders'!$N26>0,'sort orders'!L26,"")	=IF('sort orders'!$N26>0,'sort orders'!M26,"")	=IF('sort orders'!$N26>0,'s	=IF(NETWORKDAYS(AK137,E137)>0, NETWORKDAYS(AK137,E137)-1,NETWO	=A
138	=IF('sort orders'!$N27>0,'sort orders'!L27,"")	=IF('sort orders'!$N27>0,'sort orders'!M27,"")	=IF('sort orders'!$N27>0,'s	=IF(NETWORKDAYS(AK138,E138)>0, NETWORKDAYS(AK138,E138)-1,NETWO	=A
139	=IF('sort orders'!$N28>0,'sort orders'!L28,"")	=IF('sort orders'!$N28>0,'sort orders'!M28,"")	=IF('sort orders'!$N28>0,'s	=IF(NETWORKDAYS(AK139,E139)>0, NETWORKDAYS(AK139,E139)-1,NETWO	=A
140	=IF('sort orders'!$N32>0,'sort orders'!L32,"")	=IF('sort orders'!$N32>0,'sort orders'!M32,"")	=IF('sort orders'!$N32>0,'s	=IF(NETWORKDAYS(AK140,E140)>0, NETWORKDAYS(AK140,E140)-1,NETWO	=A
141	=IF('sort orders'!$N36>0,'sort orders'!L36,"")	=IF('sort orders'!$N36>0,'sort orders'!M36,"")	=IF('sort orders'!$N36>0,'s	=IF(NETWORKDAYS(AK141,E141)>0, NETWORKDAYS(AK141,E141)-1,NETWOR	=A
142	=IF('sort orders'!$N38>0,'sort orders'!L38,"")	=IF('sort orders'!$N38>0,'sort orders'!M38,"")	=IF('sort orders'!$N38>0,'s	=IF(NETWORKDAYS(AK142,E142)>0, NETWORKDAYS(AK142,E142)-1,NETWO	=A
143	=IF('sort orders'!$N39>0,'sort orders'!L39,"")	=IF('sort orders'!$N39>0,'sort orders'!M39,"")	=IF('sort orders'!$N39>0,'s	=IF(NETWORKDAYS(AK143,E143)>0, NETWORKDAYS(AK143,E143)-1,NETWO	=A
144	=IF('sort orders'!$N40>0,'sort orders'!L40,"")	=IF('sort orders'!$N40>0,'sort orders'!M40,"")	=IF('sort orders'!$N40>0,'s	=IF(NETWORKDAYS(AK144,E144)>0, NETWORKDAYS(AK144,E144)-1,NETWO	=A
145	=IF('sort orders'!$N42>0,'sort orders'!L42,"")	=IF('sort orders'!$N42>0,'sort orders'!M42,"")	=IF('sort orders'!$N42>0,'s	=IF(NETWORKDAYS(AK145,E145)>0, NETWORKDAYS(AK145,E145)-1,NETWO	=A
146	=IF('sort orders'!$N43>0,'sort orders'!L43,"")	=IF('sort orders'!$N43>0,'sort orders'!M43,"")	=IF('sort orders'!$N43>0,'s	=IF(NETWORKDAYS(AK146,E146)>0, NETWORKDAYS(AK146,E146)-1,NETWO	=A
147	=IF('sort orders'!$N44>0,'sort orders'!L44,"")	=IF('sort orders'!$N44>0,'sort orders'!M44,"")	=IF('sort orders'!$N44>0,'s	=IF(NETWORKDAYS(AK147,E147)>0, NETWORKDAYS(AK147,E147)-1,NETWO	=A

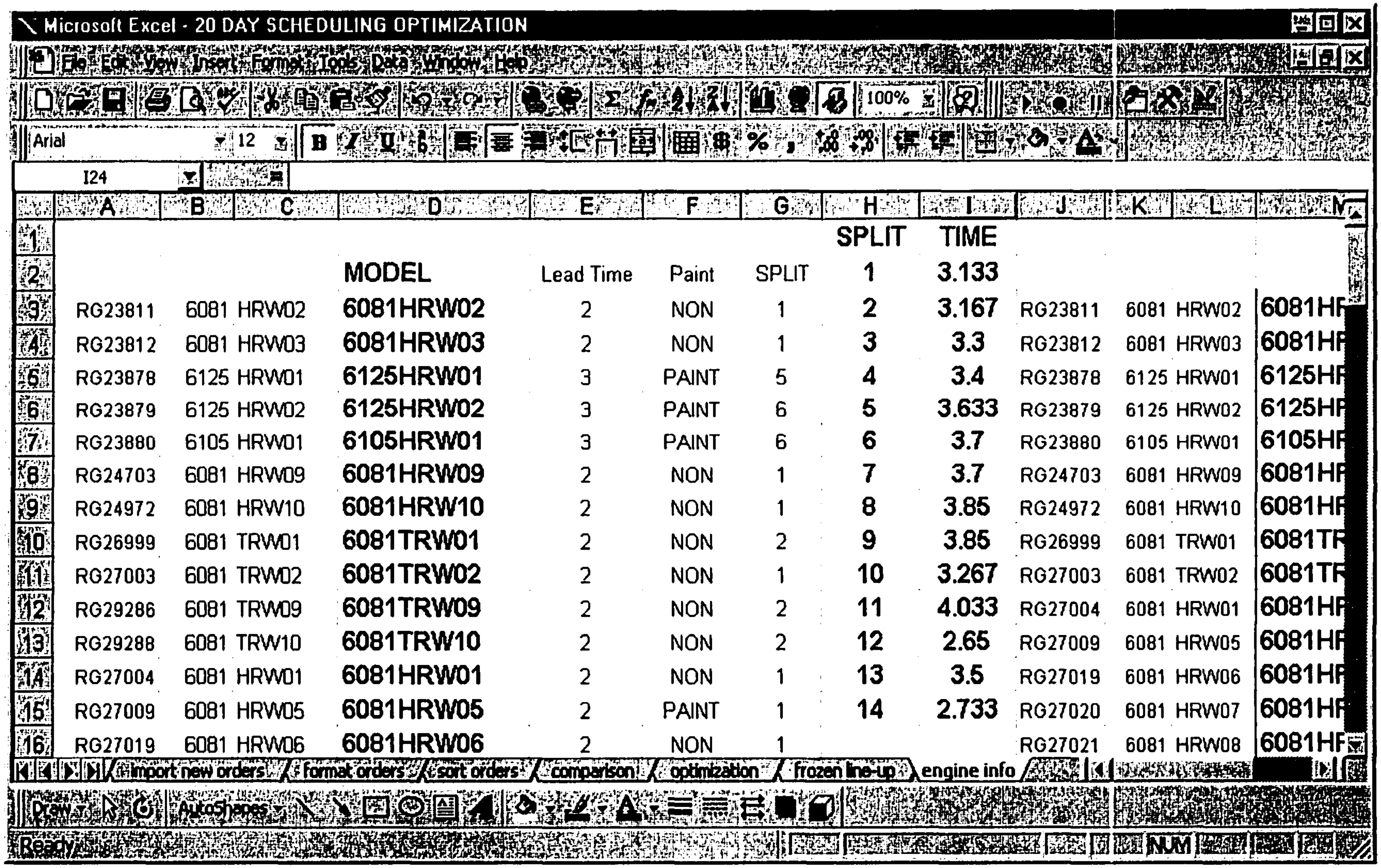
Figure 23. Optimization tab, scheduled orders with ship date relative to each customer, formulas visible in the cells.

After the optimization is performed using genetic algorithms, the schedule for the first day is frozen (Figure 24). These orders are linked to the next spreadsheet titled "frozen line-up" in the 20-day scheduling optimization model. Orders are compared with these frozen orders before they are included in the optimization to eliminate any duplication. These orders are also linked to the sequencing part of the model called sequencing model, which is discussed in detail in the next section. The last sheet in the model (Figure 25) titled "engine info," includes part number, engine model, lead time in days and split time in minutes. This information is used for the final assembly line (j-hook).



	CUSTOMER	PART #	MODEL	DUE DATE	UNIQUE #	PAINT	SPLIT
3	WATERLOO	RG27009	6081HRW05	11/05/99	RG2700911/05/991	PAINT	2
4	WATERLOO	RG27009	6081HRW05	11/05/99	RG2700911/05/992	PAINT	2
5	WATERLOO	RG27009	6081HRW05	11/05/99	RG2700911/05/993	PAINT	2
6	WATERLOO	RG23880	6105HRW01	11/05/99	RG2388011/05/991	PAINT	3
7	WATERLOO	RG23880	6105HRW01	11/05/99	RG2388011/05/994	PAINT	3
8	WATERLOO	RG23879	6125HRW02	11/05/99	RG2387911/05/992	PAINT	3
9	HARVESTER	RG24389	6081HH005	11/05/99	RG2438911/05/992	PAINT	3
10	HARVESTER	RG24389	6081HH005	11/05/99	RG2438911/05/994	PAINT	3
11	HARVESTER	RG28785	6081HH008	11/05/99	RG2878511/05/991	PAINT	3
12	HARVESTER	RG28785	6081HH008	11/05/99	RG2878511/05/994	PAINT	3
13	HARVESTER	RG28786	6081HH009	11/05/99	RG2878611/05/991	PAINT	3
14	HARVESTER	RG28787	6081HH010	11/05/99	RG2878711/05/991	PAINT	3
15	HARVESTER	RG28787	6081HH010	11/05/99	RG2878711/05/992	PAINT	3
16	HARVESTER	RG28787	6081HH010	11/05/99	RG2878711/05/993	PAINT	3
17	HARVESTER	RG28787	6081HH010	11/05/99	RG2878711/05/994	PAINT	3
18	HARVESTER	RG28787	6081HH010	11/05/99	RG2878711/05/995	PAINT	3
19	HARVESTER	RG28787	6081HH010	11/05/99	RG2878711/05/999	PAINT	3
20	HARVESTER	RG28787	6081HH010	11/05/99	RG2878711/05/9913	PAINT	3

Figure 24. Frozen line-up tab.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1								SPLIT	TIME				
2				MODEL	Lead Time	Paint	SPLIT	1	3.133				
3	RG23811	6081	HRW02	6081HRW02	2	NON	1	2	3.167	RG23811	6081	HRW02	6081H
4	RG23812	6081	HRW03	6081HRW03	2	NON	1	3	3.3	RG23812	6081	HRW03	6081H
5	RG23878	6125	HRW01	6125HRW01	3	PAINT	5	4	3.4	RG23878	6125	HRW01	6125H
6	RG23879	6125	HRW02	6125HRW02	3	PAINT	6	5	3.633	RG23879	6125	HRW02	6125H
7	RG23880	6105	HRW01	6105HRW01	3	PAINT	6	6	3.7	RG23880	6105	HRW01	6105H
8	RG24703	6081	HRW09	6081HRW09	2	NON	1	7	3.7	RG24703	6081	HRW09	6081H
9	RG24972	6081	HRW10	6081HRW10	2	NON	1	8	3.85	RG24972	6081	HRW10	6081H
10	RG26999	6081	TRW01	6081TRW01	2	NON	2	9	3.85	RG26999	6081	TRW01	6081T
11	RG27003	6081	TRW02	6081TRW02	2	NON	1	10	3.267	RG27003	6081	TRW02	6081T
12	RG29286	6081	TRW09	6081TRW09	2	NON	2	11	4.033	RG27004	6081	HRW01	6081H
13	RG29288	6081	TRW10	6081TRW10	2	NON	2	12	2.65	RG27009	6081	HRW05	6081H
14	RG27004	6081	HRW01	6081HRW01	2	NON	1	13	3.5	RG27019	6081	HRW06	6081H
15	RG27009	6081	HRW05	6081HRW05	2	PAINT	1	14	2.733	RG27020	6081	HRW07	6081H
16	RG27019	6081	HRW06	6081HRW06	2	NON	1			RG27021	6081	HRW08	6081H

Figure 25. Engine info tab.

To summarize the 20-day scheduling model: after orders are assigned to the first production day, that day's production schedule is frozen; no additions or deletions can be made to the schedule. Once the first day is frozen, it is linked to the sequencing model, which is discussed in the next section.

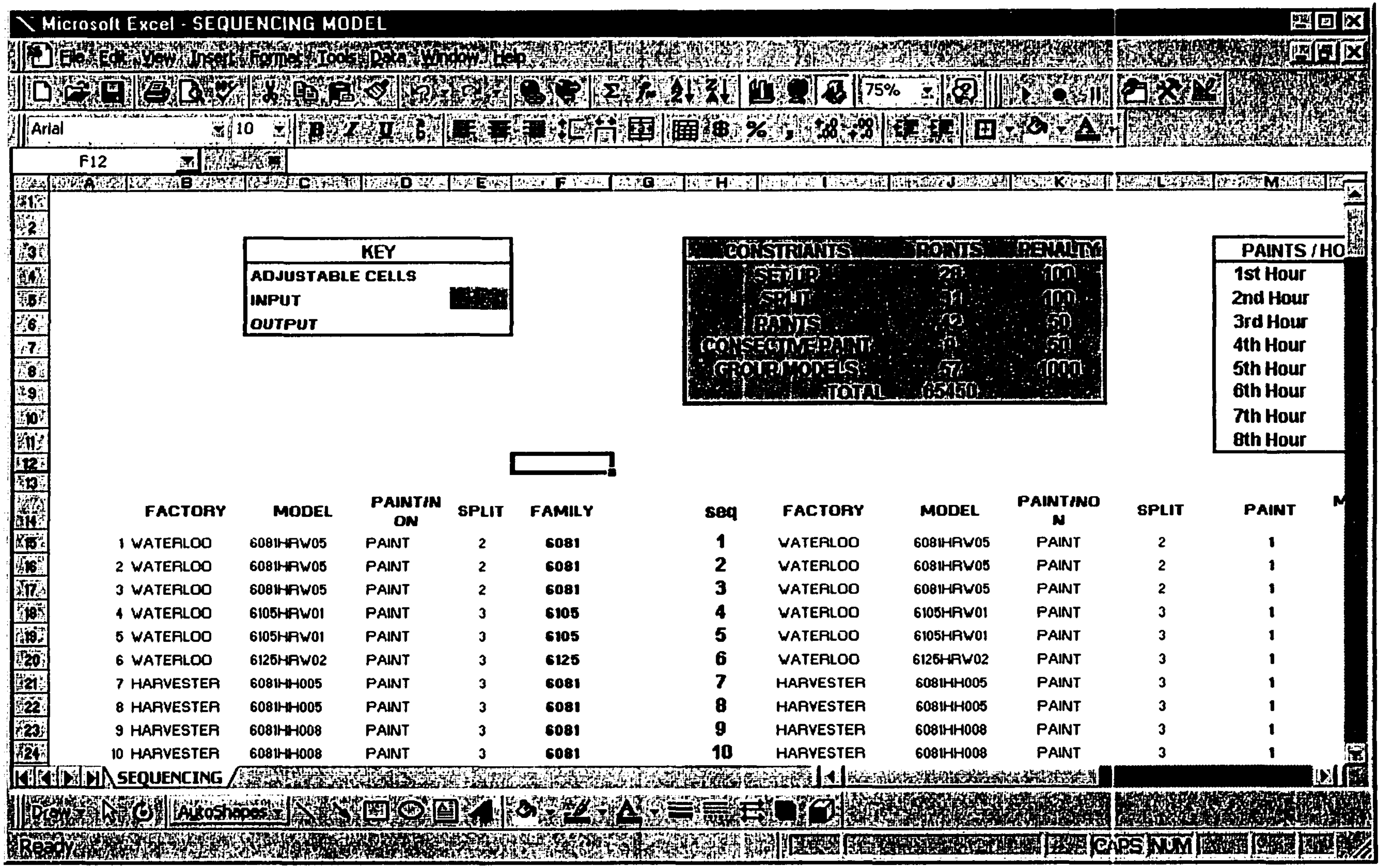
Sequencing model. In the sequencing part of the model, sequencing of engines is performed based on the following constraint criteria:

1. Total number of set-ups at the final assembly line (J-Hook)
2. Total number of split changes at the final assembly line (J-Hook)
3. Number of painted engines built per hour
4. Avoiding continuous build of painted engines
5. Grouping of similar types of engine models together

The build schedule for the next production day is frozen every day based on the scheduling constraints mentioned in the previous section. This schedule updates the worksheet titled "frozen line-up" in the scheduling model. The frozen line-up worksheet is linked to the sequencing model (Figures 26-30). Figure 26 shows the section where constraint points and penalty assigned to each constraint are calculated (cells H3:K9). Columns B through E are linked to the frozen line-up worksheet of the 20-day scheduling model. These same columns are also updated automatically every time the frozen line-up worksheet is updated in the scheduling model.

All the constraints in the sequencing model were soft constraints for which individual constraints can be violated. However, each violation had predetermined penalty points which the model applied accordingly. Cells J3:J8 in Figure 26 indicate the violations for all five constraints. As can be seen, there were 28 violations of the setup constraint, caused by 28 setup changes resulting from the sequencing of the line up. Correspondingly, for the other four constraints, violations were as follows:

(a) 31 split changes, (b) 42 paint violations, (c) 9 consecutive paint violations, and (d) 57 group models violations. Figures 27 and 28 illustrate the same information that appears in Figure 26 but the formulas are visible in the cells. Figures 29–30 illustrate the computation of each constraint for each row with and without formulas visible in the cells.



KEY	
ADJUSTABLE CELLS	
INPUT	▇
OUTPUT	

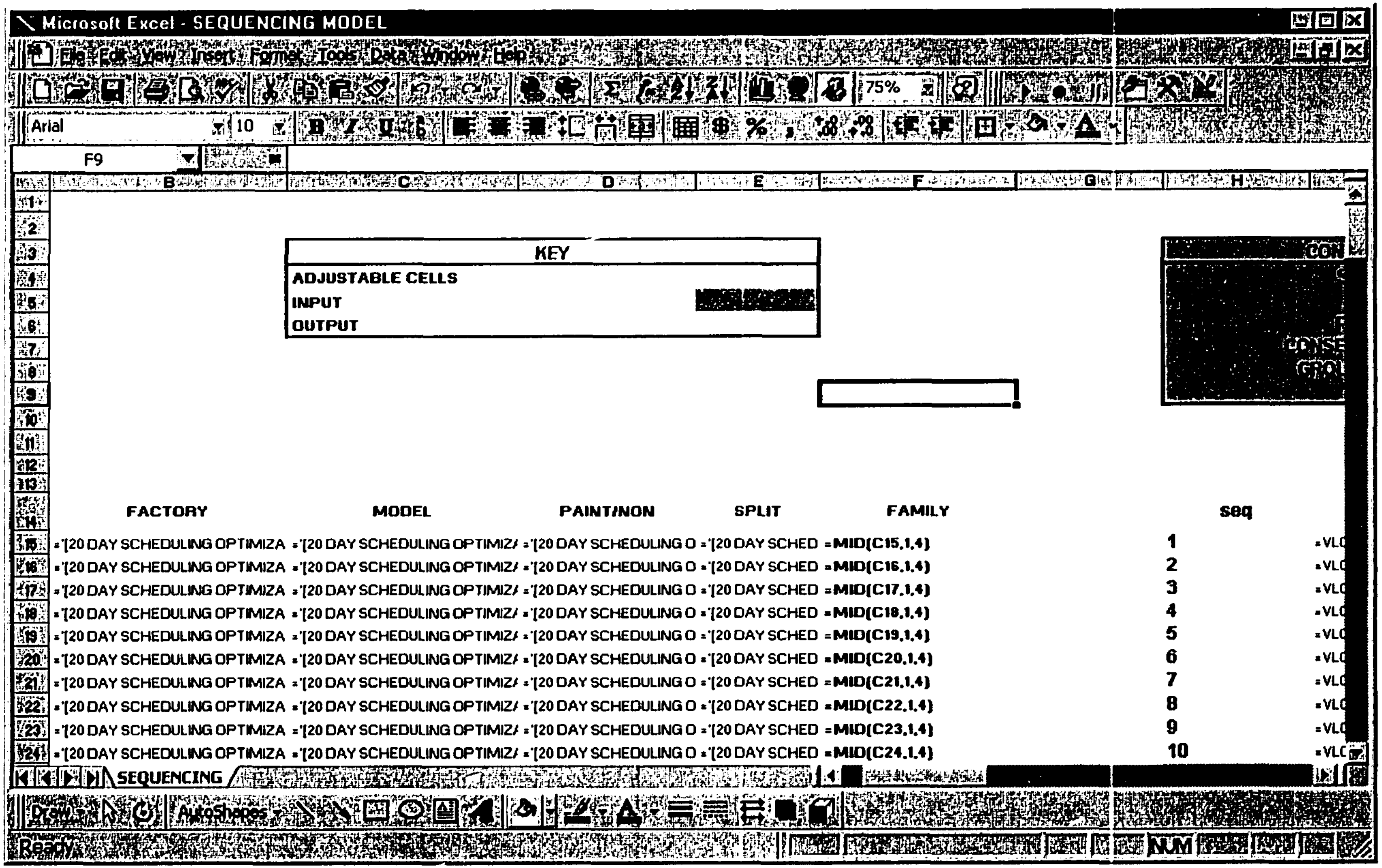
CONSTRIANTS	POINTS	PENALTY
SETUP	28	100
SPLIT	11	100
PAINTS	42	50
CONSECTIVE PAINT	0	50
GROUP MODELS	57	1000
TOTAL	65450	

PAINTS / HO
1st Hour
2nd Hour
3rd Hour
4th Hour
5th Hour
6th Hour
7th Hour
8th Hour

	FACTORY	MODEL	PAINT/NON	SPLIT	FAMILY
1	WATERLOO	6081HRW05	PAINT	2	6081
2	WATERLOO	6081HRW05	PAINT	2	6081
3	WATERLOO	6081HRW05	PAINT	2	6081
4	WATERLOO	6105HRW01	PAINT	3	6105
5	WATERLOO	6105HRW01	PAINT	3	6105
6	WATERLOO	6125HRW02	PAINT	3	6125
7	HARVESTER	6081HH005	PAINT	3	6081
8	HARVESTER	6081HH005	PAINT	3	6081
9	HARVESTER	6081HH008	PAINT	3	6081
10	HARVESTER	6081HH008	PAINT	3	6081

seq	FACTORY	MODEL	PAINT/NON	SPLIT	PAINT
1	WATERLOO	6081HRW05	PAINT	2	1
2	WATERLOO	6081HRW05	PAINT	2	1
3	WATERLOO	6081HRW05	PAINT	2	1
4	WATERLOO	6105HRW01	PAINT	3	1
5	WATERLOO	6105HRW01	PAINT	3	1
6	WATERLOO	6125HRW02	PAINT	3	1
7	HARVESTER	6081HH005	PAINT	3	1
8	HARVESTER	6081HH005	PAINT	3	1
9	HARVESTER	6081HH008	PAINT	3	1
10	HARVESTER	6081HH008	PAINT	3	1

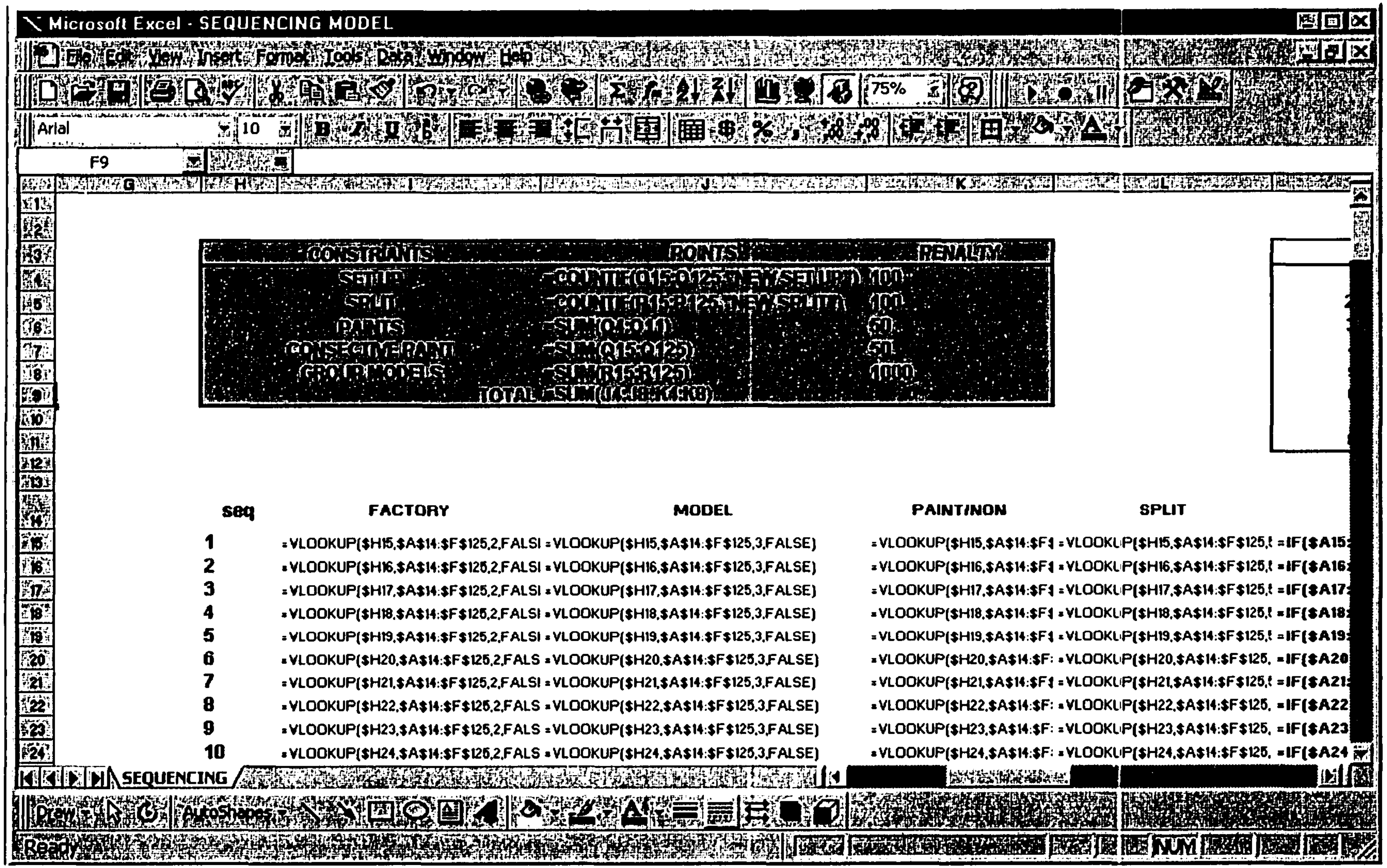
Figure 26. Sequencing tab, calculation of constraint points.



KEY	
ADJUSTABLE CELLS	
INPUT	▨
OUTPUT	

FACTORY	MODEL	PAINT/NON	SPLIT	FAMILY	seq	
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C15,1,4)	1	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C16,1,4)	2	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C17,1,4)	3	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C18,1,4)	4	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C19,1,4)	5	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C20,1,4)	6	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C21,1,4)	7	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C22,1,4)	8	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C23,1,4)	9	=VL
='[20 DAY SCHEDULING OPTIMIZA	='[20 DAY SCHEDULING OPTIMIZ	='[20 DAY SCHEDULING O	='[20 DAY SCHED	=MID(C24,1,4)	10	=VL

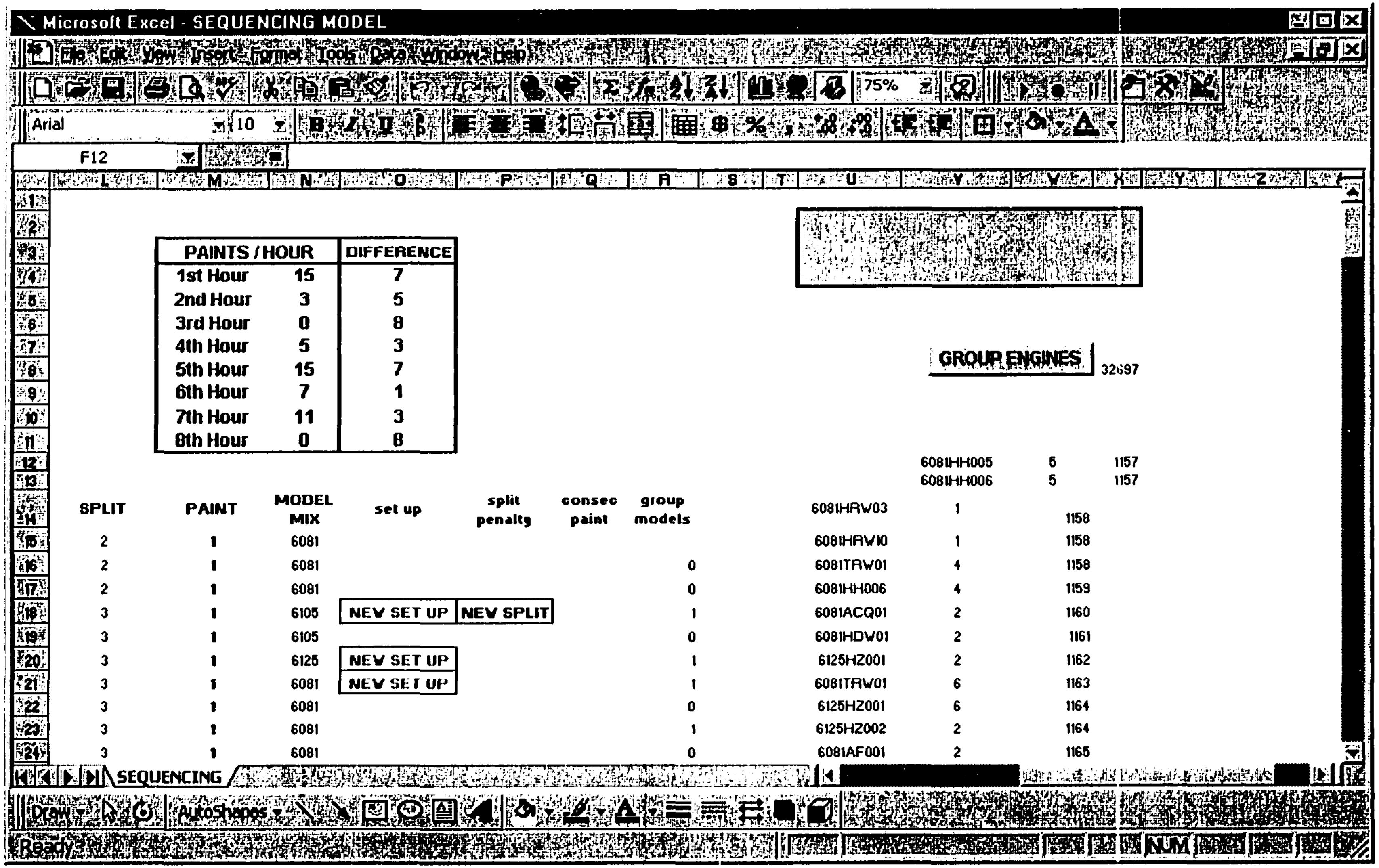
Figure 27. Sequencing tab, calculation of constraint points, with formulas visible in the cells (a).



CONSTRAINTS	POINTS	PENALTY
SETUP	=COUNTIF(Q15:Q125,"NEW SETUP")	100
SPLIT	=COUNTIF(R15:R125,"NEW SPLIT")	100
PAINTS	=SUM([illegible])	50
CONSECTIVE PAINT	=SUM(Q15:Q125)	50
GROUP MODELS	=SUM(R15:R125)	1000
TOTAL	=SUM([illegible])	

seq	FACTORY	MODEL	PAINT/NON	SPLIT	
1	=VLOOKUP($H15,$A$14:$F$125,2,FALS	=VLOOKUP($H15,$A$14:$F$125,3,FALSE)	=VLOOKUP($H15,$A$14:$F	=VLOOKUP($H15,$A$14:$F$125,	=IF($A15
2	=VLOOKUP($H16,$A$14:$F$125,2,FALS	=VLOOKUP($H16,$A$14:$F$125,3,FALSE)	=VLOOKUP($H16,$A$14:$F	=VLOOKUP($H16,$A$14:$F$125,	=IF($A16
3	=VLOOKUP($H17,$A$14:$F$125,2,FALS	=VLOOKUP($H17,$A$14:$F$125,3,FALSE)	=VLOOKUP($H17,$A$14:$F	=VLOOKUP($H17,$A$14:$F$125,	=IF($A17
4	=VLOOKUP($H18,$A$14:$F$125,2,FALS	=VLOOKUP($H18,$A$14:$F$125,3,FALSE)	=VLOOKUP($H18,$A$14:$F	=VLOOKUP($H18,$A$14:$F$125,	=IF($A18
5	=VLOOKUP($H19,$A$14:$F$125,2,FALS	=VLOOKUP($H19,$A$14:$F$125,3,FALSE)	=VLOOKUP($H19,$A$14:$F	=VLOOKUP($H19,$A$14:$F$125,	=IF($A19
6	=VLOOKUP($H20,$A$14:$F$125,2,FALS	=VLOOKUP($H20,$A$14:$F$125,3,FALSE)	=VLOOKUP($H20,$A$14:$F	=VLOOKUP($H20,$A$14:$F$125,	=IF($A20
7	=VLOOKUP($H21,$A$14:$F$125,2,FALS	=VLOOKUP($H21,$A$14:$F$125,3,FALSE)	=VLOOKUP($H21,$A$14:$F	=VLOOKUP($H21,$A$14:$F$125,	=IF($A21
8	=VLOOKUP($H22,$A$14:$F$125,2,FALS	=VLOOKUP($H22,$A$14:$F$125,3,FALSE)	=VLOOKUP($H22,$A$14:$F	=VLOOKUP($H22,$A$14:$F$125,	=IF($A22
9	=VLOOKUP($H23,$A$14:$F$125,2,FALS	=VLOOKUP($H23,$A$14:$F$125,3,FALSE)	=VLOOKUP($H23,$A$14:$F	=VLOOKUP($H23,$A$14:$F$125,	=IF($A23
10	=VLOOKUP($H24,$A$14:$F$125,2,FALS	=VLOOKUP($H24,$A$14:$F$125,3,FALSE)	=VLOOKUP($H24,$A$14:$F	=VLOOKUP($H24,$A$14:$F$125,	=IF($A24

Figure 28. Sequencing tab, calculation of constraint points, with formulas visible in the cells (b).

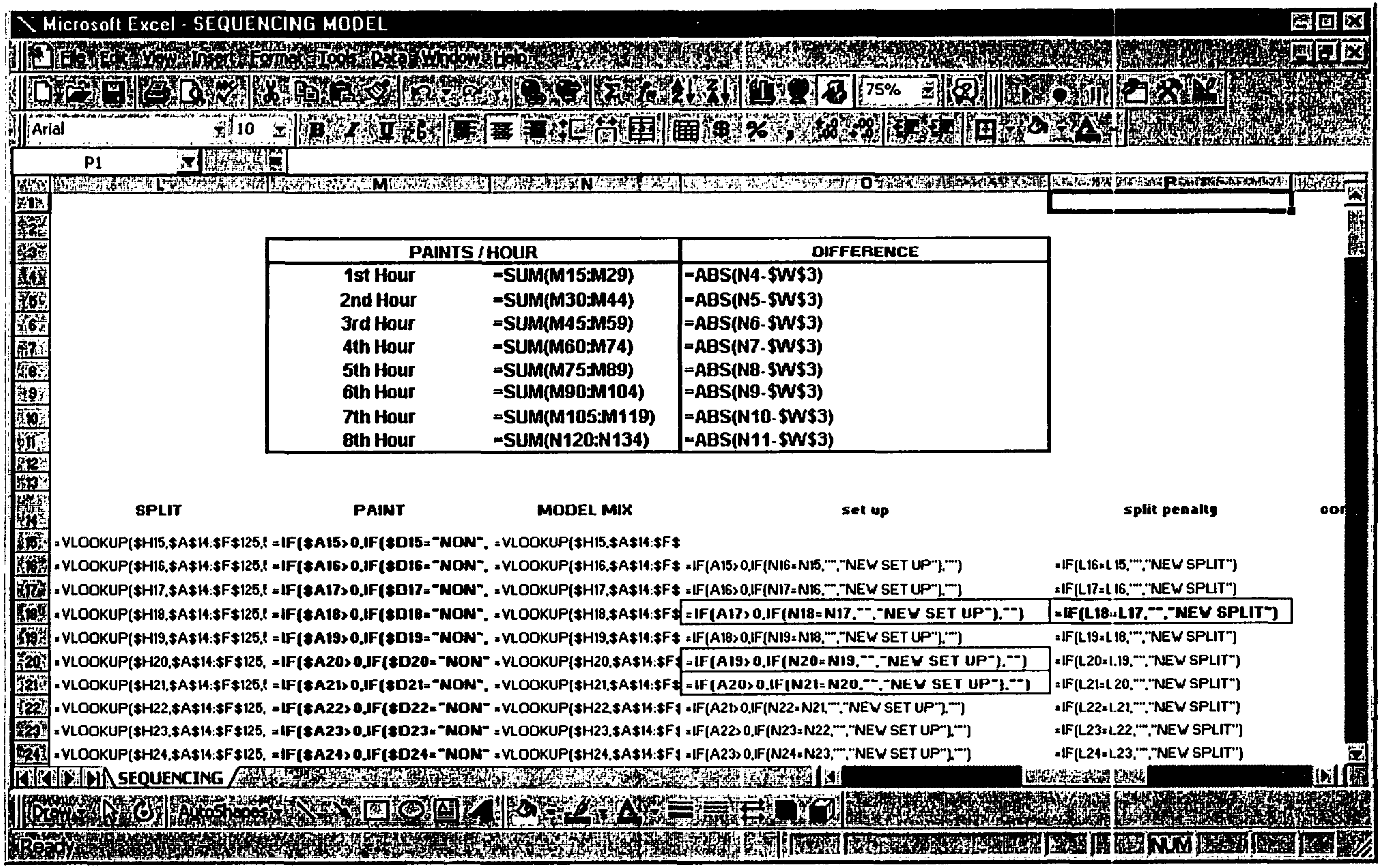


PAINTS / HOUR		DIFFERENCE
1st Hour	15	7
2nd Hour	3	5
3rd Hour	0	8
4th Hour	5	3
5th Hour	15	7
6th Hour	7	1
7th Hour	11	3
8th Hour	0	8

GROUP ENGINES 32697

SPLIT	PAINT	MODEL MIX	set up	split penalty	consec paint	group models				
								6081HH005	5	1157
								6081HH006	5	1157
							6081HRW03	1	1158	
2	1	6081					6081HRW10	1	1158	
2	1	6081				0	6081TRW01	4	1158	
2	1	6081				0	6081HH006	4	1159	
3	1	6105	NEW SET UP	NEW SPLIT		1	6081ACQ01	2	1160	
3	1	6105				0	6081HDW01	2	1161	
3	1	6125	NEW SET UP			1	6125HZ001	2	1162	
3	1	6081	NEW SET UP			1	6081TRW01	6	1163	
3	1	6081				0	6125HZ001	6	1164	
3	1	6081				1	6125HZ002	2	1164	
3	1	6081				0	6081AF001	2	1165	

Figure 29. Sequencing tab, computation of constraints.



	PAINTS / HOUR		DIFFERENCE
	1st Hour	=SUM(M15:M29)	=ABS(N4-$W$3)
	2nd Hour	=SUM(M30:M44)	=ABS(N5-$W$3)
	3rd Hour	=SUM(M45:M59)	=ABS(N6-$W$3)
	4th Hour	=SUM(M60:M74)	=ABS(N7-$W$3)
	5th Hour	=SUM(M75:M89)	=ABS(N8-$W$3)
	6th Hour	=SUM(M90:M104)	=ABS(N9-$W$3)
	7th Hour	=SUM(M105:M119)	=ABS(N10-$W$3)
	8th Hour	=SUM(N120:N134)	=ABS(N11-$W$3)

SPLIT	PAINT	MODEL MIX	set up	split penalty	co
=VLOOKUP($H15,$A$14:$F$125,	=IF($A15>0,IF($D15="NON",	=VLOOKUP($H15,$A$14:$F$			
=VLOOKUP($H16,$A$14:$F$125,	=IF($A16>0,IF($D16="NON",	=VLOOKUP($H16,$A$14:$F$	=IF(A15>0,IF(N16=N15,"","NEW SET UP"),"")	=IF(L16=L15,"","NEW SPLIT")	
=VLOOKUP($H17,$A$14:$F$125,	=IF($A17>0,IF($D17="NON",	=VLOOKUP($H17,$A$14:$F$	=IF(A16>0,IF(N17=N16,"","NEW SET UP"),"")	=IF(L17=L16,"","NEW SPLIT")	
=VLOOKUP($H18,$A$14:$F$125,	=IF($A18>0,IF($D18="NON",	=VLOOKUP($H18,$A$14:$F$	=IF(A17>0,IF(N18=N17,"","NEW SET UP"),"")	=IF(L18=L17,"","NEW SPLIT")	
=VLOOKUP($H19,$A$14:$F$125,	=IF($A19>0,IF($D19="NON",	=VLOOKUP($H19,$A$14:$F$	=IF(A18>0,IF(N19=N18,"","NEW SET UP"),"")	=IF(L19=L18,"","NEW SPLIT")	
=VLOOKUP($H20,$A$14:$F$125,	=IF($A20>0,IF($D20="NON"	=VLOOKUP($H20,$A$14:$F	=IF(A19>0,IF(N20=N19,"","NEW SET UP"),"")	=IF(L20=L19,"","NEW SPLIT")	
=VLOOKUP($H21,$A$14:$F$125,	=IF($A21>0,IF($D21="NON",	=VLOOKUP($H21,$A$14:$F$	=IF(A20>0,IF(N21=N20,"","NEW SET UP"),"")	=IF(L21=L20,"","NEW SPLIT")	
=VLOOKUP($H22,$A$14:$F$125,	=IF($A22>0,IF($D22="NON"	=VLOOKUP($H22,$A$14:$F	=IF(A21>0,IF(N22=N21,"","NEW SET UP"),"")	=IF(L22=L21,"","NEW SPLIT")	
=VLOOKUP($H23,$A$14:$F$125,	=IF($A23>0,IF($D23="NON"	=VLOOKUP($H23,$A$14:$F	=IF(A22>0,IF(N23=N22,"","NEW SET UP"),"")	=IF(L23=L22,"","NEW SPLIT")	
=VLOOKUP($H24,$A$14:$F$125,	=IF($A24>0,IF($D24="NON"	=VLOOKUP($H24,$A$14:$F	=IF(A23>0,IF(N24=N23,"","NEW SET UP"),"")	=IF(L24=L23,"","NEW SPLIT")	

Figure 30. Sequencing tab, computation of constraints, with formulas visible in the cells.

Site Selection

The site selected for this research was an engine manufacturing facility of a midwestern manufacturer of agriculture equipment, which has been employing the latest technology throughout the years. MMAE allocates more than 2% of its gross sales for research and development, indicating the company's commitment to innovation and its desire to stay ahead of its competition.

MMAE completed its first MRP installation in 1979 and has implemented parts of JIT since 1981 (Williams, 1986). By 1986, the company had implemented MRP in all its plants worldwide. JIT was first implemented within MMAE at a facility that produces hay and forage equipment for agricultural use. Considerable improvements, including a 58% reduction in inventories, were reported after implementing parts of the JIT system.

The engine manufacturing plant of MMAE has long been perceived as the focus factory throughout the organization. It was the second plant within MMAE to achieve the ISO 9000 certification. This facility employs traditional (MRP) and contemporary (JIT) manufacturing systems, a condition that serves the purpose of the present research.

The design and development of EMP was initiated in 1973. This facility has 915,000 square feet, 340,000 of which is allocated to the assembly area. EMP began production of diesel engines in February, 1976. The number of engines produced in 1995 was 29,500 including marine, natural gas, and diesel. This volume is made up of 400 series (7.6 and 8.1 liter) and 500 series (10.1 liter) engines. The engines produced at this facility are shipped to internal customers (MMAE agricultural and industrial divisions) and to numerous original equipment manufacturers (OEM). The share of OEM production has grown from 3% of volume in 1976 to 15% in 1995 and is expected to reach 50% of volume by the year 2005.

EMP provides purchased and manufactured service parts for the engines built at this facility. The service performance level is measured in the following two ways:

- Fill out of the factory to the Parts Distribution Center (PDC)
- Fill from PDC to dealers

The management goal is to fill 100% of all orders from the factory to PDC and 97% from PDC to dealers each month. EMP currently is filling orders from the factory to PDC at 93% and from PDC to dealers at 98%.

Software Selection

Intense reliance on the legacy computer systems has been one of the concerns of MMAE. EMP also relies heavily on legacy computer systems for production planning and control. Many MMAE facilities have begun using Microsoft Excel as a production-planning tool. This usage was a factor in selecting Excel for the research model.

In the new information-driven economy, selecting software to help achieve organizational goals has become more complex than ever before. The selection of Evolver as an optimization tool was based on its price and availability through MMAE. Evolver, an optimization add-on for Microsoft Excel, uses genetic algorithms to solve complex optimization problems in such areas as finance, distribution, scheduling, resource allocation, manufacturing, budgeting, and engineering. Virtually any type of problem that can be modeled in Excel can be solved by Evolver, including previously unsolvable problems. Evolver, which requires no knowledge of programming or genetic algorithm theory, is available in three versions: standard, professional, and industrial. The professional and industrial versions have increased problem capacities and advanced features, including the Evolver Developer's Kit. As noted in the literature review, genetic algorithms are becoming prevalent as an optimization tool for scheduling problems. Many software vendors offer genetic algorithm-based optimization software, but Evolver by Palaside Inc. was one of the first in the market.

Data Collection

The master scheduler plans production (via Excel) for the fiscal year in monthly time-buckets. Production for each three-month period (current and following two months) is planned in weekly buckets. The master scheduler gives the production in weekly buckets in Excel workbook to the scheduler, who is responsible for the engine line-up for the next 20 days. The scheduler performs the line-up in daily buckets for the next 20 days in the HOST system.

Customer orders are kept in the legacy computer system called Common Worldwide Interfactory System (CWIS). These orders are auto-downloaded into the MRP master schedule. All customers have offset days within the master scheduling process. An offset is the number of production days between the launch and the ship on the assembly line. MRP generates the master schedule in monthly buckets after considering the customer requirement date and number of offset days. Monthly buckets are broken down in weekly buckets when the master scheduler runs a program in the HOST MRP.

The purpose of this research was to develop and evaluate a model that will generate an improved engine schedule and sequence based on CM when compared with the current method. The actual line-up schedule and sequence that were used to build engines for the 100 production days between summer of 1999 and spring of 2000 at EMP were used for the comparison. These data were used in the simulation for the current scheduling and sequencing method (control condition), as well as for the proposed scheduling and sequencing model for optimization (experimental condition). After the scheduling and sequencing optimizations were performed, the results of these optimizations were used in simulation.

In the proposed model, the master scheduler would perform the engine line-up in Excel using the optimization tool Evolver. This line-up would be auto-downloaded in the HOST system. The model is intended to provide EMP's management with the ability to perform what-if analysis in a timely manner.

Statistical Analysis

After the output from the simulation run for both methods, current and proposed, was obtained, statistical analysis was performed. Various statistical tools were used to perform the analysis. The five variables compared and analyzed were as follows:

1. Cycle time of engines for the final assembly line and four downstream processes
2. Queue size in front of four downstream processes after final assembly line
3. Utilization of work centers in the final assembly line and four downstream processes

4. Flow rate of engines through the final assembly line and four downstream processes

5. Total output of engines through final assembly line and four downstream processes

Expected improvements in the five variables of the proposed scheduling and sequencing model are as follows:

1. Reduction in cycle time of engines for the final assembly line and four downstream processes (smaller number is better)

2. Reduction in queue size in front of four downstream processes after final assembly line (smaller number is better)

3. Increase in the utilization percentage of work centers in the final assembly line and four downstream processes (larger number is better)

4. Even flow rate of engines through the final assembly line and four downstream processes

5. Increase in total output of engines through final assembly line and four downstream processes (larger number is better)

Some analysis was performed as part of the simulation output, such as determining minimum and maximum values and total output of engines, but the majority of the analysis was done after assembling the simulation output from both methods, current and proposed. A sample output from the model was used to determine that the data were normally distributed. The statistical tools used to analyze the data included the following: arithmetic averages, minimum and maximum values for each dependent variable, standard deviation, percentage of utilization of work centers, and t-tests.

Model Validation

According to a key expert in GPSS/H and PROOF simulation modeling at the corporate office of MMAE (G. Rehn, personal communication, [e-mail], December 22, 1999), simulations at MMAE have proved highly valid although the number of validations of simulations has been limited. Two formal validations in the 1980s and one informal in the early 1990s have been made. A validation of a simulation of one of MMEA's plant that manufacture cotton pickers for its 2X conveyor system in the early 1980s found that in areas primarily equipment oriented, the correlation between the method in use and the simulated method was high (98%) but in the manpower-related instances, the confidence level was in the low 90s.

In 1988 a formal model validation was done for a simulation for the AGV assembly system in conjunction with the test acceptance. A statistician concluded that there was no significant difference between the simulation model and the behavior of the actual system. He recommended that the model be used to predict the effectiveness of future systems because it was quicker and easier to identify tendencies with the model.

In the validation performed in the early 1990s, a simulation model was compared with actual output in order to demonstrate the value of Optimax software. A month's actual line-up at a seeding plant was used as an input for the simulated model. The actual output and the simulated output were so close that no statistical analysis was performed.

Thus in a limited number of cases, model have proved to be highly valid at MMAE. It should also be noted that the key expert at the corporate office of MMAE was consulted whenever questions arose regarding the design and testing of the model.

The proposed model is scheduled to be implemented at the EMP's final assembly line in the spring of 2000. Due to the time constraints for this research, model validation was conducted through computer simulation, using the software GPSS/H and PROOF, products of the Wolverine Software Corporation in Annandale, Virginia. GPSS/H is a simulation language, and PROOF is an animation software used within Excel file format. Excel serves as a user interface to the line-up model. It contains the launch sequence, shipping schedule, initial inventory, process cycle times, operating schedule by department (number of shifts in operation, etc.), number of operators/shift, and some equipment parameters such as number of load bars in the system. All these items are data-driven variables or inputs to the model. The parameters, once specified, define a specific simulation scenario to be tested through the model. An Excel macro that captures all the data defined in the Excel and creates various text files in a specific format understood by the simulation code was used.

GPSS/H, a simulation language, was used to write a model of the line-up alternatives. The simulation code accounts for all the resources, capacities, and process logic of the system. The model reads in all the data provided by the Excel interface and uses those conditions to execute all the "process" rules defined in the simulation code that represents the process flow of engines from the final assembly line to ship. At the end of the simulation run, the model generates output reports describing production volumes attained, operator utilization, equipment utilization, inventory levels, and total process cycle time, which is a function of the all the individual process cycle times and the dynamic delays associated with resource availability. The model also "writes" the graphic commands to a file to drive an animation depiction of the simulation test.

PROOF, the animation software, post-processes the graphic commands written by the simulation model. The result depicts the flow of the processes and illustrates the overall flow of the system. The animation first highlights any process issues and promotes understanding of the overall system. The related GPSS/H output then serves to quantify the performance. PROOF can also be used for some of the input data to the simulation, most often to show the configuration of the layout being tested. PROOF can translate DXF file formats from CAD programs and use them in the animation. Many of the layout capacities and conveyor speeds and times come from the layout of the system, once it has been translated into PROOF.

An output file in plain text format is created each time a simulation run is performed and the outcome is illustrated in the output file. A copy of the output is attached in Appendix A.

Summary

This research was designed to identify production planning and control (PP&C) constraints at EMP and to develop and validate scheduling and sequencing model based on these constraints. The site for the research was an engine manufacturing plant of a midwestern manufacturer of agriculture equipment. The plant employs both traditional and contemporary manufacturing systems.

The independent variable in the research design is the method of scheduling and sequencing, the experimental condition being the proposed model and the control condition, the current scheduling and sequencing method. Dependent variables are cycle time, queue time, utilization of work centers, flow of engines, and total output of engines.

The software selected for the research model was Excel, with Evolver as an optimization tool.

A two-part model, based on constraints management philosophy of production planning and control methods, was developed by the researcher in Excel, one part for scheduling and the other for sequencing. Using data from the 100 production days during the fall of 1999 and the spring of 2000, simulations for the current scheduling and sequencing method and for the proposed model were compared. Output from the simulations for the experimental and control conditions was statistically analyzed, using arithmetic averages, minimum and maximum. values for each dependent variable, standard deviation, percentage of utilization of work centers, and t-tests.

CHAPTER IV

SIMULATION RESULTS AND DISCUSSION

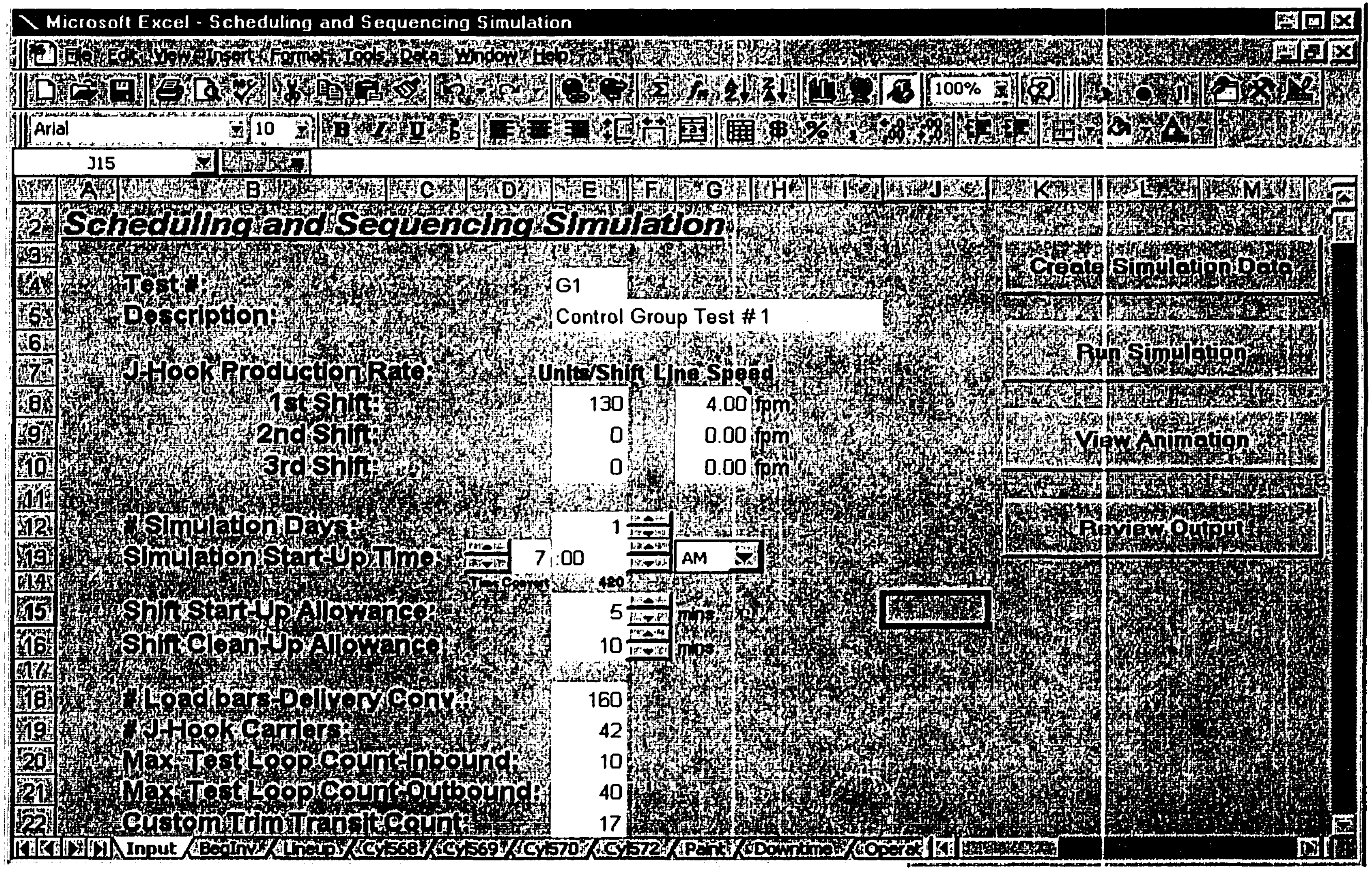
As stated earlier, the purpose of this research was to develop and evaluate a model that would generate an improved engine schedule and sequence based on constraint management (CM) in comparison to the currently used method. The actual lineup schedule and sequence that were used to build engines for the 100 production days between summer of 1999 and spring of 2000 at EMP were used for the comparison. Dates for the data were selected after review by the key expert in the area of production planning and control at EMP (D. Eck, personal communication, April 24, 2000). The actual dates for the data used in this study are listed in Table 3. These data were used in the simulation for the current scheduling and sequencing method (control condition), as well as for the proposed scheduling and sequencing model for optimization (experimental condition).

The simulation was developed by the key expert in GPSS/H and PROOF simulation modeling at the corporate office of MMAE. GPSS/H is a simulation language, and PROOF is an animation software used within Excel file format. Excel serves as a user interface to the lineup model. It contains the launch sequence, shipping schedule, initial inventory, process cycle times, operating schedule by department (number of shifts in operation, etc.), number of operators/shift, and some equipment parameters such as number of load bars in the system (see Figures 31 and 32). All these items are data-driven variables or inputs to the model. After specifying the parameters, each simulation run was conducted with a specific simulation number. All the parameters maintained the same values for the 200 simulation runs. The only values

Table 3

Line-up Dates and Test Numbers

Line-up date	Test no.	Line-up date	Test no.	Line-up date	Test no.
6/21/1999	1	9/15/1999	35	11/23/1999	68
6/22/1999	2	9/16/1999	36	11/24/1999	69
6/23/1999	3	9/17/1999	37	11/25/1999	70
6/24/1999	4	9/20/1999	38	11/30/1999	71
6/25/1999	5	9/21/1999	39	12/1/1999	72
6/28/1999	6	9/22/1999	40	12/2/1999	73
6/29/1999	7	9/23/1999	41	12/3/1999	74
6/30/1999	8	10/4/1999	42	12/6/1999	75
7/1/1999	9	10/5/1999	43	12/7/1999	76
7/2/1999	10	10/6/1999	44	12/8/1999	77
7/6/1999	11	10/7/1999	45	12/9/1999	78
7/7/1999	12	10/8/1999	46	12/10/1999	79
8/9/1999	13	10/11/1999	47	12/13/1999	80
8/10/1999	14	10/12/1999	48	12/14/1999	81
8/11/1999	15	10/13/1999	49	12/15/1999	82
8/13/1999	16	10/14/1999	50	12/16/1999	83
8/16/1999	17	10/15/1999	51	12/17/1999	84
8/17/1999	18	11/1/1999	52	12/20/1999	85
8/18/1999	19	11/2/1999	53	12/21/1999	86
8/19/1999	20	11/3/1999	54	12/22/1999	87
8/20/1999	21	11/4/1999	55	1/6/2000	88
8/23/1999	22	11/5/1999	56	1/7/2000	89
8/24/1999	23	11/8/1999	57	1/10/2000	90
8/30/1999	24	11/9/1999	58	1/11/2000	91
8/31/1999	25	11/10/1999	59	1/12/2000	92
9/1/1999	26	11/11/1999	60	1/13/2000	93
9/2/1999	27	11/12/1999	61	1/14/2000	94
9/3/1999	28	11/15/1999	62	1/18/2000	95
9/7/1999	29	11/16/1999	63	1/19/2000	96
9/8/1999	30	11/17/1999	64	1/20/2000	97
9/9/1999	31	11/18/1999	65	1/21/2000	98
9/10/1999	32	11/19/1999	66	1/24/2000	99
9/13/1999	33	11/22/1999	67	1/25/2000	100
9/14/1999	34				



Scheduling and Sequencing Simulation

Test #: G1

Description: Control Group Test # 1

J-Hook Production Rate: Units/Shift Line Speed

1st Shift: 130 4.00 fpm

2nd Shift: 0 0.00 fpm

3rd Shift: 0 0.00 fpm

Simulation Days: 1

Simulation Start-Up Time: 7 :00 AM

Time Convert 420

Shift Start-Up Allowance: 5 mins

Shift Clean-Up Allowance: 10 mins

Load bars-Delivery Conv.: 160

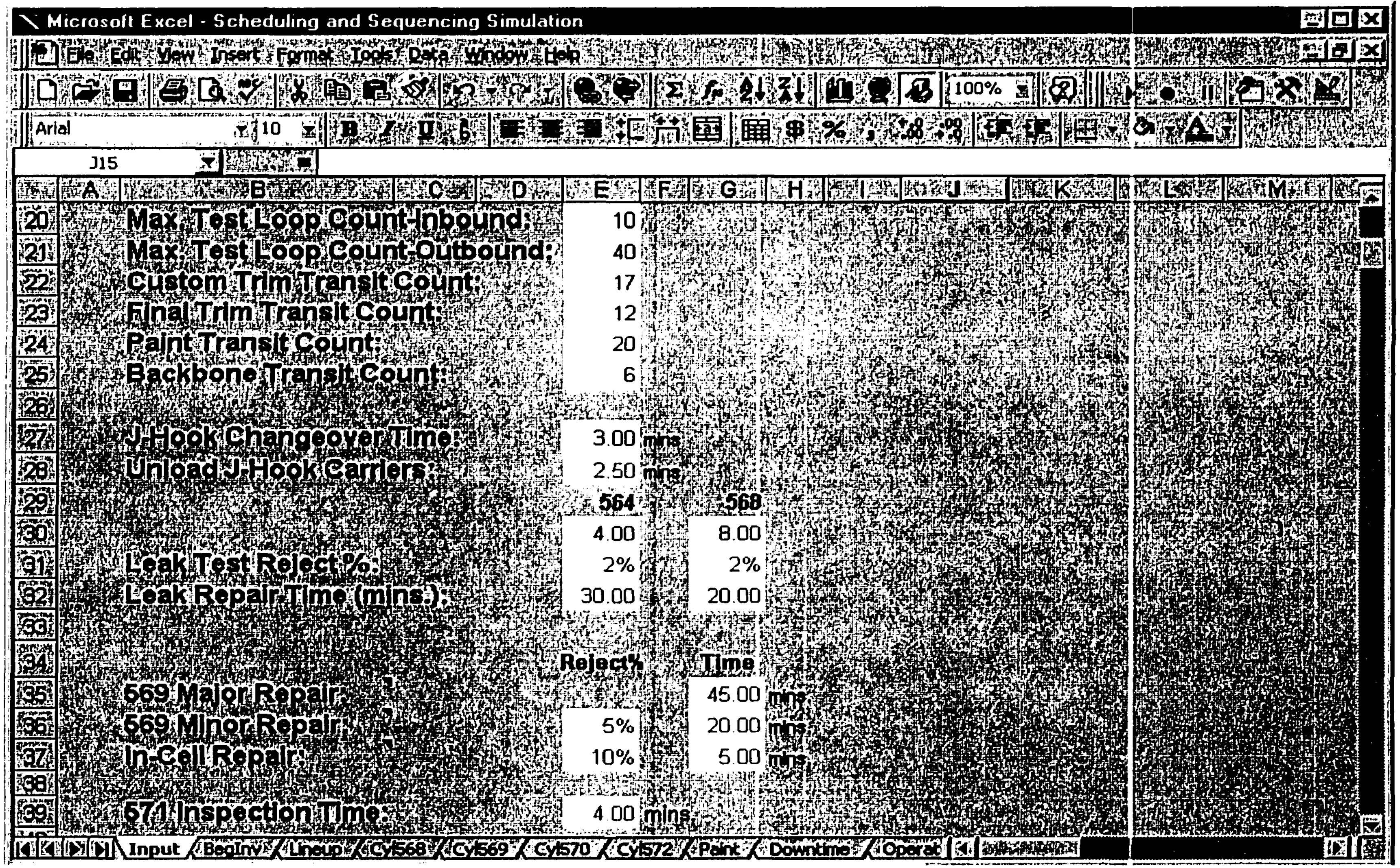
J-Hook Carriers: 42

Max. Test Loop Count-Inbound: 10

Max. Test Loop Count-Outbound: 40

Custom Trim Transit Count: 17

Figure 31. Excel interface for the simulation run, top section.



	A	B	C	D	E	F	G	H
20		Max. Test Loop Count-Inbound:			10			
21		Max. Test Loop Count-Outbound:			40			
22		Custom Trim Transit Count:			17			
23		Final Trim Transit Count:			12			
24		Paint Transit Count:			20			
25		Backbone Transit Count:			6			
26								
27		J-Hook Changeover Time:			3.00	mins		
28		Unload J-Hook Carriers:			2.50	mins		
29					564		568	
30					4.00		8.00	
31		Leak Test Reject %:			2%		2%	
32		Leak Repair Time (mins.):			30.00		20.00	
33								
34					Reject%		Time	
35		569 Major Repair:					45.00	mins
36		569 Minor Repair:			5%		20.00	mins
37		In-Cell Repair:			10%		5.00	mins
38								
39		571 Inspection Time:			4.00	mins		

Figure 32. Excel interface for the simulation run, bottom section.

that changed were the lineup sequences. An Excel macro captured all the data defined in the Excel interface and created various text files in a specific format understood by the simulation code.

GPSS/H was used to write a model of the lineup alternatives. The simulation code (see Appendix B) accounts for all the resources, capacities, and process logic of the system. The model reads in all the data provided by the Excel interface and uses those conditions to execute all the "process" rules defined in the simulation code that represents the process flow of engines from the final assembly line to ship. At the end of the simulation run, the model generates output reports describing production volumes attained, operator utilization, equipment utilization, inventory levels, and total process cycle time, which is a function of all the individual process cycle times and the dynamic delays associated with resource availability. The model also "writes" the graphic commands to a file to drive an animation depiction of the simulation test. (See Appendix C for a snapshot of animation depiction of simulation run.)

PROOF post-processes the graphic commands written by the simulation model. The result depicts the flow of the processes and illustrates the overall flow of the system. The animation first highlights any process issues and promotes understanding of the overall system. The related GPSS/H output then serves to quantify the performance. PROOF can also be used for some of the input data to the simulation, most often to show the configuration of the layout being tested. PROOF can translate DXF file formats from CAD programs and use them in the animation. Many of the layout capacities and conveyor speeds and times come from the layout of the system, once it has been translated into PROOF.

An output file in plain text format is created each time a simulation run is performed, and the outcome is illustrated in the output file. A copy of the output appears in Appendix A.

The research questions stated in chapter I were the bases for this experimental study. These questions are reiterated below for quick reference.

1. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the cycle time for the final assembly line and four downstream processes at an engine manufacturing plant (EMP) of a midwestern manufacturer of agricultural equipment (MMAE)?

2. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the queue size for the final assembly line and four downstream processes at EMP?

3. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the utilization of work centers in the final assembly line and four downstream processes at EMP?

4. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the flow rate of engines through the final assembly line and four downstream processes at EMP?

5. What is the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on the total output of engines through the final assembly line and four downstream processes at EMP?

Various statistical tools were used to analyze the output from the simulation run for both methods, current and proposed. The five variables compared and analyzed were as follows:

1. Cycle time of engines for the final assembly line and four downstream processes
2. Queue size in front of four downstream processes after final assembly line
3. Utilization of work centers in the final assembly line and four downstream processes
4. Flow rate of engines through the final assembly line and four downstream processes
5. Total output of engines through the final assembly line and four downstream processes

Expected improvements in the five variables of the proposed scheduling and sequencing model were as follows:

1. Reduction in cycle time of engines for the final assembly line and four downstream processes (smaller number is better)
2. Reduction in queue size in front of four downstream processes after final assembly line (smaller number is better)
3. Increase in the utilization percentage of work centers in the final assembly line and four downstream processes (larger number is better)
4. Even flow rate of engines through the final assembly line and four downstream processes

5. Increase in total output of engines through final assembly line and four downstream processes (larger number is better)

Cycle Time

The results of the simulations indicated very little reduction in average cycle time after 100 runs for the control condition and 100 simulation runs for the experimental condition (see Figure 33 for a comparison of each condition's cycle time for the 100 simulation runs.) The average cycle time for the control condition was 9.04 hours with a standard deviation of 1.14 and average cycle time for the experimental condition was 8.97 hours with a standard deviation of 1.01. Results of t-test indicated the following values: t-value = 1.24, df = 99, and two-tailed significance = .219. Thus, the difference between the control condition and the experimental condition results was not statistically significant, with an alpha level of .05.

A smaller standard deviation value for the experimental condition indicates that there is less variation in cycle time. In the manufacturing environment, less variability is better. One reason for a less-than-expected reduction in cycle time could be the increased production of painted engines for the experimental condition, which requires additional processes. (See Figure 33, which shows a spike for Test 51, a day when all engines built were painted.) Cycle time was reduced for 48 out of 100 days for the experimental condition versus 39 days for the control condition; for 13 days, cycle times were identical for both conditions (see Table 4).

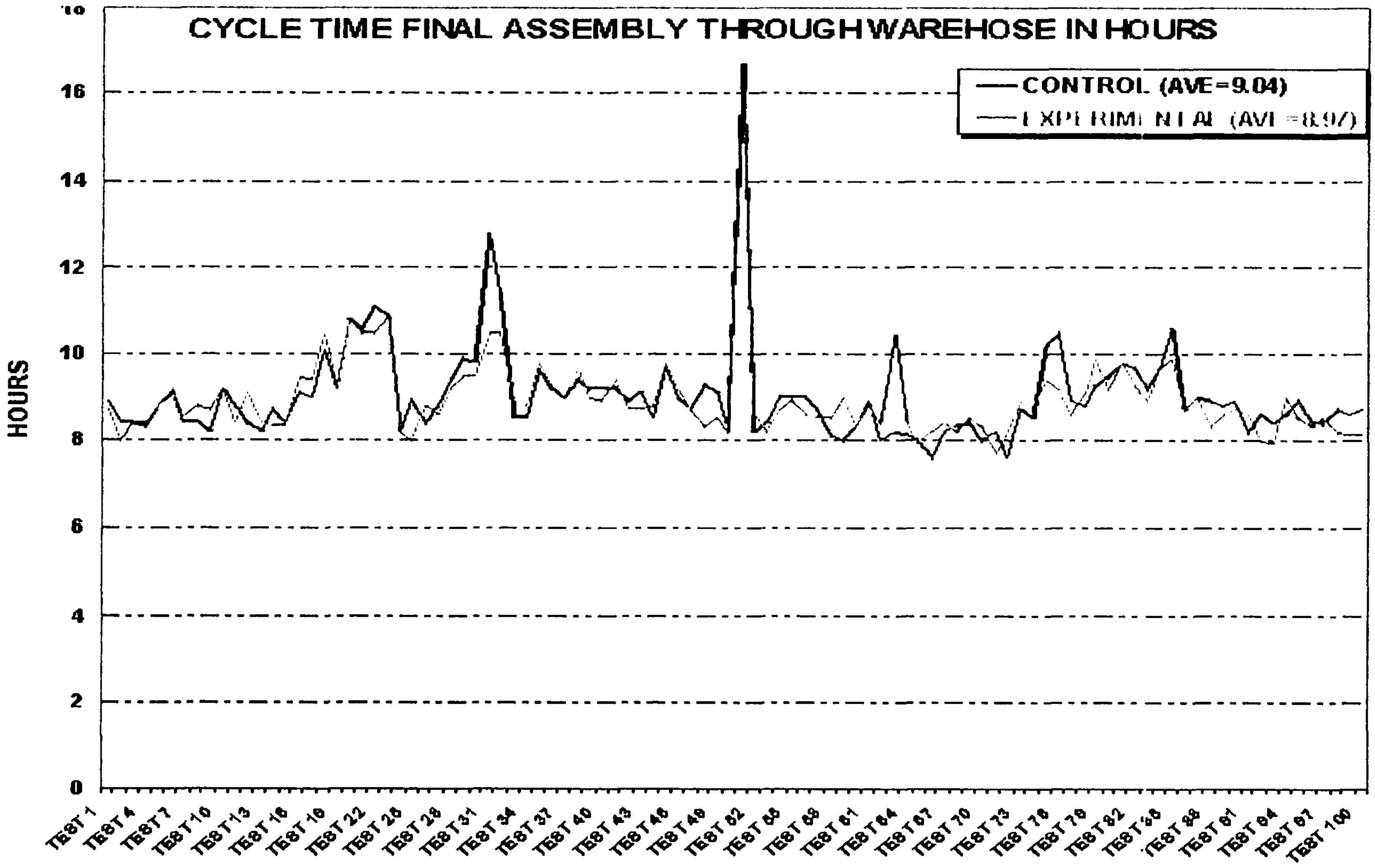


Figure 33. Cycle time final assembly through warehouse in hours. The spike for Test 51 is due to the fact that on that particular day, all of the engines built were painted.

Table 4

Comparison Data for Cycle Time

Measures	Control	Experimental
Average (minutes)	9.04	8.97
SD	1.14	1.01
No. of days of reduced cycle time	39.00	48.00

Queue Size

The results of the simulations indicated very little reduction in queue size after 100 runs for the control condition and 100 simulation runs for the experimental condition. (See Figure 34 for a comparison of the queue size of each condition for the 100 simulation runs.) The average queue size for the control condition was 110.27 engines with a standard deviation of 2.45, and the average queue size for the experimental condition was 110.12 engines with a standard deviation of 2.29. Results of t-test indicated the following values: t-value = 0.54, df = 99, and two-tailed significance = .588. Since the value of two-tailed significance was greater than .05, the difference between results for the control and the experimental conditions was not statistically significant with an alpha level of .05.

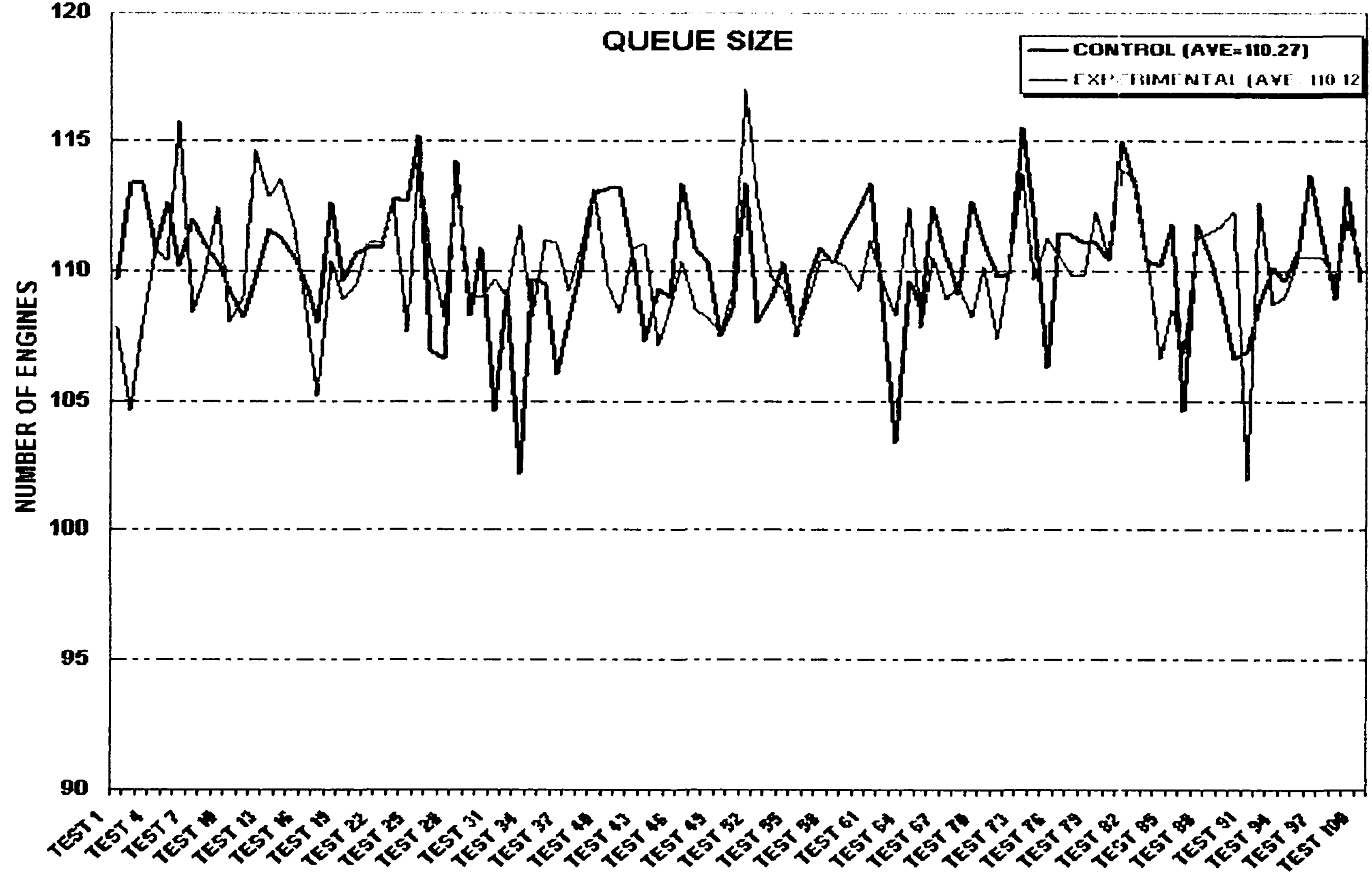


Figure 34. Queue size.

Again a slightly smaller standard deviation value for the experimental condition indicates less variability in the system. Performance in the control condition was better for 44 days and in the experimental condition on 53 days; for the remaining 3 of 100 days, both performed the same (see Table 5). Improvements in queue sizes were observed during the simulation runs for the experimental condition. For the control condition, several times there was "feast or famine" in the queues during daily runs, but data were collected only for average queue sizes. The experimental condition demonstrated uniform queue size throughout the daily simulation runs. A uniform queue size throughout the day is preferred over a queue of wide variability.

Table 5

Comparison Data for Queue Size

Measures	Control	Experimental
Average	110.27	110.12
SD	2.45	2.29
No. of days of reduced queue size	44.00	53.00

Utilization of Work Centers

The results of the simulations indicated improvement in utilization of work centers after 100 runs for the control condition and 100 simulation runs for the experimental condition. (See Figure 35 for a comparison of the control condition and the experimental condition for utilization of work centers for the 100 simulation runs.) The average utilization for the control condition was 41.33% with a standard deviation of 4.22, and the average utilization for the experimental condition was 42.25% with a standard deviation of 3.95. Results of the t-test indicated the following values: t-value = 3.72, df = 99, and two-tailed significance = .000. The difference between results for the control and the experimental conditions was statistically significant, with an alpha level of .05.

The utilization of work centers of test cells, custom trim, paint, and final trim was recorded and measured. Since the final assembly line was a computer controlled line, utilization of work centers was not recorded. Various operators were assigned to more than one work center, but measurements were recorded for the utilization of centers not for the utilization of operators. Total utilization for the four downstream processes of the experimental condition was increased by 2.23%. Utilization of work centers in the four downstream processes for the control condition is presented in Table 6 and for the experimental condition in Table 7.

Performance in the control condition was better than that in the experimental condition on 35 days, and performance for the experimental condition was better on 64 days; for the remaining day, both performed the same (see Table 8).

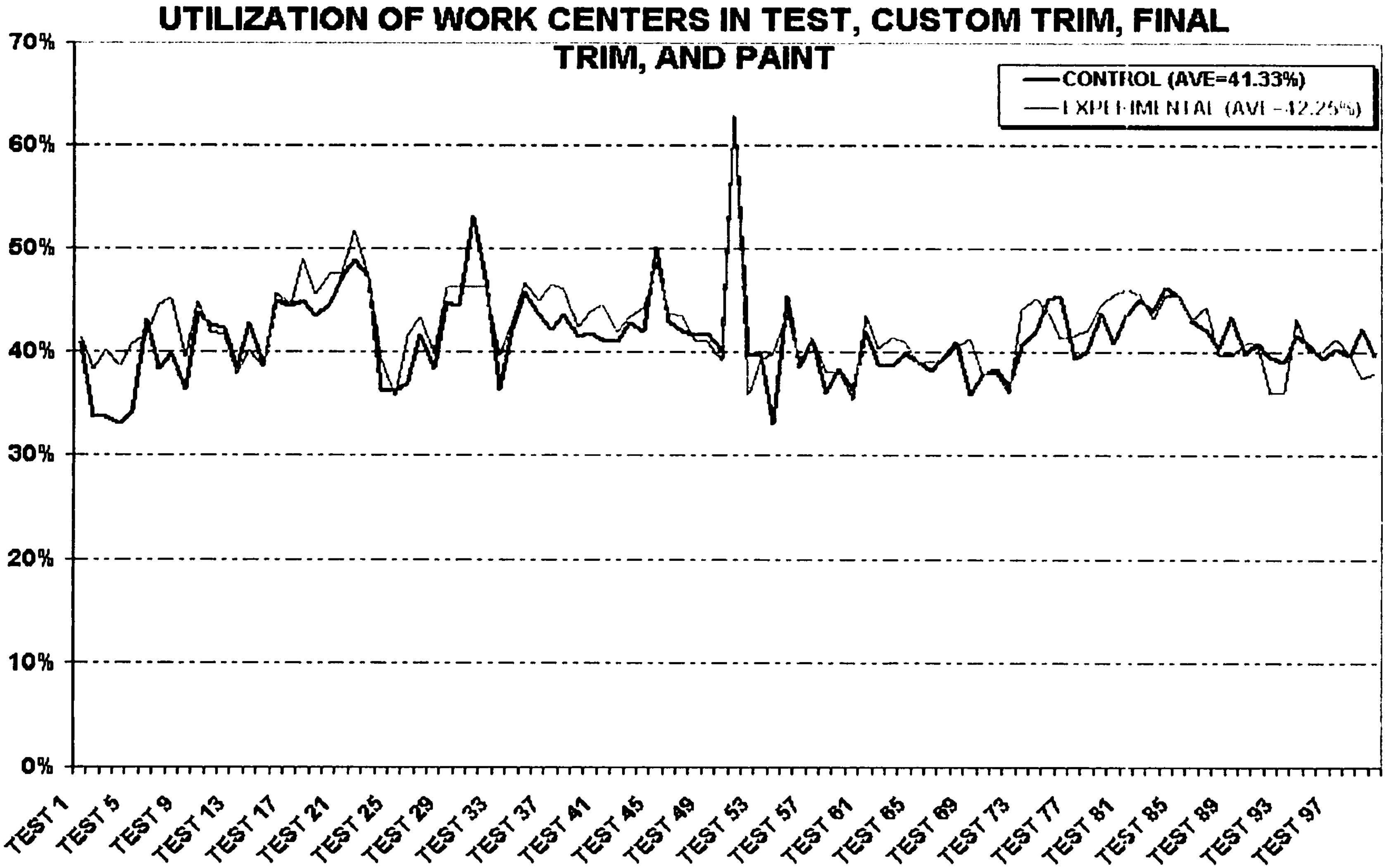


Figure 35. Utilization of work centers in test cells, custom trim, final trim, and paint. As noted the spike for Test 51 is due to the fact that on that particular day, all of the engines built were painted

Table 6

Utilization of Work Centers for the Control Condition

Work centers	Average (%)	SD (%)
Test cells	50.23	3.04
Custom trim	33.43	8.04
Final trim	39.67	2.95
Paint	**42.00**	**8.42**
TOTAL	41.33	4.22

Table 7

Utilization of Work Centers for the Experimental Condition

Work centers	Average (%)	SD (%)
Test cells	50.58	2.55
Custom trim	34.44	7.75
Final trim	40.18	3.03
Paint	**43.80**	**7.75**
TOTAL	42.25	3.95

Table 8

Comparison Data for Utilization of Work Centers

Measures	Control	Experimental
Average	41.33%	42.25%
SD	4.22	3.95
No. of days of increased total utiliz.	35.00	64.00

Because paint was thought to be the constraint of the system, the results for paint utilization are discussed separately. (See Figure 36 for a comparison of the control condition and the experimental condition for utilization of work centers in paint.) Paint utilization increased for the experimental condition, as expected, but the increase was not statistically significant. The average utilization of work centers in paint for the control and the experimental conditions was 41.99% and 43.80%, respectively. Performance in the control condition was better than that in the experimental condition on 31 days, and performance for the experimental condition was better on 68 days; for the remaining day, both performed the same (see Table 9).

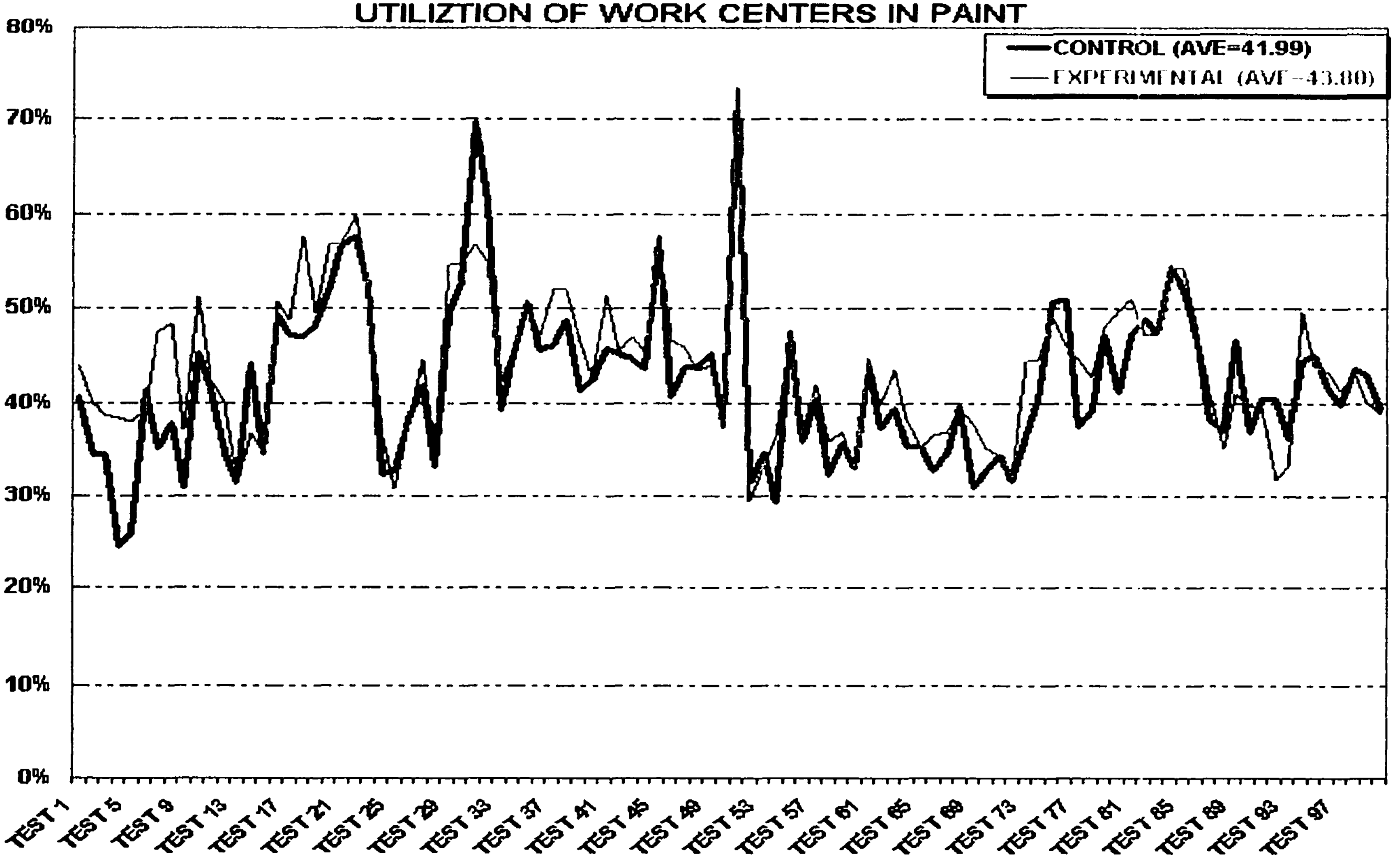


Figure 36. Utilization of work centers in paint. Results from Test 51 were atypical because all the engines built on that day were painted, a process requiring more time.

Table 9

Comparison Data for Paint Utilization

Measures	Control	Experimental
Average	41.99%	43.80%
SD	8.42	7.75
No. of days of increased paint utiliz.	31.00	68.00

Flow Rate of Engines

A better, more even flow of engines through the final assembly line (j-hook) and four downstream processes (test cells, custom trim, final trim, and paint) was the anticipated improvement for the experimental condition, but this was not achieved. Tables 10 and 11 present average and standard deviations of flow rates of engines in minutes for the control and the experimental conditions, respectively. Because paint was considered to be the constraint of the system, special attention was paid to this operation's flow rate. However, data gathered from both groups indicated that the custom trim operation is the constraint. For the control condition simulation run, it took 16.13 minutes to process an engine in custom trim versus 15.90 minutes in paint. For the experimental condition simulation run the data indicated similar results, 15.56 minutes for each engine in the custom trim operation versus 15.31 minutes in paint. Even though the difference in minutes between custom trim and paint was very minimal, it was surprising nonetheless to find out that another operation might become the bottleneck.

Table 10

Flow Rate of Engines for the Control Condition (Minutes/Engine)

Processes	Average	SD
J-hook	4.21	0.23
Test cells	11.26	0.66
Custom trim	16.13	3.45
Final trim	10.85	1.87
Paint	**15.90**	**3.73**
TOTAL	58.35	

Table 11

Flow Rate of Engines for the Experimental Condition (Minutes/Engine)

Processes	Average	SD
J-hook	4.18	0.10
Test cells	11.66	4.78
Custom trim	15.56	3.11
Final trim	10.47	0.93
Paint	**15.31**	**3.05**
TOTAL	57.18	

The average standard deviation of flow rate of engines in minutes for the five processes for the control and experimental conditions was 2.60 and 2.59, respectively. The number of days performance in the control condition was better than that in the experimental condition were 49, and the number of days performance in the experimental condition was better was 50; for the remaining day, performance was the same for both conditions (see Table 12).

For each condition, custom trim and paint, both of which were more time consuming than other operations, were reduced in cycle times, thereby evening the flow. The experimental condition demonstrated a reduction of 3.50% for custom trim and 3.70% for the paint operation. The experimental condition also demonstrated a reduction of 2.00% in total flow minutes versus the control condition flow minutes, but the goal to have a better flow for the five processes was not achieved.

Table 12

Comparison Data for Even Flow

Measure	Control	Experimental
Average	2.60	2.59
SD	0.23	0.22
No. of days of better even flow	49.00	50.00

Total Output of Engines

Simulation results indicated an increase in the total number of engines processed in the system after 100 runs for the control condition and 100 simulation runs for the experimental condition (see Figure 37). The average number of engines processed each day in the final assembly line and four downstream processes was 467.27 for the control condition with a standard deviation of 49.43. The comparative figure for the experimental condition was 478.07 engines with a standard deviation of 41.92. The smaller standard deviation number for the experimental condition indicates less variability compared with the control condition. The number of days performance in the control condition was better than in the experimental condition was 37, and the number of days performance in the experimental condition was better was 62; for the remaining day, performance was the same for both conditions (see Table 13). Results of the t-test indicated the following values: t-value = 3.18, df = 99, and two-tailed significance = .002, with an alpha level of .05. Thus, the difference between the control condition and the experimental condition results was statistically significant. Once again, it should be noted that the data from Test 51, a day when all engines built were painted, a process requiring more time, were atypical.

On average the total number of engines processed in the system increased by 10.8 per day in the experimental condition. The experimental condition produced more engines on 62 out of 100 days, versus 37 days for the control condition. One-day total output was the same for both conditions. Averages with standard deviations for the final assembly line (j-hook) and the four downstream processes are presented in Tables 14 and 15 for the control and experimental conditions, respectively.

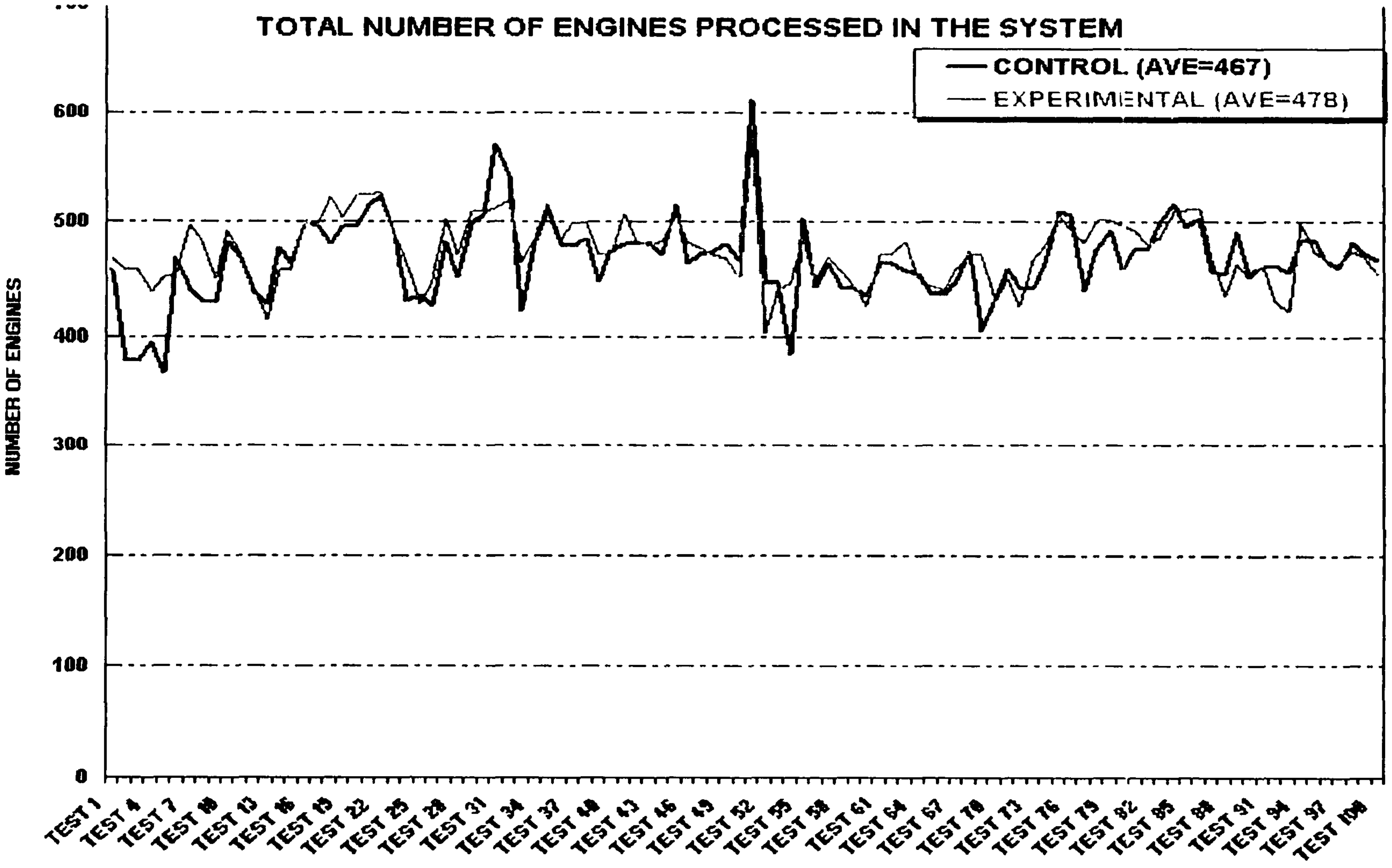


Figure 37. Total engine processed in the system. The spike for Test 51 represents the atypical situation of all engines built that day being painted.

Table 13

Comparison Data for Total Output

Measures	Control	Experimental
Average	467.27	478.07
SD	49.43	41.91
No. of days of increased total output	37.00	62.00

Because paint was thought to be the bottleneck of the system, special attention was paid to this operation. The average number of engines painted for the control condition was 71.88 with a standard deviation of 15.58, and the average number of engines painted for the experimental condition was 74.92 engines with a standard deviation of 13.99. Results of the t-test indicated the following values: t-value = 4.03, df = 99, and two-tailed significance = .000 with an alpha level of .05. Thus the difference between the results for the control and the experimental conditions was statistically significant.

Paint output was increased by 3.04 units or 4.23%. On average, more engines were painted in the experimental condition, on 61 out of 100 days, versus 18 days for the control condition. On 21 days, output was the same for both groups. (See Figure 38 for a comparison of paint production in each condition for the 100 simulation runs.)

Table 14

Number of Engines Processed in the System (Control Condition)

Processes	Average	SD
J-hook	105.04	4.69
Test cells	110.83	6.32
Custom trim	72.58	15.54
Final trim	106.94	7.31
Paint	**71.88**	**15.58**
TOTAL	467.27	49.43

Table 15

Number of Engines Processed in the System (Experimental Condition)

Processes	Average	SD
J-hook	105.55	2.39
Test cells	114.41	5.31
Custom trim	74.86	15.02
Final trim	108.33	5.21
Paint	**74.92**	**13.99**
TOTAL	478.07	41.92

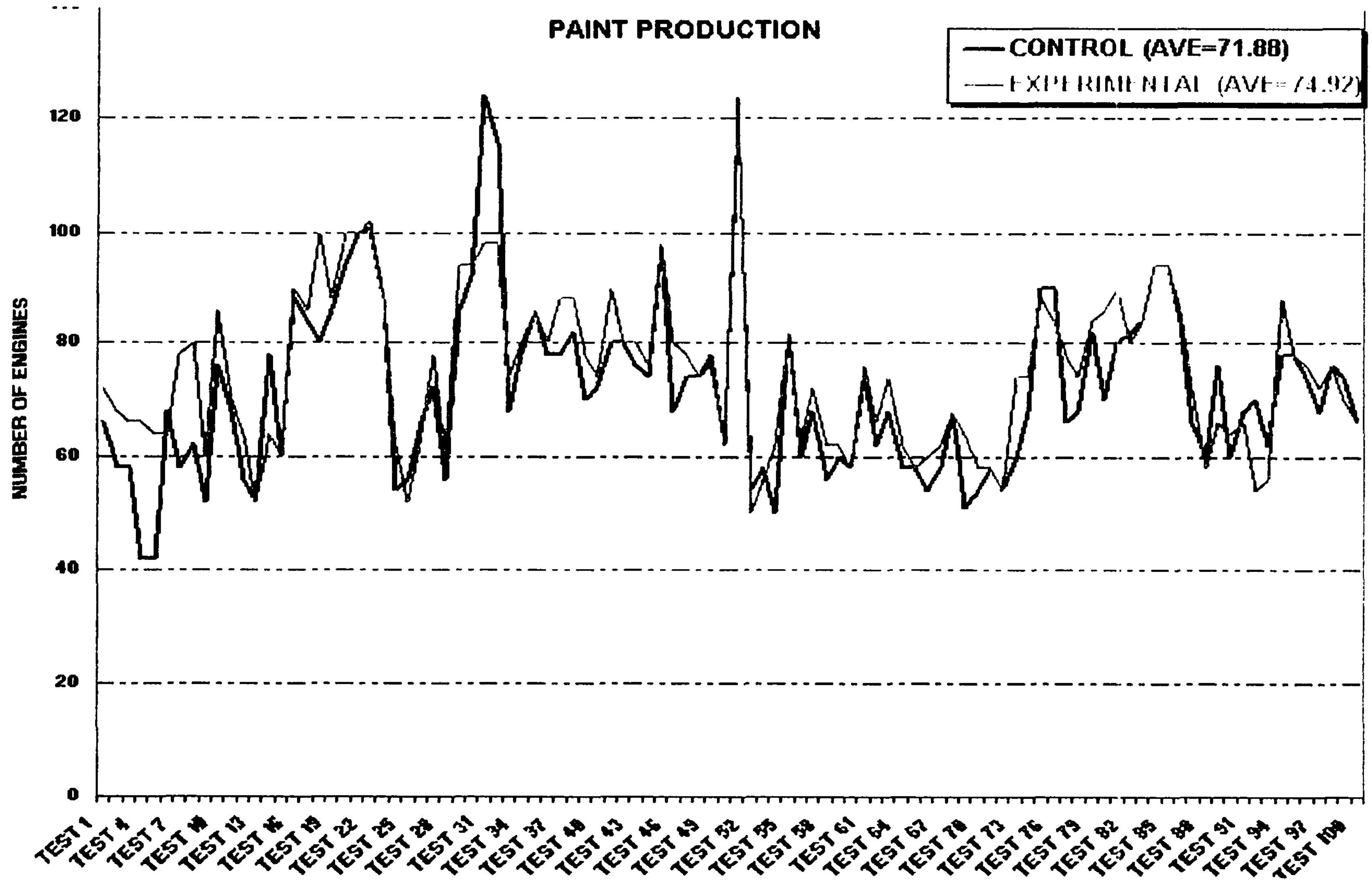


Figure 38. Paint output for the control and the experimental conditions.

A smaller standard deviation for both total output and paint production indicated less variability in the system for the experimental condition. As mentioned earlier, a lesser amount of variability is better in the manufacturing environment. The relatively small standard deviation for total output and paint production indicates more consistent production was achieved for the experimental condition.

CHAPTER V

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Summary

This research was an extension of a previous unpublished study, which investigated the PP&C methods being used at a midwestern manufacturing organization involved in the production of agricultural equipment. The current research study identified the constraints inherent in the production planning and control system and then developed and validated a master production scheduling and sequencing optimization model based on constraints management and utilizing genetic algorithms.

As noted earlier, production planning and control are among the most critical activities in manufacturing. The expected results of this research were to allow manufacturing organizations to maximize the effectiveness of PP&C methods, thereby improving their competitive position in the global economy. To that end, the goal of this research was to develop an optimization model based on constraints management and genetic algorithms to address the constraints in the PP&C methods being used at the factory under study. Published reports of the application of CM in a line assembly environment have been limited. However, according to the research literature, CM has been applied successfully in the job shop environment. In the current research, only three of the five steps of CM were applied. Although the results for the five variables were not statistically significant, results for the experimental condition were the same or better than those for the control condition. It is important to note that improvements are more difficult to achieve in a line assembly environment because there is much less flexibility than in a job shop environment.

The specific objectives of this research were as follows: (a) identify the system's constraints, (b) develop a scheduling and sequencing model to address the identified constraints, (c) develop and validate the proposed model by simulation, and (d) identify and document improvements attributed to the operational change resulting from the implementation of the optimization model.

The research examined the impact of the master production scheduling and sequencing model based on constraints management and utilizing genetic algorithms on five variables for the final assembly line and four downstream processes at an engine manufacturing plant (EMP) of a midwestern manufacturer of agricultural equipment (MMAE). The variables were cycle time, queue size, utilization of work centers, flow rate of engines, and total output of engines.

A two-part model based on constraints management philosophy of production planning and control methods was developed by the researcher in Excel, one part for scheduling and the other for sequencing. Using data from 100 production days during the fall of 1999 and the spring of 2000, simulations for the current scheduling and sequencing method (the control condition) and for the proposed method (the experimental condition) were compared. Output from the simulations for the experimental and control conditions was statistically analyzed.

Conclusions

In the interpretation of output from the simulation runs, it is important to note that daily simulation runs were discrete in nature. Lineup data for each simulation run were used exclusively for that simulation run only; there was no carryover capacity or other resources from previous days to be used the next day. If the production of the constraint operation was reduced for some reason, makeup the next day would not be possible because new line-up data would initiate the next day's simulation run.

During the 200 simulations run, the cycle time of engines for the final assembly line and four downstream processes was reduced, but the reduction was not statistically significant. Queue size was also reduced, as expected, but once again, the reduction was not statistically significant. Total utilization of work centers was increased, as expected, and the increase was statistically significant. Improvement for the flow rate of engines was minimal. The total output of engines increased, and the increase was statistically significant.

Every effort was made to simulate the actual manufacturing environment of the EMP. But since simulation models are just abstractions of reality, they cannot completely mirror the real-world system under study (Law & Kelton, 1991). Results from the simulation outputs can provide insight as to how and why performance for the experimental condition and the control condition differed (Guide, 1992). However, the effectiveness of this model cannot be known conclusively until it is properly implemented at EMP in the fall of 2000.

The exact results of this research are only applicable for the EMP if the manufacturing environment replicated in the model still exists. Generalizations of the findings of this research should be made with caution.

Recommendations

The following recommendations for future research are provided in view of the findings of this study:

1. In this research, all simulation parameters (shipping schedule, initial inventory, process cycle times, operating schedule by department, number of shifts in operation, number of operators/shift, number of load bars in the system) were held constant for the control and the experimental conditions, except the line-up sequence. It is recommended that the values for the simulation parameters could be manipulated.

2. This research model was designed for the assembly operation, but a similar model could be developed for the manufacturing environment, particularly repetitive-type operations.

3. Data collection for the variables during the simulation runs was limited in scope. Only averages and minimum and maximum values were collected. Averages do not always paint a complete picture of the situation. For example the researcher observed during the simulation runs for queue size that the number of engines at 8:00 a.m. in front of one process for the control condition was zero and an hour later that number was 15. The average for two hours was 7.5. Queue sizes for the experimental condition simulation during the same time period were 8 and 7 for an average of 7.5. Because only averages were recorded, performance for both conditions appeared to be the same. But in reality, this would not be the case. The experimental condition's results would be preferred because of the consistency of queue size. In the future, simulation data should include different measures, ones that more accurately reflect reality.

4. It is recommended that multiple models could be built, based on different production planning and control strategies (JIT, MRP, etc.), and the results could be compared and analyzed.

REFERENCES

Aggarwal, S. C. (1985). MRP, JIT, OPT, FMS? Making sense of production operations systems. Harvard Business Review, 63(5).

Amerine, H. T., Ritchey, J. A., Moodie, C. L., & Kmec, J. F. (1993). Manufacturing organization and management (6th ed.). Englewood Cliffs, NJ: Prentice Hall.

Anderson, J. C., Schroeder, R. G., Tupy, S. E., & White, E. M. (1982). Material requirements planning systems: The state of the art. Production and Inventory Management Journal, 22(4), 51-67.

Ashton, J. E., Johnson, M. D., & Cook, F. Y. (1990). Shop floor control in a system job shop: Definitely not MRP. Production and Inventory Management Journal, 31(2), 27-31.

Back, T. (1996). Evolutionary algorithms in theory and practice. New York: Oxford University Press.

Back, T., Hammel, U., & Schwefel, H. (1997). Evolutionary computation: Comments on the history and current state. IEEE Transaction on Evolutionary Computation, 1(1), 3-17.

Baghi, S., Uckun, S., Miyabe, Y., & Kawamura, K. (1991). Exploring problem-specific recombination operators for job shop scheduling. Proceeding of the Fourth International Conference on Genetic Algorithms (pp. 10-17). San Mateo, CA: Morgan Kaufmann.

Bai, S. X., & Tsai, Y. -K. (1994). A production control problem in competition. Computers Math. Applic, 29(5), 65-80.

Bartezzaghi, E., & Turco, F. (1989). The impact of just-in-time on production system performance: An analytical framework. International Journal of Operations and Production Management, 9(8), 40-61.

Bennett, K., Ferris, M., & Ioannidis, Y. (1991). A genetic algorithm for database query optimization. Proceedings of the Fourth International Conference on Genetic Algorithms (pp. 400-407). San Mateo, CA: Morgan Kaufmann.

Berry, W., Schmitt, T., & Vollmann, T. (1982). Capacity planning techniques for manufacturing systems. Journal of Operations Management, 3(1), 13-25.

Bremermann, J. H. (1962). Optimization through evolution and recombination. In M. C.Yovits, G. T. Jacobi, & G. D. Goldstine (Eds.), Self-organizing systems (pp. 93-106). Washington DC: Spartan Books.

Bremermann, J. H. (1967). Quantitative aspects of goal seeking self-organizing systems. Progress in theoretical biology, 1(1), 59-77.

Bremermann, J. H. (1968). Numerical optimization procedures derived from biological evolution process. In H. L. Oestreicher, & D. R. Moore, (Eds.), Cybernetic problems in bionics (pp. 543-562). New York: Gordon & Breach.

Bremermann, J. H. (1973). On the dynamics of trajectories of evolution processes. In A. Locker, (Ed.), Biogenesis, evolution, homeostasis (pp. 29-37), New York: Springer-Verlag.

Burnham, J. M. (1987). Some conclusions about JIT manufacturing. Production and Inventory Control Society, 3(28), 7-10.

Celley, A. F., Clegg, W. H., Smith, A. W., & Vonderembase, M. A. (1986). Implementation of JIT in the United States. Journal of Purchasing and Material Management, 22(4), 9-15.

Chambers, L. (1991). Practical handbook of genetic algorithm applications (Vol.1, preface). New York: Van Nostrand Reinhold.

Choudhry, A. N. (1998). Production planning and control systems in a repetitive manufacturing environment. Unpublished master's research study. University of Northern Iowa, Cedar Falls.

Cleveland, G. A., & Smith, S. F. (1989). Using genetic algorithms to schedule flow shop releases. Proceedings of the Third International Conference on Genetic Algorithms (pp. 160-169). San Mateo, CA: Morgan Kaufmann.

Crossley, W. A. (1995). Using genetic algorithms as an automated methodology for conceptual design of rotorcraft. Unpublished doctoral dissertation, Arizona State University.

Corsten, H., & May, C. (1996). Artificial neural networks for supporting production planning and control. Technovation, 16(2), 67-76.

Cox, J. F., Blackstone, J. H., & Spencer, M. S. (Eds.) (1995). APICS dictionary (8^{th} ed.). Falls Church, VA: APICS.

Cox, J. F., & Clark, S. J. (1984). Problems in implementing and operating a manufacturing resource planning information system. Journal of Management Information Systems, 1(1), 81-99.

Cox, J. F., & Spencer, M. S. (1998). The constraints management handbook. Boca Raton, FL: St. Lucie Press.

Crawford, K. M., Blackstone, J. H., & Cox, J. F. (1988). A study of JIT implementation and operating problems. International Journal of Production Research, 26(9), 1561-1568.

Das, S. K. (1995). MRP-3: An MRP-II extension combining finite capacity scheduling & JIT/kanban type shop floor control. Unpublished doctoral dissertation, The State University of New Jersey, Rutgers.

Davis, L. (1985). Job shop scheduling with genetic algorithms. Proceedings of the First International Conference on Genetic Algorithms. San Mateo, CA: Morgan Kaufmann.

Davis, L. (1991). Handbook of Genetic Algorithms. New York: Van Nostrand Reinhold.

Duclos, L. K., & Spencer, M. S. (1995). The impact of a constraint buffer in a flow shop. International Journal of Production Economics, 42(2), 175-186.

Dugdale, D., & Jones, C. (1995). The theory of constraints. Accountancy, 134 (9), 8-9.

Fraser, A. S. (1957). Simulation of genetic systems by automatic digital computers. Australian Journal of Biology Science, 10, 329-346.

Fraser, A. S. (1962). Simulation of genetic systems. Journal of Theoretical Biology, 2, 329-346.

Fraser, A. S. (1968). The evolution of purposive behavior. In J. D. Foerster, J. D.White, L. J. Peterson, & J. K. Russell, (Eds.), Purposive systems (pp. 15-23). New York: Spartan Books.

Gilbert, J. P. (1990). The state of JIT implementation and development in the USA. International Journal of Production Research, 28(6), 1099-1109.

Gilbert, J. P., & Schonberger, R. J. (1983). Inventory-based production control Systems: A historical analysis. Production and Inventory Management Journal, 24(2), 1-13.

Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. Reading, MA: Addison-Wesley.

Goldratt, E. M. (1984). The goal. Great Barrington, MA: North River Press.

Goldratt, E. M. (1990a). The hystack syndrome sifting information out of the data ocean. New York: North River Press.

Goldratt, E. M. (1990b). What is this thing called theory of constraints and how should it be implemented? New York: North River Press.

Goldratt, E. M. (1994). It's not luck. Great Barrington, MA: North River Press.

Goldratt, E. M., & Fox, R. E. (1986). The race. Great Barrington, MA: North River Press.

Golhar, D. Y., & Stamm, C. L. (1991). The just-in-time philosophy: A literature review. International Journal of Production Research, 29(4), 657-676.

Golhar, D. Y., Stamm, C. L., & Smith, W. P. (1990). JIT implementation in small manufacturing firms. Production and Inventory Management Journal, 31(2), 44-48.

Grenfenstette, J. J. (1987). Incorporating problem specific knowledge into genetic algorithm. In L. D. Davis (Ed.), Genetic algorithms and simulated annealing. London: Pitman.

Guide, V. D., Jr. (1992). A simulation of present production planning and control versus synchronous manufacturing at a Naval Aviation depot. Unpublished doctoral dissertation, University of Georgia, Athens.

Hall, R. W. (1983). Zero inventories. Homewood, IL: Dow Jones-Irwin.

Hall, R. W. (1987). Attaining manufacturing excellence. Homewood, IL: Dow Jones-Irwin.

Hay, E. J. (1988). The just-in-time breakthrough: Implementing the new manufacturing basics. New York: John Wiley & Sons.

Heard, E. (1984). JIT and MRP: Can this marriage be saved? American Production and Inventory Control Society 27th Annual Conference Proceedings (pp. 124-128). Falls Church, VA: APICS.

Hobbs, O. K., Jr. (1997). Managing JIT toward maturity. Production and Inventory Management Journal, 38(1), 47-50.

Holland, J. (1975). Adaptation in natural and artificial systems. Ann Arbor: The University of Michigan Press.

Holland, J. (1992). Genetic algorithms. Scientific American, 267(1), 66-72.

Hopp, W. J., & Spearman, M. L. (1996). Factory physics: Foundations of manufacturing management. Chicago: Irwin.

Holland, J. (1969). Adaptive plans optimal for payoff-only environments. Proceedings of the 2nd Hawaii International Conference on System Sciences (pp. 917-920).

Im, J. H. & Lee, S. M. (1989). Implementation of just in time systems in U.S. manufacturing firms. International Journal of Operations and Production Management, 1(9), 5-13.

Jayson, S. (1987). Goldratt and Fox: Revolutionizing the factory floor. Management Accounting, 69(5), 18-22.

Lambrecht, M., & Decaluwe, L. (1988). JIT and constraints theory: The issue of bottleneck management. Production and Inventory Management Journal, 29(3), 61-65.

Law, A. M., & Kelton, W. D. (1991). Simulation modeling and analysis (2nd ed.). New York: McGraw-Hill.

Manoharan, J. (1997). Performance evaluation of JIT and TOC through simulation. Doctoral dissertation. University of Louisville, Louisville.

Mars, P., Chen, J., & Nambiar, R. (1996). Learning algorithms: Theory and applications in signal processing, control, and communications. Boca Raton, FL: CRC Press.

McMullen, T. B. (1997). The TOC management system for manufacturers. APICS: The Performance Advantage, 7(4) 52-56.

Michalewicz, Z. (1994). Genetic algorithms + data structures = evolution programs(2nd ed.). New York: Springer-Verlag.

Mitchell, M. (1996). An introduction to genetic algorithms. Cambridge, MA: MIT Press.

Monden, Y. (1983). Toyota production systems. Norcross, GA: Industrial Engineering and Management Press.

Nagendra, P. B. (1995). MRP-3: An MRP-II extension combining finite capacity scheduling & JIT/kanban type shop floor control. Unpublished doctoral dissertation, The State University of New Jersey, Rutgers.

Nahmias, S. (1989). Production and operation analysis. New York: Irwin.

Nakano, R., & Yamada, T. (1991). Conventional genetic algorithms for job shop problems. Proceedings of the Third International Conference on Genetic Algorithms (pp. 474-479). San Mateo, CA: Morgan Kaufmann.

Nissen, V. (1993). Evolutionary algorithms in management science: An overview and list of references. University of Gottingen, Interdisziplinares Graduiertenklogg, Gottengen (Germany).

Ohno, T. (1982). The Origin of Toyota production system and kanban system: Applying just-in-time. Norcross, GA: IIE Press.

Orlicky, J. (1975). Materials requirements planning. New York: McGraw-Hill.

Rack, F. H., & Rack, W. K. (1993). Who are managing change? Goldratt Institute Inc. Jonah Course, Texas Tech University.

Ramasesh, R. V. (1992). A logistics-based inventory model for JIT procurement. International Journal of Operations & Production Management, 13(6), 44-58.

Reed, J., Toombs, R., & Barricelli, N. A. (1967). Simulation of biological evolution and machine learning. Journal of Theoretical Biology, 17, 319-342.

Rice, J. W., & Yoshikawa, T. (1982). A comparison of kanban and MRP concepts for the control of repetitive manufacturing systems. Production and Inventory Management Journal, 23(1), 1-12.

Sage, L. (1984). Just in time: A philosophy in manufacturing excellence. Zero Inventory Philosophy & Practices Seminar Proceedings (pp. 3-9). American Production and Inventory Control Society.

Schmitt, T., Berry, W., & Vollmann, T. (1988). An analysis of capacity planning procedures for material requirements planning. Decision Sciences, 15(4), 522-541.

Schonberger, R. J. (1986). World class manufacturing: The lessons of simplicity applied. New York: The Free Press.

Spencer, M. S. (1992). Production planning and control systems in repetitive manufacturing: a comparative analysis based on the product-process structure. Unpublished doctoral dissertation, University of Georgia, Athens.

Spencer, M. S., & Cox, J. F. (1995). Optimum production technology (OPT) and the theory of constraints (TOC): Analysis and genealogy. International Journal of Production Research, 33(6), 1495-1504.

Spencer, M. S., Daugherty, P. J., & Rogers, D. S. (1996). Logistics support for JIT implementation. International Journal of Production Research, 34(3), 701-714.

Sugimoro, Y. (1977). Toyota production system and kanban system. International Journal of Production Research, 15(6), 553-564.

Syswerda, G. (1991). Schedule optimization using genetic algorithms. In L. D. Davis (Ed.), Handbook of genetic algorithms (pp. 332-349). New York: Van Nostrand Reinhold.

Taylor, L. J., III. (1994). A simulation study of protective WIP inventory: A comparative analysis based on material requirements planning, just-in-time, and the theory of constraints in a typical flow shop assembly line environment. Unpublished doctoral dissertation, Texas Tech University, Lubbock.

Umble, M. M., & Srikanth, M. L. (1990). Synchronous manufacturing principles for world class manufacturing. Dallas, TX: South-Western Publishing.

Veral, E. (1995). Using production and transfer batches in flowshops under MRP control. Journal of Operations Management, 12(2), 89-100.

Vollmann, T., Berry, W., & Whybark, D. (1988). Manufacturing planning and control systems (2nd ed.). Homewood, IL: Irwin.

White, R. E. (1993). An empirical assessment of JIT in U.S. manufacturers. Production and Inventory Management Journal, 34(2), 38-42.

Whitley, D., Starkweather, T., & Fuquay, D. (1989). Scheduling and the traveling salesman: The genetic edge recombination operator. Proceedings of the Third International Conference on Genetic Algorithms. San Mateo, CA: Morgan Kaufmann.

Wight, O. (1974). Production and inventory control in the computer age. Boston: Cahners Books.

Wight, O. (1984). Manufacturing resource planning (MRP II): Unlocking America's productivity potential (rev. ed.). Brattleboro, VT: The Book Press.

Williams, D. (1986). JIT versus MRP: The push-pullers call a truce. Automotive Industries, 30-31.

Woeppel, M. (1991). The theory of constraints: A synopsis. The Jonaha Line, 1(1), 1-8.

APPENDIX A

Simulation Output (Control Condition)

SCHEDULING AND SEQUENCING MODEL SIMULATION CONTROL CONDITION

```
TEST: NUMBER 1

INPUT CONDITIONS:
      AVG. LINE RATE-1ST:      130.0 ENGINES/SHIFT
      AVG. LINE RATE-2ND:        0.0 ENGINES/SHIFT
      AVG. LINE RATE-3RD:        0.0 ENGINES/SHIFT
      # LOAD BARS - MAIN:      160
      HEAVY REPAIR:             45.0 MINS.
      LIGHT REPAIR:             20.0 MINS. @   5% REJECT RATE
      CELL DELAY:                5.0 MINS. @  10% DELAY RATE
      # EFFECTIVE DOCKS:         3
-------------------------------------------------------------

RESULTS AFTER:  1 SIMULATION DAYS
------------------------------------


ENGINE PRODUCTION SUMMARY:
---------------------------
                              TOTAL     AVG./DAY
                              -----     --------
J-HOOK PRODUCTION:             105        105.0
TEST PRODUCTION:               112        112.0
CUSTOM TRIM PRODUCTION:         65         65.0
FINAL TRIM PRODUCTION:         109        109.0
PAINT PRODUCTION:               66         66.0
ENGINE SHIPPED:                131        131.0
TRUCKS SHIPPED:                 10         10.0

ENGINE PROCESS SUMMARY:
---------------------------
                                         AVG.      MAX.      MIN.    CURRENT
                                         -----     -----     -----   -------
# ENGINES IN PROCESS/ J-HOOK TO 572:      53.6      100        28        28
# ENGINES IN 572 (TRUCK GRIDS):           67.8      118        11        95
# TRUCK GRIDS:                             5.7       10     -----         7
TOTAL ENGINES AFTER J-HOOK:              121.4      173        83       123
TRUCK DOCK USAGE SUMMARY:                  0.3        3     -----         0

PROCESS TIME IN DAYS/ J-HOOK TO 572:       0.4      1.1       0.1
WAREHOUSE TIME IN DAYS:                    0.4      1.0       0.0
TRUCK LOAD TIME IN DAYS:                   0.2      0.6       0.0

ENGINE FINISH SEQUENCE VARIATION:          0.4       52       -70


FLOW RATE BY DEPARTMENT:
---------------------------
             TOTAL      # ENGINES   #SHIFT   DAYS/   EFFECTIVE    CALCULATED
DEPARTMENT   PRODUCED      /DAY      /DAY    WEEK    MINS./DAY    FLOW RATE
(MINS/ENGINE)
----------   --------   ---------   ------   ----   ----------   ---------
  564          105        105.0        1       5        440           4.2
  568           17         17.0        2       5        880          51.8
  569          112        112.0        3       5       1245          11.1
  570           65         65.0        2       5       1120          17.2
  571          109        109.0        2       5       1120          10.3
  572          131        131.0        1       5        440           3.4
```

```
 570P            66          66.0         2          5        1120            17.0

J-HOOK CHANGEOVERS                 TOTAL        AVG./DAY
-------------------                -----        --------
         # CHANGEOVERS:              4             4.0
         CHANGEOVER TIME(HOURS):    0.2            0.2
         % CHANGEOVER:              3.8 %          ----

HOURLY FLOW METER SUMMARY (UNITS/HOUR)
---------------------------------------
AREA                            AVG.    MAX.    MIN.
---------------                 -----   -----   -----
JHOOK                           11.7    17.0     6.0
TEST                             7.0    15.0     1.0
CUSTOM TRIM                      6.5    13.0     3.0
FINAL TRIM                       6.8    12.0     1.0
PAINT                            5.5     8.0     2.0

CRITICAL QUEUE SUMMARY
-----------------------
AREA                            AVG.    MAX.    MIN.
---------------                 -----   -----   -----
EMPTY                           83.0    90.0    56.0
ATTIC                            1.3    11.0     0.0
TEST LOOP                        6.4    16.0     0.0
CUSTOM TRIM                      4.0    15.0     0.0
FINAL TRIM                      10.3    12.0     0.0
PAINT                            4.7    14.0     0.0

ENGINE PRODUCTION DETAIL:
---------------------------
DAILY ENGINES SHIPPED:
                    PRODUCTION DAYS:
ENGINE                 1    2    3    4    5    6    7    8    9   10     TOTAL
---------------     ---- ---- ---- ---- ---- ---- ---- ---- ---- ----    ------
6081HRW03              4    0    0    0    0    0    0    0    0    0         4
6125HRW01              2    0    0    0    0    0    0    0    0    0         2
6125HRW02              2    0    0    0    0    0    0    0    0    0         2
6105HRW01              1    0    0    0    0    0    0    0    0    0         1
6081TRW01             20    0    0    0    0    0    0    0    0    0        20
6081TRW02             18    0    0    0    0    0    0    0    0    0        18
6081HRW01              7    0    0    0    0    0    0    0    0    0         7
6081HRW05              2    0    0    0    0    0    0    0    0    0         2
6081HRW06              2    0    0    0    0    0    0    0    0    0         2
6081HRW07              3    0    0    0    0    0    0    0    0    0         3
6081HRW08              1    0    0    0    0    0    0    0    0    0         1
6125HRW04              4    0    0    0    0    0    0    0    0    0         4
6081HDW01              2    0    0    0    0    0    0    0    0    0         2
6081HDW05              2    0    0    0    0    0    0    0    0    0         2
6081HDW06              2    0    0    0    0    0    0    0    0    0         2
6101AT012              3    0    0    0    0    0    0    0    0    0         3
6101AT010             12    0    0    0    0    0    0    0    0    0        12
6081HH006              3    0    0    0    0    0    0    0    0    0         3
6081TF001             22    0    0    0    0    0    0    0    0    0        22
6081HF001             12    0    0    0    0    0    0    0    0    0        12
6081AF001              7    0    0    0    0    0    0    0    0    0         7
-------------------------------------------------------------------------------
TOTAL                131    0    0    0    0    0    0    0    0    0       131

DAILY TRUCK SHIPMENT BY CUSTOMER:
                    PRODUCTION DAYS:
```

```
CUSTOMER                1    2    3    4    5    6    7    8    9   10      TOTAL
---------------       ---- ---- ---- ---- ---- ---- ---- ---- ---- ----    ------
WATERLOO                5    0    0    0    0    0    0    0    0    0          5
DAVENPORT               2    0    0    0    0    0    0    0    0    0          2
HITACHI                 1    0    0    0    0    0    0    0    0    0          1
HARVESTER               1    0    0    0    0    0    0    0    0    0          1
OEM                     1    0    0    0    0    0    0    0    0    0          1
----------------------------------------------------------------------------------
TOTAL                  10    0    0    0    0    0    0    0    0    0         10

ENGINE PROCESS DETAIL:
------------------------
PROCESS TIME BETWEEN J-HOOK & 572 (IN HOURS):
ENGINE               # COMPLETE    AVG.     MAX.     MIN.
---------------      ----------   -----    -----    -----
6081HRW03                 8        12.1     21.7      2.3
6125HRW01                 4         8.6     10.5      6.9
6125HRW02                 4         8.7      9.6      7.5
6105HRW01                 2         3.7      4.4      3.1
6081TRW01                56         4.5     24.0      1.4
6081TRW02                46         7.0     22.9      2.0
6081HRW01                14        12.4     22.0      2.3
6081HRW05                 4         7.6     10.8      4.5
6081HRW06                 4        12.4     21.7      2.8
6081HRW07                 6        12.8     22.2      2.8
6081HRW08                 2        12.5     21.9      3.1
6125HRW04                 8         9.6     10.5      8.6
6081HDW01                 4         7.3      9.8      4.7
6081HDW05                 4         7.9     11.4      4.9
6081HDW06                 4         7.4     10.1      5.0
6125ADW70                 2        10.3     10.9      9.8
6101AT012                 6        11.9     12.2     11.6
6101AT010                24        11.3     13.7      9.0
6081HH006                 8         7.8     10.8      4.3
6081TF001                44        10.4     15.3      6.2
6081HF001                24        13.2     26.1      5.2
6081AF001                14         9.8     23.4      5.1
------------------------------------------------------------
TOTAL:                  292         8.9     26.1      1.4

WAREHOUSE TIME (IN HOURS):
ENGINE               # COMPLETE    AVG.     MAX.     MIN.
---------------      ----------   -----    -----    -----
6081HRW03                 8         2.9      4.5      1.7
6125HRW01                 2        17.6     18.1     17.1
6125HRW02                 2        16.7     17.3     16.2
6105HRW01                 1        19.9     19.9     19.9
6081TRW01                40        11.1     21.3      1.4
6081TRW02                36         7.2     18.6      1.1
6081HRW01                14         2.9      4.7      1.2
6081HRW05                 4         6.1     11.0      1.3
6081HRW06                 4         2.6      3.7      1.8
6081HRW07                 6         2.1      3.6      1.1
6081HRW08                 2         2.2      3.2      1.2
6125HRW04                 4        15.2     15.9     14.5
6081HDW01                 4         6.3     11.1      1.3
6081HDW05                 4         6.0     11.1      1.3
6081HDW06                 4         6.3     11.2      1.4
6101AT012                 3        11.1     11.2     10.9
6101AT010                12        13.1     15.2     11.3
6081HH006                 6         6.2     11.5      1.1
```

```
 6081TF001               22       16.6    20.9    11.0
 6081HF001               12       19.7    22.5    10.9
 6081AF001                7       21.7    23.0    20.2
------------------------------------------------------
 TOTAL:                 197       10.3    23.0     1.1

 TRUCK GRID TIME (AWAITING SHIPMENT) IN HOURS:
 CUSTOMER           # COMPLETE    AVG.    MAX.    MIN.
---------------     ----------   -----   -----   -----
 WATERLOO                 9        8.1    21.3     1.9
 DAVENPORT                4        6.4    11.2     1.6
 HITACHI                  1       15.2    15.2    15.2
 HARVESTER                2        6.5    11.5     1.5
 OEM                      1       23.0    23.0    23.0
------------------------------------------------------
 TOTAL:                  17        8.8    23.0     1.5

 TRUCK LOAD TIME IN HOURS:
 CUSTOMER           # COMPLETE    AVG.    MAX.    MIN.
---------------     ----------   -----   -----   -----
 WATERLOO                 9        3.6    14.5     1.1
 DAVENPORT                4        5.8    11.0     1.3
 DUBUQUE                  1       10.9    10.9    10.9
 HARVESTER                2        5.7    10.3     1.1
 OEM                      1       10.9    10.9    10.9
------------------------------------------------------
 TOTAL:                  17        5.3    14.5     1.1

 FINISH SEQUENCE VARIANCE:
 ENGINE             # COMPLETE    AVG.    MAX.    MIN.
---------------     ----------   -----   -----   -----
 6081HRW03                 8       1.6      15     -12
 6125HRW01                 4      -2.5      10     -17
 6125HRW02                 4      -2.7       7      -9
 6105HRW01                 2      37.5      40      35
 6081TRW01                56      19.8      52     -50
 6081TRW02                46       9.6      48     -30
 6081HRW01                14      -3.3      14     -17
 6081HRW05                 4       6.0      23      -6
 6081HRW06                 4       1.0      15     -11
 6081HRW07                 6       0.2      15     -11
 6081HRW08                 2       3.0      14      -8
 6125HRW04                 8      -9.8      -5     -17
 6081HDW01                 4       1.3      16     -14
 6081HDW05                 4      -4.2      16     -16
 6081HDW06                 4       1.5      19     -13
 6125ADW70                 2     -13.5     -11     -16
 6101AT012                 6     -20.8     -18     -22
 6101AT010                24     -17.0      -3     -39
 6081HH006                 8       3.1      17     -13
 6081TF001                44     -19.4      15     -59
 6081HF001                24      -4.2      32     -70
 6081AF001                14       6.3      26     -10
------------------------------------------------------
 TOTAL:                  292       0.4      52     -70

 TECHNICIAN PERFORMANCE BY DEPARTMENT
 DEPT: 568
 OPERATING DAYS/WEEK: 5
                             # ENGINES     AVG. TIME/
 TECHNICIAN        SHIFT     PROCESSED     ENGINE        % ULT.
```

```
---------------------------------------------------------------
  BBTRIM               1          46          4.5         47.3 %
  BBTRIM1              1          46          4.5         47.3 %
  LREPAIR              1           3         20.0         13.6 %
  LEAKTEST2            1          18          7.6         30.9 %
  BBTRIM               2           1          0.0          0.0 %
  BBTRIM1              2           1          0.0          0.0 %
  LREPAIR              2           1          0.0          0.0 %
  LEAKTEST2            2           1          0.0          0.0 %

DEPT: 569
OPERATING DAYS/WEEK: 5
                               # ENGINES     AVG. TIME/
TECHNICIAN           SHIFT     PROCESSED     ENGINE          % ULT.
---------------------------------------------------------------
  TEST5                1          50          7.9         95.0 %
  TEST7                1          46          7.8         86.1 %
  TEST9                1          43          7.6         79.0 %
  TEST11               1          38          7.8         71.4 %
  RETORQ1              1          74          3.6         64.1 %
  RETORQ2              1           7          3.4          4.0 %
  REPAIR1              1           8         37.1         71.5 %
  REPAIR2              1           6         27.6         40.0 %
  REPAIR3              1           5         26.0         31.3 %
  TEST5                2          23          7.3         40.4 %
  TEST7                2          19          7.4         33.9 %
  TEST9                2          20          7.1         34.2 %
  TEST11               2          21          7.3         36.8 %
  RETORQ1              2          29          2.9         20.1 %
  REPAIR1              2           6         24.7         35.7 %
  REPAIR2              2           4         31.0         29.9 %
  REPAIR3              2           3         30.0         21.7 %
  TEST5                3           1          0.0          0.0 %
  TEST7                3           1          0.0          0.0 %
  TEST9                3           1          0.0          0.0 %
  RETORQ1              3           9          2.7          5.8 %
  REPAIR1              3           1          0.0          0.0 %

DEPT: 570
OPERATING DAYS/WEEK: 5
                               # ENGINES     AVG. TIME/
TECHNICIAN           SHIFT     PROCESSED     ENGINE          % ULT.
---------------------------------------------------------------
  CUSTOMT1             1          91          2.4         40.9 %
  CUSTOMT2             1         106          2.3         46.6 %
  CUSTOMT3             1          77          2.4         35.0 %
  CUSTOMT4             1          98          2.3         42.5 %
  CUSTOMT5             1          85          2.5         40.3 %
  CUSTOMT1             2          45          2.8         21.2 %
  CUSTOMT2             2          55          2.5         23.2 %
  CUSTOMT3             2          49          2.1         17.4 %
  CUSTOMT4             2          71          2.2         26.7 %
  CUSTOMT5             2          62          2.2         23.1 %

DEPT: 571
OPERATING DAYS/WEEK: 5
                               # ENGINES     AVG. TIME/
TECHNICIAN           SHIFT     PROCESSED     ENGINE          % ULT.
---------------------------------------------------------------
  FTRIM1               1          43          4.9         39.6 %
  FTRIM2               1          95          1.7         30.1 %
```

```
 FTRIM3           1          108          1.4        28.3 %
 FTRIM4           1          114          1.4        30.2 %
 FTRIM5           1          102          1.5        29.3 %
 FTRIM6           1          100          1.6        29.5 %
 FTRIM7           1          100          1.6        30.6 %
 FTRIM8           1           96          1.5        27.5 %
 FTRIM9           1           70          1.7        22.7 %
 FTRIM1           2           75          5.6        71.3 %
 FTRIM2           2          180          1.6        50.2 %
 FTRIM3           2          224          1.6        59.4 %
 FTRIM4           2          188          1.9        61.6 %
 FTRIM5           2          199          1.7        56.5 %
 FTRIM6           2          199          1.9        63.5 %
 FTRIM7           2          197          1.7        55.8 %
 FTRIM8           2          174          1.9        55.1 %
 FTRIM9           2          121          2.2        45.2 %

DEPT: 572
OPERATING DAYS/WEEK: 5
                               # TRUCKS     AVG. TIME/
TECHNICIAN         SHIFT      PROCESSED       TRUCK       % ULT.
-----------------------------------------------------------------
 ANALYST           1             11          4.5         11.4 %
 SHIPPER1          1             11         10.9         27.3 %
 SHIPPER2          1             11         11.8         29.5 %
 TRUCKER1          1              4         18.7         17.0 %
 TRUCKER2          1              5         20.0         22.7 %
 TRUCKER3          1              4         18.7         17.0 %
 CLERK             1             11          9.1         22.7 %

DEPT: 570 PAINT
OPERATING DAYS/WEEK: 5
                               # ENGINES     AVG. TIME/
TECHNICIAN         SHIFT      PROCESSED     ENGINE       % ULT.
-----------------------------------------------------------------
 PAINTR1           1             27          5.8         29.5 %
 PAINTR2           1             26          6.7         33.1 %
 PAINTR1           2             42          6.2         44.1 %
 PAINTR2           2             46          7.2         55.8 %
```

Simulation Output (Experimental Condition)

SCHEDULING AND SEQUENCING MODEL SIMULATION EXPERIMENTAL CONDITION

```
 TEST: NUMBER 1

 INPUT CONDITIONS:
       AVG. LINE RATE-1ST:     130.0 ENGINES/SHIFT
       AVG. LINE RATE-2ND:       0.0 ENGINES/SHIFT
       AVG. LINE RATE-3RD:       0.0 ENGINES/SHIFT
       # LOAD BARS - MAIN:     160
       HEAVY REPAIR:            45.0 MINS.
       LIGHT REPAIR:            20.0 MINS. @   5% REJECT RATE
       CELL DELAY:               5.0 MINS. @  10% DELAY RATE
       # EFFECTIVE DOCKS:        3
----------------------------------------------------------

 RESULTS AFTER:  1 SIMULATION DAYS
-----------------------------------


 ENGINE PRODUCTION SUMMARY:
--------------------------
                            TOTAL     AVG./DAY
                            -----     --------
 J-HOOK PRODUCTION:           106        106.0
 TEST PRODUCTION:             111        111.0
 CUSTOM TRIM PRODUCTION:       68         68.0
 FINAL TRIM PRODUCTION:       111        111.0
 PAINT PRODUCTION:             72         72.0
 ENGINE SHIPPED:              139        139.0
 TRUCKS SHIPPED:               17         17.0

 ENGINE PROCESS SUMMARY:
--------------------------
                                          AVG.      MAX.      MIN.    CURRENT
                                         -----     -----     -----    -------
 # ENGINES IN PROCESS/ J-HOOK TO 572:     49.6        87        30         30
 # ENGINES IN 572 (TRUCK GRIDS):          54.3        99        11         68
 # TRUCK GRIDS:                            8.7        15     -----         10
 TOTAL ENGINES AFTER J-HOOK:             103.9       150        86         98
 TRUCK DOCK USAGE SUMMARY:                 0.5         3     -----          0

 PROCESS TIME IN DAYS/ J-HOOK TO 572:      0.4       1.1       0.1
 WAREHOUSE TIME IN DAYS:                   0.4       0.9       0.0
 TRUCK LOAD TIME IN DAYS:                  0.3       0.7       0.0

 ENGINE FINISH SEQUENCE VARIATION:         0.5        42       -54


 FLOW RATE BY DEPARTMENT:
--------------------------
             TOTAL      # ENGINES    #SHIFT   DAYS/   EFFECTIVE    CALCULATED
 DEPARTMENT  PRODUCED     /DAY        /DAY    WEEK    MINS./DAY    FLOW RATE
(MINS/ENGINE)
 ----------  --------   ---------    ------   ----   ----------    ---------
   564          106       106.0        1        5       440            4.2
   568           20        20.0        2        5       880           44.0
   569          111       111.0        3        5      1245           11.2
   570           68        68.0        2        5      1120           16.5
   571          111       111.0        2        5      1120           10.1
   572          139       139.0        1        5       440            3.2
```

```
  570P            72          72.0        2       5       1120           15.6

 J-HOOK CHANGEOVERS                        TOTAL       AVG./DAY
-------------------                        -----       --------
          # CHANGEOVERS:                     7            7.0
          CHANGEOVER TIME(HOURS):          0.3            0.3
          % CHANGEOVER:                    6.6 %          ----

 HOURLY FLOW METER SUMMARY (UNITS/HOUR)
--------------------------------------
 AREA                                 AVG.     MAX.     MIN.
---------------                      -----    -----    -----
 JHOOK                                11.8     18.0      6.0
 TEST                                  9.3     15.0      1.0
 CUSTOM TRIM                           6.2     11.0      1.0
 FINAL TRIM                            7.9     13.0      3.0
 PAINT                                 5.5      8.0      2.0

 CRITICAL QUEUE SUMMARY
-----------------------
 AREA                                 AVG.     MAX.     MIN.
---------------                      -----    -----    -----
 EMPTY                                85.5     90.0     67.0
 ATTIC                                 0.8      9.0      0.0
 TEST LOOP                             1.2      8.0      0.0
 CUSTOM TRIM                           4.7     14.0      1.0
 FINAL TRIM                            9.9     12.0      2.0
 PAINT                                 5.7     15.0      0.0

 ENGINE PRODUCTION DETAIL:
---------------------------
 DAILY ENGINES SHIPPED:
                     PRODUCTION DAYS:
 ENGINE                 1    2    3    4    5    6    7    8    9   10    TOTAL
---------------      ---- ---- ---- ---- ---- ---- ---- ---- ---- ----   ------
 6081HRW03              8    0    0    0    0    0    0    0    0    0        8
 6125HRW01              2    0    0    0    0    0    0    0    0    0        2
 6125HRW02              2    0    0    0    0    0    0    0    0    0        2
 6105HRW01              1    0    0    0    0    0    0    0    0    0        1
 6081TRW01             20    0    0    0    0    0    0    0    0    0       20
 6081TRW02              9    0    0    0    0    0    0    0    0    0        9
 6081HRW01             14    0    0    0    0    0    0    0    0    0       14
 6081HRW05              2    0    0    0    0    0    0    0    0    0        2
 6081HRW06              4    0    0    0    0    0    0    0    0    0        4
 6081HRW07              6    0    0    0    0    0    0    0    0    0        6
 6081HRW08              2    0    0    0    0    0    0    0    0    0        2
 6125HRW04              4    0    0    0    0    0    0    0    0    0        4
 6081HDW01              2    0    0    0    0    0    0    0    0    0        2
 6081HDW05              2    0    0    0    0    0    0    0    0    0        2
 6081HDW06              2    0    0    0    0    0    0    0    0    0        2
 6101AT012              3    0    0    0    0    0    0    0    0    0        3
 6101AT010             12    0    0    0    0    0    0    0    0    0       12
 6081HH006              3    0    0    0    0    0    0    0    0    0        3
 6081TF001             22    0    0    0    0    0    0    0    0    0       22
 6081HF001             12    0    0    0    0    0    0    0    0    0       12
 6081AF001              7    0    0    0    0    0    0    0    0    0        7
--------------------------------------------------------------------------------
 TOTAL                139    0    0    0    0    0    0    0    0    0      139

 DAILY TRUCK SHIPMENT BY CUSTOMER:
                     PRODUCTION DAYS:
```

```
CUSTOMER           1    2    3    4    5    6    7    8    9   10    TOTAL
---------------  ---- ---- ---- ---- ---- ---- ---- ---- ---- ----   ------
WATERLOO           9    0    0    0    0    0    0    0    0    0        9
DAVENPORT          1    0    0    0    0    0    0    0    0    0        1
HITACHI            2    0    0    0    0    0    0    0    0    0        2
HARVESTER          1    0    0    0    0    0    0    0    0    0        1
OEM                4    0    0    0    0    0    0    0    0    0        4
--------------------------------------------------------------------------
TOTAL             17    0    0    0    0    0    0    0    0    0       17

ENGINE PROCESS DETAIL:
-----------------------
PROCESS TIME BETWEEN J-HOOK & 572 (IN HOURS):
ENGINE            # COMPLETE   AVG.    MAX.    MIN.
---------------   ----------   -----   -----   -----
6081HRW03                  8     4.5     4.8     4.1
6125HRW01                  3     9.1     9.6     8.8
6125HRW02                  4     9.9    12.3     8.6
6105HRW01                  2     5.4     7.0     3.9
6081TRW01                 59     6.5    21.9     1.4
6081TRW02                 18    12.2    21.6     2.3
6081HRW01                 14     4.7     6.9     3.5
6081HRW05                  4     9.6    10.3     8.9
6081HRW06                  4     4.3     5.7     3.1
6081HRW07                  6     4.4     5.9     3.3
6081HRW08                  2     6.1     8.7     3.4
6125HRW04                  8     9.3    10.0     8.7
6081HDW01                  4    11.6    25.6     5.5
6081HDW05                  4     7.9     9.1     5.4
6081HDW06                  4     9.1    17.0     5.3
6125ADW70                  2     9.5    10.0     9.1
6101AT012                  6     9.9    10.7     9.0
6101AT010                 24     8.6    12.6     6.9
6081HH006                  6     7.1     9.6     4.2
6081TF001                 44    11.8    26.1     6.3
6081HF001                 24    11.2    21.8     5.2
6081AF001                 14     8.5    23.1     4.2
----------------------------------------------------
TOTAL:                   264     8.7    26.1     1.4

WAREHOUSE TIME (IN HOURS):
ENGINE            # COMPLETE   AVG.    MAX.    MIN.
---------------   ----------   -----   -----   -----
6081HRW03                  8    10.6    19.3     2.2
6125HRW01                  2    15.5    15.6    15.4
6125HRW02                  2    13.8    15.6    12.0
6105HRW01                  1    16.9    16.9    16.9
6081TRW01                 40     6.7    17.3     1.2
6081TRW02                 18     3.8     7.1     1.1
6081HRW01                 14    10.8    20.4     1.4
6081HRW05                  2    15.0    15.0    15.0
6081HRW06                  4    10.7    20.1     1.5
6081HRW07                  6    10.4    20.0     1.4
6081HRW08                  2     8.6    16.0     1.3
6125HRW04                  4    14.9    15.3    14.0
6081HDW01                  4     5.3    17.2     1.2
6081HDW05                  4     8.4    14.2     1.3
6081HDW06                  4     7.0    14.3     3.9
6101AT012                  3    15.0    15.7    13.9
6101AT010                 12    17.1    19.0    12.6
6081HH006                  6     7.1    12.7     1.1
```

```
6081TF001              22        17.5     20.6     14.7
6081HF001              24         5.5     14.6      1.7
6081AF001              14         7.7     15.7      1.2
-----------------------------------------------------------
TOTAL:                196         9.5     20.6      1.1

TRUCK GRID TIME (AWAITING SHIPMENT) IN HOURS:
CUSTOMER          # COMPLETE     AVG.     MAX.     MIN.
---------------   ----------    -----    -----    -----
WATERLOO               12        10.7     20.4      1.9
DAVENPORT               2         9.4     17.2      1.6
HITACHI                 2        17.9     19.0     16.7
HARVESTER               2         7.5     12.7      2.2
OEM                     6        12.7     20.6      3.2
-----------------------------------------------------------
TOTAL:                 24        11.4     20.6      1.6

TRUCK LOAD TIME IN HOURS:
CUSTOMER          # COMPLETE     AVG.     MAX.     MIN.
---------------   ----------    -----    -----    -----
WATERLOO               12         6.8     16.8      1.1
DAVENPORT               3         1.2      1.3      1.2
HITACHI                 2        14.6     16.7     12.6
HARVESTER               2         6.7     12.3      1.1
OEM                     5         7.1     16.6      1.2
-----------------------------------------------------------
TOTAL:                 24         6.8     16.8      1.1

FINISH SEQUENCE VARIANCE:
ENGINE            # COMPLETE     AVG.     MAX.     MIN.
---------------   ----------    -----    -----    -----
6081HRW03               8        23.6       29       19
6125HRW01               3        -8.0       -2      -18
6125HRW02               4       -14.8        2      -29
6105HRW01               2        25.0       30       20
6081TRW01              59        13.8       42      -20
6081TRW02              18         9.9       29      -25
6081HRW01              14        18.1       26        2
6081HRW05               4       -11.8       -4      -20
6081HRW06               4        34.0       41       25
6081HRW07               6        33.7       41       24
6081HRW08               2        20.0       40        0
6125HRW04               8       -10.3        1      -19
6081HDW01               4       -20.5        3      -54
6081HDW05               4         1.5       17      -25
6081HDW06               4         8.3       20       -2
6125ADW70               2       -11.0       -7      -15
6101AT012               6       -15.2       -6      -21
6101AT010              24       -13.5        0      -34
6081HH006               6        -1.8        6      -13
6081TF001              44       -17.6        5      -48
6081HF001              24        -4.1       13      -23
6081AF001              14       -10.6        2      -34
-----------------------------------------------------------
TOTAL:                264         0.5       42      -54

TECHNICIAN PERFORMANCE BY DEPARTMENT
DEPT: 568
OPERATING DAYS/WEEK: 5
                                # ENGINES      AVG. TIME/
TECHNICIAN          SHIFT       PROCESSED      ENGINE         % ULT.
```

```
----------------------------------------------------------------
  BBTRIM            1           48          4.4          48.4 %
  BBTRIM1           1           47          4.5          47.9 %
  LREPAIR           1            3         20.0          13.6 %
  LEAKTEST2         1           16          7.5          27.3 %
  BBTRIM            2           10          3.2           7.2 %
  BBTRIM1           2            9          3.5           7.2 %
  LREPAIR           2            1          0.0           0.0 %
  LEAKTEST2         2            6          6.7           9.1 %

 DEPT: 569
 OPERATING DAYS/WEEK: 5
                             # ENGINES     AVG. TIME/
 TECHNICIAN         SHIFT    PROCESSED     ENGINE         % ULT.
----------------------------------------------------------------
  TEST5             1            49          8.0          94.4 %
  TEST7             1            45          8.1          87.5 %
  TEST9             1            43          8.2          84.8 %
  TEST11            1            41          8.3          82.3 %
  RETORQ1           1            63          3.7          56.4 %
  RETORQ2           1            17          3.2           9.2 %
  REPAIR1           1             7         29.7          50.1 %
  REPAIR2           1             6         18.4          26.6 %
  REPAIR3           1             5         36.0          43.4 %
  TEST5             2            24          7.5          43.5 %
  TEST7             2            20          7.2          34.6 %
  TEST9             2            18          7.3          31.7 %
  TEST11            2            18          7.4          32.2 %
  RETORQ1           2            35          3.0          25.3 %
  REPAIR1           2             5         24.4          29.4 %
  REPAIR2           2             4         32.4          31.2 %
  REPAIR3           2             3         30.0          21.7 %
  TEST5             3             1          0.0           0.0 %
  TEST7             3             1          0.0           0.0 %
  TEST9             3             1          0.0           0.0 %
  RETORQ1           3             1          0.0           0.0 %
  REPAIR1           3             1          0.0           0.0 %

 DEPT: 570
 OPERATING DAYS/WEEK: 5
                             # ENGINES     AVG. TIME/
 TECHNICIAN         SHIFT    PROCESSED     ENGINE         % ULT.
----------------------------------------------------------------
  CUSTOMT1          1            87          2.9          47.0 %
  CUSTOMT2          1           100          2.6          49.9 %
  CUSTOMT3          1            86          2.1          34.1 %
  CUSTOMT4          1           101          2.4          46.0 %
  CUSTOMT5          1            90          2.3          38.4 %
  CUSTOMT1          2            59          1.6          16.2 %
  CUSTOMT2          2            65          2.3          24.8 %
  CUSTOMT3          2            55          2.1          19.8 %
  CUSTOMT4          2            60          2.3          23.0 %
  CUSTOMT5          2            57          2.2          21.4 %

 DEPT: 571
 OPERATING DAYS/WEEK: 5
                             # ENGINES     AVG. TIME/
 TECHNICIAN         SHIFT    PROCESSED     ENGINE         % ULT.
----------------------------------------------------------------
  FTRIM1            1            55          5.1          52.9 %
  FTRIM2            1           128          1.6          38.0 %
```

```
  FTRIM3              1              155            1.3          39.3 %
  FTRIM4              1              144            1.6          43.5 %
  FTRIM5              1              141            1.6          41.3 %
  FTRIM6              1              131            1.7          40.9 %
  FTRIM7              1              124            1.5          36.2 %
  FTRIM8              1              128            1.7          41.4 %
  FTRIM9              1              102            1.8          34.3 %
  FTRIM1              2               62            5.5          58.1 %
  FTRIM2              2              154            1.6          41.5 %
  FTRIM3              2              196            1.5          50.8 %
  FTRIM4              2              148            1.9          48.8 %
  FTRIM5              2              153            1.7          43.4 %
  FTRIM6              2              155            1.9          49.2 %
  FTRIM7              2              157            1.6          42.9 %
  FTRIM8              2              148            1.8          44.5 %
  FTRIM9              2              103            2.1          36.4 %

DEPT: 572
OPERATING DAYS/WEEK: 5
                                # TRUCKS      AVG. TIME/
TECHNICIAN         SHIFT       PROCESSED       TRUCK        % ULT.
-----------------------------------------------------------------
  ANALYST             1               17            4.7          18.2 %
  SHIPPER1            1               16           12.2          44.3 %
  SHIPPER2            1               18           11.4          46.6 %
  TRUCKER1            1                6           20.8          28.4 %
  TRUCKER2            1                7           19.6          31.2 %
  TRUCKER3            1                7           21.4          34.1 %
  CLERK               1               18            9.4          38.6 %

DEPT: 570 PAINT
OPERATING DAYS/WEEK: 5
                                # ENGINES      AVG. TIME/
TECHNICIAN         SHIFT       PROCESSED      ENGINE        % ULT.
-----------------------------------------------------------------
  PAINTR1             1               31            6.0          35.4 %
  PAINTR2             1               30            6.8          38.7 %
  PAINTR1             2               43            6.0          43.5 %
  PAINTR2             2               48            7.2          58.4 %
```

APPENDEX B

Simulation Code

SCHEDULING AND SEQUENCING MODEL SIMULATION CODE

```
DEVELOPED BY G. Rehn

*         SIMULATE 3
          REALLOCATE COM,900000
          REALLOCATE STO,500,CHA,500
          REALLOCATE FAC,500,HSV,300,GRP,300
*
OCOLORC STARTMACRO
PUTTIME MACRO
        BPUTPIC         FILE=ATF,(#A,#B)
Set C* Color *
        ENDMACRO
*
          INTEGER     &I,&J,&K,&L,&M,&N,&CDOWN(18),&MAX,&CHOK,&PRVENG
          INTEGER     &KEYCNT,&EFAM(20)
          REAL        &R,&S,&T,&DAY,&CONVS,&JHKCO,&PRORATE(6),&JHKCOTIM
          REAL        &MTBF(20),&DTIM(20),&DELAY1(20)
          CHAR*12     &PRTNO(100),&ENG,&CLR(8),&ECLR(100),&NULL
          VCHAR*12    &PRVCUS,&PCLR(6),&NUM,&ITOCHAR(50)
*
****************************************************
*
* Define job attributes (fullword)
*
 ENGINE   EQU       1,PF                 //ENGINE ID
 DELRT    EQU       2,PF                 //DELIVERY DESTINATION
 REPAIR   EQU       3,PF                 //ENGINE REPAIR INDICATOR
 RETEST   EQU       4,PF                 //RETEST COUNT
 CVSEC    EQU       5,PF                 //CURRENT CONTROL ZONE
 SEQNM    EQU       6,PF                 //ASSEMBLY SEQUENCE #
 SUBR     EQU       7,PF                 //SUBROUTINE PARAMETER
 PTR      EQU       8,PF                 //POINTER PARAMETER
 CTR      EQU       9,PF                 //COUNTER PARM.
 PLOC     EQU       10,PF                //PREVIOUS location
 CLOC     EQU       11,PF                //CURRENT LOCATION
 OPNUM    EQU       12,PF                //OPERATION #
 INDX     EQU       13,PF                //ULT. INDEX
 JNDX     EQU       14,PF                //ULT. INDEX
 SHFT     EQU       15,PF                //TECHNICIAN SHIFT
 PCT      EQU       16,PF                //WORKING PCT.
 RJCT     EQU       17,PF                //REJECT INDICATOR (0=NO;1=YES)
 SSEQN    EQU       18,PF                //GRAND SEQUENCE #
 LCTR     EQU       19,PF              //LOOP COUNTER
 MOD      EQU       20,PF                //MODULE INDICATOR
 LOC1     EQU       21,PF                //LOCATION 1
 LOC2     EQU       22,PF                //LOCATION 2
 LOC3     EQU       23,PF                //LOCATION 3
 LOC4     EQU       24,PF                //LOCATION 4
 LOC5     EQU       25,PF                //LOCATION 5
 LOC6     EQU       26,PF                //LOCATION 6
 OPER1    EQU       27,PF                //1ST OPERATOR IN SERIES
 OPERL    EQU       28,PF                //LAST OPERATOR IN SERIES
 TECHN    EQU       29,PF                //TECHNICIAN #
 PCODE    EQU       30,PF                //LOAD BAR PROCESS CODE
 NOOPR    EQU       31,PF
 TSEQN    EQU       32,PF                //TRUCK SEQUENCE#
*
* Define job attributes (floating)
*
 DELAY    EQU       1,PL                 //HOT JOB DELAY
 ITIME    EQU       2,PL                 //INDEX TIME
 WAIT     EQU       3,PL                 //WAIT TIME
 CYCLE    EQU       4,PL                 //CYCLE
 LAPTIM   EQU       5,PL                 //TOTAL SYSTEM TIME
 CMPEST   EQU       6,PL                 //COMPLETION ESTIMATE
```

```
ACMBRK    EQU         7,PL                //ACCUMULATED BREAK
*
****************************************************
*
* File Variables.
*
          VCHAR*8    &TESTID              Test ID.
          VCHAR*80   &TESTDSCR            Test description.
          REAL       &PRODVOL(3)          Daily production volume.
          INTEGER    &RUNDAYS             # of days to run.
          INTEGER    &LBCTMAIN            Load bar count in main conveyor.
          INTEGER    &LBCTPNT             Load bar count in paint.
          REAL       &TIMEOVEN            Time in paint oven (min).
*
          REAL       &TRIM(100)            Time: Trim
          REAL       &TXFR(100)            Time: Transfer
          REAL       &TRM571(100)          Time: Final Trim
*
* Data Declarations
*
          INTEGER    &SHIFTNO,&STAOPMIN,&STAOPMID,&STAOPMAX,&DAYNO
          VCHAR*100  &STRING1
          INTEGER    &OFLDENGS
          INTEGER    &FTRMENGS
          INTEGER    &BBMRG(40)
*
* CUSTOM TRIM STORAGES: 140-159
*
CTRMQ     EQU        140,S,L         //TRIM STATION
CTLINE    EQU        141(9),S,L      //TRIM LINE
SSTG0     EQU        155,S,L,C       //STAGING ZONE
STRMI     EQU        156,S,L,C       //DELIVERY PATH TO TRIMS
STRM1     EQU        157,S,L,C       //TRANSFER SWITCH
TRMOUT    EQU        158,S,L,C       //EXIT TRIM AREA
TCTLINE   EQU        151,S
TFTLINE   EQU        152,S
*
          STORAGE    S140,17/S141-S148,1/S149,7/S(SSTG0),2
          STORAGE    S(STRMI),5/S(STRM1),1/S(TRMOUT),5
          STORAGE    S(TCTLINE),5/S(TFTLINE),9
*
* FINAL TRIM Storages: 160-179
*
FTRMQ     EQU        160,S,L              //FINAL TRIM QUEUE
FTLINE    EQU        161(9),S,L           //FINAL TRIM STATION
BBSWT     EQU        179,L                //BACKBONE SWITCH
*
          STORAGE    S(FTRMQ),12
          STORAGE    S161-S169,1
*
* Paint Loop Storages: 180-199
*
SPNT1     EQU        180,S
SPNT2     EQU        181,S
SPNT3     EQU        182,S
SPNT4     EQU        183,S
SPNT5     EQU        184,S
SPNT6     EQU        185,S
SPNT7     EQU        186,S
SPNT8     EQU        187,S,C
*
          STORAGE    S(SPNT1),20/S(SPNT2),3/S(SPNT3),3/S(SPNT4),1
          STORAGE    S(SPNT5),1/S(SPNT6),12/S(SPNT7),5/S(SPNT8),11
*
* 569 Repair Storages: 250-259
*
REPRQ     EQU        250,S,L                        //REPAIR QUEUE
EREPR     EQU        259,S                          //EXIT REPAIR
*
```

```
         STORAGE    S(REPRQ),4/S(EREPR),1
*
* 568 Process Storages: 260-279
*
 PRO568  EQU        260(9),F,S,C,L                //568 REPAIR
 TRM568  EQU        271(6),F,S,C,L                //568 TRIM LINE
 LEAKQ   EQU        269,S,L,C                     //LEAK TEST QUEUE
 LKTST2  EQU        270,S                         //LEAK TEST
         STORAGE    S260,5/S261-S263,8            //REPAIR SPURS
         STORAGE    S(LEAKQ),14/S270,1            //LEAK TEST QUEUE
         STORAGE    S272-S276,1/S271,3            //568 TRIM LINE
*
* Power & Free Storages: 280-290
*
 SPF1    EQU       281,S,L
 SPF2    EQU       282,S
 SPL     EQU       283,S
 SPO     EQU       284,S,F
*
         STORAGE   S(SPF2),9/S(SPL),1/S(SPO),5
*
* Gather statistics, traffic control Storages: 290+
*
DOCKS    EQU       290,S
EWIPQ    EQU       292,S                  //ENGINES FROM JHOOK TO 572
TOTALQ   EQU       293,S                  //TOTAL ENGINES IN FACTORY
*
         STORAGE   S200,1
*
* TRANSIT COUNTS STORAGES 301-304
*
 CSTRMCNT EQU      301,S                  //CUSTOM TRIM TRANSIT COUNT
 FNTRMCNT EQU      302,S                  //FINAL TRIM TRANSIT COUNT
 PAINTCNT EQU      303,S                  //PAINT TRANSIT COUNT
 TSTLCOUT EQU      304,S                  //TEST LOOP COUNT OUT
 BACKBCNT EQU      305,S,L,C              //BACKBONE TRANSIT LIMIT
 BLUBFQ   EQU      306,C,Q,L              //BLUEBIRD FLOOR QUEUE
*
************************************************************
*
* Define Facilities and Storages.
*
 BBCNV    EQU      1(26),S,L
 MORN     EQU      27,L
 COUNT    EQU      28,XF,L                     //OLD RC COUNT
 LJHOOK   EQU      29,L                        //Logic flag: JHOOK transfer.
 LTXFR    EQU      30,L                        //Logic flag: TXFR  transfer.
 CELLSW   EQU      31(6),L,S
 RTORKQ   EQU      37,S,C,L                    //RETORQUE QUEUE
 RTORQ1   EQU      38,S,L                      //RETORQUE STA 1
 RTORQ2   EQU      39,S,L                      //RETORQUE STA 2
 RTORQE   EQU      40,S,L                      //RETORQUE QUEUE
 BACKUP   EQU      400,S
 RECR1I   EQU      401,S,L                     //569 RECIRC IN
 RECR1    EQU      402,S,L,C                   //569 RECIRC
 RECR1O   EQU      403,S,L                     //569 EXIT
 SPIN1    EQU      404,L                       //INDICATES RC 1 SEARCHING
 RCCAL1   EQU      404,C                       //CALL CHAIN FOR RC1
 RECR2I   EQU      405,S,L                     //570 RECIRC IN
 RECR2M   EQU      406,S,L,C                   //570 RECIRC MIDDLE
 RECR2O   EQU      407,S,L                     //570 EXIT
 RECR2    EQU      408,S                       //TOTAL RECIR LOOP
 SPIN2    EQU      409,L                       //INDICATES RC 2 SEARCHING
 RCCAL2   EQU      410,C                       //RECIRC #2 CALL CHAIN
 FAILR    EQU      450,C                       //HOLD FAILURE COUNT
 DLAYSW   EQU      451(20),L                   //DELAY CONDITION SWITCHES
*
* STORAGE DEFINITION
*
```

```
          STORAGE     S1,6/S2,4/S3,14/S4,1/S5,2/S6,6/S7,13/S8,13
          STORAGE     S9,3/S10,9/S11,7/S12,2/S13,2/S14,7/S15,2
          STORAGE     S16,12/S17,27/S18,90/S19,8 //BACKBONE
          STORAGE     S31,2/S32,7/S33,2          //TEST CELL LOOP
          STORAGE     S34,1/S35,6/S36,4          //TEST CELL QUEUES
          STORAGE     S37,7/S38-S40,1            //RETORQUE STATIONS
          STORAGE     S(RECR1O),1/S(RECR1),50/S(BACKUP),3
*
* Define Facilities.
*
 EXIT1 EQU        77,F,XF              EXIT PATH CLEAR
 EXIT2 EQU        78,F,XF              EXIT PATH CLEAR
 ENTR1 EQU        79,F,XF              EXIT PATH CLEAR
 ENTR2 EQU        80,F,XF              EXIT PATH CLEAR
 CTEST1 EQU       81,F                 TEST CELL
 CTEST2 EQU       82,F                 TEST CELL
 CTEST3 EQU       83,F                 TEST CELL
 CTEST4 EQU       84,F                 TEST CELL
 CTEST5 EQU       85,F                 TEST CELL
 CTEST6 EQU       86,F                 TEST CELL
 CTEST7 EQU       87,F                 TEST CELL
 CTEST8 EQU       88,F                 TEST CELL
 CTEST9 EQU       89,F                 TEST CELL
 CTEST10 EQU      90,F                 TEST CELL
 CTEST11 EQU      91,F                 TEST CELL
 CTEST12 EQU      92,F                 TEST CELL
 CTEST13 EQU      93,F                 TEST CELL
 CTEST14 EQU      94,F                 TEST CELL
 CTEST15 EQU      95,F                 TEST CELL
 CTEST16 EQU      96,F                 TEST CELL
 CTEST17 EQU      97,F                 TEST CELL
 CTEST18 EQU      98,F                 TEST CELL
 CMCHT EQU        100(18),F            ASSOC. CELL RUN TIME
*
 CSPED EQU        1,XL                 //CONV. SPEED
*
* PROCESS CODES & GROUPS
*
 RCRQ1    SYN        1                 //RECIRCULATE
 CLTEST   SYN        2                 //TEST CELL
 RTORQ    SYN        3                 //RETORQUE
 AUDIT    SYN        4                 //AUDIT
 OFFLD    SYN        5                 //OFFLOAD
 REPAIRS  SYN        6                 //REPAIRS
 CSTRIM   SYN        7                 //CUSTOM TRIM
 FNTRIM   SYN        8                 //FINAL TRIM
 PNTSYS   SYN        9                 //PAINT SYSTEM
 BBTRIM   SYN        10                //BLUE BIRD TRIM
*
 GRCRQ1   EQU        1,G
 GCLTEST  EQU        2,G
 GRTORQ   EQU        3,G
 GAUDIT   EQU        4,G
 GOFFLD   EQU        5,G
 GREPAIRS EQU        6,G
 GCSTRIM  EQU        7,G
 GFNTRIM  EQU        8,G
 GPNTSYS  EQU        9,G
 GBBTRIM  EQU        10,G
 GFNTRBB  EQU        11,G
 GFNTRPT  EQU        12,G
 GPNTSBB  EQU        13,G
 TRKGRID  EQU        51(50),XH,G        //TRUCK GRID
 TGRIDS   EQU        1,Q                //NUMBER OF OPEN GRIDS
*
* MATRIX DEFINITIONS
*
 CSECT    MATRIX     ML,50,50           //CONV. SECTION TRAVEL DISTANCE
 PROD     MATRIX     ML,100,20          //PRODUCTION MATRIX
```

```
TECHBD   MATRIX     ML,100,10           //TECHNICIAN BREAKDOWN
TCHASN   MATRIX     MX,200,10           //TECHNICIAN ASSIGNMENTS
CUSTRM   MATRIX     ML,20,20            //CUSTOM TRIM CLASSxSTATION
FNLTRM   MATRIX     ML,20,20            //FINAL TRIM CLASSxSTATION
FLOWRT   MATRIX     ML,6,5              //FLOW RATE COLLECTION
KEYQUE   MATRIX     ML,6,5              //CRITICAL QUEUE COLLECTION
*
*********************************************************
*
* File Definitions for files used in every scenario
*
INFILE   FILEDEF    'INPUT.DAT'       //General Input Parmaters
LAYOUT   FILEDEF    'LAYOUT.DAT'      //Layout Definition
INVEN    FILEDEF    'INV.DAT'         //Beginning Inventory
ALINEUP  FILEDEF    'LINEUP.DAT'      //Assembly Lineup
DPT568   FILEDEF    'CYL568.DAT'      //568 Cycle Times
DPT569   FILEDEF    'CYL569.DAT'      //569 Cycle Times
DPT570   FILEDEF    'CYL570.DAT'      //570 Cycle Times
DPT572   FILEDEF    'CYL572.DAT'      //572 Cycle Times
DPT571   FILEDEF    'PAINT.DAT'       //571 Paint Parameters
TECHS    FILEDEF    'TECHS.DAT'       //Technician Assignments
OPDAT    FILEDEF    'OPERAT.DAT'      //Operation Schedules
CSTRM    FILEDEF    'CTRIM.DAT'       //Custom Trim Line
FNTRM    FILEDEF    'FTRIM.DAT'       //Final Trim Line
DWNTIM   FILEDEF    'DWNTIM.DAT'      //Downtime Scenarios
ATF      FILEDEF    'TTPS1.ATF'       //ttps Trace File
OUT      FILEDEF    'OUTPUT.DAT'      //Output Report
TSUM     FILEDEF    'TESTSUM.DAT',APPEND //ACCUMULATION TEST SUMMARY
*
* INITIALIZATION
*
        INITIAL    XL$CSPED,60.0       CONV. SPEED
        INITIAL    ML$CSECT(1,1),.84/ML$CSECT(2,1),.54
        INITIAL    ML$CSECT(3,1),1.75/ML$CSECT(4,1),.14
        INITIAL    ML$CSECT(5,1),.32/ML$CSECT(6,1),.75
        INITIAL    ML$CSECT(7,1),1.36/ML$CSECT(8,1),1.59
        INITIAL    ML$CSECT(9,1),.39/ML$CSECT(10,1),1.14
        INITIAL    ML$CSECT(11,1),.90/ML$CSECT(12,1),.30
        INITIAL    ML$CSECT(13,1),.26/ML$CSECT(14,1),.91
        INITIAL    ML$CSECT(15,1),.27/ML$CSECT(16,1),3.02
        INITIAL    ML$CSECT(17,1),3.35/ML$CSECT(18,1),11.22
        INITIAL    ML$CSECT(19,1),1.04
        INITIAL    ML$CSECT(31,1),.26/ML$CSECT(32,1),.84
        INITIAL    ML$CSECT(33,1),.18/ML$CSECT(38,1),.28
        INITIAL    ML$CSECT(39,1),.50
*
        INITIAL    ML$CSECT(1,11),.12/ML$CSECT(2,11),.33
        INITIAL    ML$CSECT(3,11),.35/ML$CSECT(4,11),.12
        INITIAL    ML$CSECT(9,11),.07
        INITIAL    ML$CSECT(11,11),.08/ML$CSECT(12,11),.07
        INITIAL    ML$CSECT(13,11),.12
        INITIAL    ML$CSECT(15,11),.15
        INITIAL    ML$CSECT(17,11),.14
        INITIAL    ML$CSECT(32,11),.13/ML$CSECT(33,11),.13
        INITIAL    ML$CSECT(38,11),.42/ML$CSECT(39,11),.21
*
        INITIAL    ML$CSECT(1,12),.33/ML$CSECT(2,12),.14
        INITIAL    ML$CSECT(3,12),.22/ML$CSECT(4,12),.12
        INITIAL    ML$CSECT(9,12),.08
        INITIAL    ML$CSECT(11,12),.08/ML$CSECT(12,12),.15
        INITIAL    ML$CSECT(13,12),.13
        INITIAL    ML$CSECT(15,12),.22
        INITIAL    ML$CSECT(17,12),.14
        INITIAL    ML$CSECT(32,12),.15/ML$CSECT(33,12),.14
*
        INITIAL    ML$CSECT(1,22),16.92/ML$CSECT(1,23),29.92
        INITIAL    ML$CSECT(1,24),43.03/ML$CSECT(1,25),54.66
        INITIAL    ML$CSECT(1,26),41.55/ML$CSECT(1,27),28.55
        INITIAL    ML$CSECT(1,28),39.45/ML$CSECT(1,32),38
```

```
      INITIAL   ML$CSECT(1,33),58.82/ML$CSECT(1,34),37.24
      INITIAL   ML$CSECT(1,35),165.21/ML$CSECT(1,36),82.92
*
         LET       &BBMRG(1)=1             //BB MERGE
         LET       &BBMRG(2)=1             //BB MERGE
         LET       &BBMRG(3)=1             //BB MERGE
         LET       &BBMRG(4)=1             //BB MERGE
         LET       &BBMRG(9)=1             //BB MERGE
         LET       &BBMRG(11)=1            //BB MERGE
         LET       &BBMRG(12)=1            //BB MERGE
         LET       &BBMRG(13)=1            //BB MERGE
         LET       &BBMRG(15)=1            //BB MERGE
         LET       &BBMRG(17)=1            //BB MERGE
         LET       &BBMRG(32)=1            //BB MERGE
         LET       &BBMRG(33)=1            //BB MERGE
         LET       &BBMRG(35)=1            //BB MERGE
         LET       &BBMRG(37)=1            //BB MERGE
******************************************************
******************************************************
*
* BEGINNING OF BLOCK STATEMENTS
*
******************************************************
******************************************************
*
* CODE ADDITIONS FOR BLOCK AND JHOOK LINE
*
******************************************************
         REAL      &APATH(100)                  //ASSEMBLY PATH DISTANCES
         LET       &APATH(1)=11.71
         LET       &APATH(2)=22.06
         LET       &APATH(3)=8
         LET       &APATH(4)=8
         LET       &APATH(5)=8
         LET       &APATH(6)=8
         LET       &APATH(7)=333.98
         LET       &APATH(8)=18.28
         LET       &APATH(9)=423.15
         STORAGE   S201,1/S202,2/S203-S206,1/S207,41/S208,2/S209,52
         REAL      &FSP                         //FAST CONV. SPEED
         LET       &FSP=60.0
         REAL      &SSP                         //SLOW CONV. SPEED
         REAL      &OSP                         //OLD SPEED
         INTEGER   &INV572(100)                 //FINISHED ENGINE INV
         INTEGER   &INPROC                      //IN PROCESS ENGINES FROM JHOOK ON
         INTEGER   &ESHPD(100)                  //ENGINES SHIPPED
         INTEGER   &TRKLD(1000)                  //TRUCK LOAD
         VCHAR*12  &PARTNO(100)                 //ENGINE PART #
         VCHAR*30  &DUM,&DUM1,&DUM2,&DUM3,&DUM4 //INPUT CHARACTERS
         INTEGER   &FINORD                      //FINISH ORDER OF ENGINES
         INTEGER   &SDAY
 JHOOK   SYN       200                          //J-HOOK OFFSET
 ASMLINE EQU       201(9),L,S                   //ASSEMBLY LINE ZONES
 ASMPOS  EQU       200(9),F                     //ASSEMBLY POSITION
 ASMUL   EQU       210,F,C,L                    //ASSEMBLY UNLOAD
 JHLEAK  EQU       211,F,C,L                    //564 LEAK TEST
 JHPRO   EQU       212(48),F                    //ASSEMBLY CHAIN PROCESS
 PRO569  EQU       51(20),F                     //569 PROCESSES
 PRO571  EQU       125(15),F                    //571 PROCESSES
* FACILITIES 300-400 RESERVED FOR TECHNICIANS
 ASMLD   EQU       200,C,L                      //ASSEMBLY LOAD
 SSHRT   EQU       201,C                        //SHIPMENT SHORTAGES
 INPRO   EQU       202,C                        //PROCESS CHAIN
 NOTCH   EQU       203,C                        //NO TECHNICIANS
 HOLD    EQU       204,C                        //HOLD CHAIN
 BBITQ   EQU       205,C                        //BLUEBIRD INPUT TRUCKER
 FLRPNTQ EQU       206,C                        //FLOOR PAINT QUEUE
 BBOTQ   EQU       207,C                        //BLUEBIRD OUTPUT TRUCKER
 BBTRMQ  EQU       208,C                        //BLUEBIRD TRIM QUEUE
```

```
ACELLS   EQU        209,C                              //ACTIVE CELLS
FINV     EQU        215,C                              //FINISHED INVENTORY
TRKHLD   EQU        216,C                              //TRUCK HOLD
STPSHP   EQU        217,C                              //STOP SHIPMENT
PDLAY    EQU        218,C                              //DELAYED JOB
MATCH    EQU        218,L                              //MATCH ONE DELY @ TIME
DINIT    EQU        219,L                              //INDICATES INITILIZATION DONE
SWING    EQU        219,C                              //HOLD POSITION FOR SWING TECHS
ONTRK    EQU        220,C                              //ENGINES ON TRUCK
FLRSTGQ  EQU        221,C                              //FLOOR STAGE QUEUE
TCHNS    SYN        300                                //TECHNICIAN OFFSET
*
SHIPS    MATRIX     MX,800,5                           //SHIPMENT SCHEDULE
ESYSPRF  MATRIX     ML,10,5                            //MISC. SYSTEM PERFORMANCE
DSHIPS   MATRIX     MH,100,21                          //DAILY ENGINE SHIPMENTS
TSHIPS   MATRIX     MH,100,21                          //DAILY TRUCK BY CUSTOMERS
SDLAY    MATRIX     ML,100,5                           //SHIPMENT DELAYS
SEQVAR   MATRIX     ML,100,5                           //SEQUENCE VARIATION
PROTIME  MATRIX     ML,100,5                           //PROCESS TIME TO WH
WHSETIM  MATRIX     ML,100,5                           //WAREHOUSE TIME
GRIDTIM  MATRIX     ML,100,5                           //GRID TIME BY CUSTOMER
TRKLDTIM MATRIX     ML,100,5                           //TRUCK LOAD TIME BY CUSTOMER
FINSEQ   FVARIABLE  PF(SSEQN)-&FINORD                  //ENGINE SEQ VS. FINISH ORDER
*
* TTPS Project Inputs
*
         REAL       &ASMMAX                            //ASSEMBLY MAXIMUM
         REAL       &JHSPD(3)                          //JHOOK SPEED/SHIFT
         REAL       &PERF(10)                          //TECH. PERFORMANCE/MODULE
         REAL       &JHKUL                             //J-HOOK UNLOAD
         REAL       &LEAKRJ(2)                         //LEAK TEST REJECT% (1 & 2)
         REAL       &LEAKTST(2)                        //LEAK TEST TIMES (1 & 2)
         REAL       &LEAKRPR(2)                        //LEAK REPAIR TIMES (1 & 2)
         REAL       &HRPRTIM                           //HEAVY REPAIR TIME
         REAL       &LRPRTIM                           //LIGHT REPAIR TIME
         REAL       &CRPRTIM                           //CELL REPAIR TIME
         REAL       &LRPRRJ                            //LIGHT REPAIR REJECT%
         REAL       &CRPRRJ                            //CELL REPAIR REJECT%
         REAL       &SIN568(100)                       //STARTING INVENTORY IN 568
         REAL       &SIN569(100)                       //STARTING INVENTORY IN 569
         REAL       &SIN570(100)                       //STARTING INVENTORY IN 570
         REAL       &SIN572(100)                       //STARTING INVENTORY IN 572
         REAL       &CTRIM(100)                        //568 COMPRESSOR TRIM
         REAL       &BTRIM(100)                        //568 BLUEBIRD TRIM
         REAL       &RPASS(10)                         //REAL DATA INPUT VARIABLE
         REAL       &COPTN(100)                        //COMPRESSOR OPTIONS
         REAL       &CTEST(100)                        //TEST CELL CYCLE TIME
         REAL       &HOOK(100)                         //TEST CELL HOOK TIME
         REAL       &UNHK(100)                         //TEST CELL UNHOOK
         REAL       &RHOOK(100)                        //RTIME FOR TEST CELL HOOK
         REAL       &RTORK(100)                        //RETORQUE TIME/ENGINE
         REAL       &TRJT1(100)                        //1ST TEST REJECT%
         REAL       &TRJT2(100)                        //2ND TEST REJECT%
         REAL       &BLOWO(100)                        //PAINT MASK & BLOW-OFF
         REAL       &MASK(100)                         //MASK TIME
         REAL       &PCOAT(100)                        //PRIME COAT CYCLE TIME
         REAL       &TCOAT(100)                        //TOP COAT CYCLE TIME
         REAL       &RECTRKS                           //# REC'D TRUCKS/DAY
         REAL       &SHPTIM(10)                        //572 CYCLE TIMES
         REAL       &FLASH                             //PAINT FLASH TIME/STOP
         REAL       &COOL                              //PAINT COOL TIME/STOP
         REAL       &PNTFSP                            //PAINT DELIVERY SPEED
         REAL       &PNTSSP                            //PAINT PROCESS CHAIN SPEED
         REAL       &INSPCT                            //INSPECT TIME
         REAL       &CYADJ                             //CYCLE TIME ADJUST
         REAL       &EPROD(10)                         //ENGINE PRODUCTION BY MODULE
         INTEGER    &LBCTJHK                           //# J-HOOK CARRIERS
         INTEGER    &NOMDLS                            //# ENGINE MODELS
         INTEGER    &TCRTE(100)                        //TEST CELL ROUTING(0=ANY)
```

```
          INTEGER     &DOCK                           //# SHIPPING/REC DOCKS
          INTEGER     &PCMAX                          //MAX. #LOADS ON PROCESS CHAIN
          INTEGER     &SVAR                           //SAME STATION VARIABLE
          INTEGER     &TLAST                          //LAST TECHNICIAN
          INTEGER     &TCC                            //TECHNICIAN COUNT
          INTEGER     &TECHC(300)                     //TECHNICIAN COUNT
          INTEGER     &GBCNT(10)                      //GLOBAL COUNT
          INTEGER     &OPCNT(100)                     //ENGINE OPTION COUNT
          INTEGER     &ENGC1(100)                     //ENGINE COUNT (1ST TEST)
          INTEGER     &ENGC2(100)                     //ENGINE COUNT(2ND TEST)
          INTEGER     &MD                             //MODULE POINTER
          INTEGER     &SF                             //SHIFT POINTER
          INTEGER     &LOC(6)                         //LOCATION PARAMETER
          INTEGER     &WDAYS(10)                      //#WORKS DAYS/WEEK/MODULE
          INTEGER     &WEEK                           //# PRODUCTION DAYS/WEEK
          INTEGER     &CSTLS                          //CUSTOM TRIM LAST STATION
          INTEGER     &FNTLS                          //FINAL TRIM LAST STATION
          INTEGER     &ECLASI(100)                    //ENGINE CLASS INTEGER BY ENGINE
          INTEGER     &BBLIM                          //BACKBONE LIMIT
          INTEGER     &ATHEAD                         //# AT HEAD OF ATTIC
          VCHAR*20    &CUSTMR(100)                    //CUSTOMER BY ENGINE
          VCHAR*20    &CUSTID(100)                    //CUSTOMER ID (UNIQUE)
          VCHAR*20    &PDATE                          //PREVIOUS SHIP DATE
          VCHAR*20    &PTRUCK                         //PREVIOUS TRUCK #
          VCHAR*20    &STECH(10)                      //SHIPPING TECHNICIANS
          VCHAR*10    &MODID(10)                      //MODULE ID NAME
          VCHAR*20    &TCHNM(100)                     //TECHNICIAN NAME
          VCHAR*10    &CPASS(10)                      //CHARACTER VALUE PASS
          VCHAR*10    &SNAME(300)                     //STATION NAME
          VCHAR*10    &ECLASC(20)                     //ENGINE CLASS CHAR-DEFINITION
*
* VARIABLE DEFINITION
*
 CTRVL FVARIABLE  ML$CSECT(1,PF$CVSEC)/XL$CSPED     CONV. TRAVEL
     1 BVARIABLE  FS(81)*LS(41)
     2 BVARIABLE  FS(82)*LS(42)
     3 BVARIABLE  FS(83)*LS(43)
     4 BVARIABLE  FS(84)*LS(44)
     5 BVARIABLE  FS(85)*LS(45)
     6 BVARIABLE  FS(86)*LS(46)
     7 BVARIABLE  FS(87)*LS(47)
     8 BVARIABLE  FS(88)*LS(48)
     9 BVARIABLE  FS(89)*LS(49)
    10 BVARIABLE  FS(90)*LS(50)
    11 BVARIABLE  FS(91)*LS(51)
    12 BVARIABLE  FS(92)*LS(52)
    13 BVARIABLE  FS(93)*LS(53)
    14 BVARIABLE  FS(94)*LS(54)
    15 BVARIABLE  FS(95)*LS(55)
    16 BVARIABLE  FS(96)*LS(56)
    17 BVARIABLE  FS(97)*LS(57)
    18 BVARIABLE  FS(98)*LS(58)
 CELB1 BVARIABLE  (BV1=1)OR(BV2=1)OR(BV3=1)OR(BV4=1)OR(BV5=1)OR(BV6=1)
 TOSTO BVARIABLE  PF(RJCT)=1
 TORPR BVARIABLE  (PF$RJCT=1)AND(SNF32)           //TO REPAIR
 RQRWK BVARIABLE  (FNU13)AND(SNF14)AND(Q13=0)     //RETORQUE REWORK
 OPNRT BVARIABLE  PF(PTR)'GE'11*PF(PTR)'LE'13
 PTWAY BVARIABLE  XF79'E'PF1+XF80'E'PF1
 BATLD BVARIABLE  SE(SPO15)+S(SPNT0)'L'2
 SFTCO BVARIABLE  (PF$SHFT=&SF)AND(PF$MOD=&MD)      //SHIFT CHANGEOVER
 DLAY1  BVARIABLE (PF$CLOC=&LOC(1))OR(PF$CLOC=&LOC(2))OR(PF$CLOC=&LOC(3))OR_
                  (PF$CLOC=&LOC(4))OR(PF$CLOC=&LOC(5))OR(PF$CLOC=&LOC(6))
 PBATCH   BVARIABLE  (CH(SPNT3)>=2)AND(SE(SPNT4))AND(SE(SPNT5))
 RTQUL    BVARIABLE  (LS38)AND(LS39)
 ENG105   BVARIABLE  (&ECLASI(PF$ENGINE)=10)OR_
                     (&ECLASI(PF$ENGINE)=11)OR_
                     (&ECLASI(PF$ENGINE)=12)OR_
                     (&ECLASI(PF$ENGINE)=13)
 RTQBYP   BVARIABLE  (PF(PCODE)=PNTSYS)OR(PF(ENGINE)=0)
```

```
 BBMTR     BVARIABLE  (LC260)AND(SNF(SPO))AND(SE271) //BLUEBIRD METER
 NOBKUP    BVARIABLE  (SF$BACKUP)AND(CH$BACKUP>0)AND(SNF$RECR1) //BACKUP CONDITIONBBD0100
*
* FUNCTION DEFINITIONS
*
     2 FUNCTION    PF(CVSEC),D2           //CHAIN DIRECT
34,21/36,22
*
     3 FUNCTION    PF$DELRT,D2            TEST CELL ENTRANCE PATH
6,33/18,32
*
     4 FUNCTION    PF$DELRT,D18         "IN" PATH TRAVEL TIME
1,.49/2,.42/3,.36/4,.28/5,.22/6,.14/7,.5/8,.42/9,.36/10,.29
11,.23/12,.15/13,.50/14,.42/15,.36/16,.29/17,.23/18,.15
*
     5 FUNCTION    PF$DELRT,D18         "OUT" PATH TRAVEL TIME
1,.06/2,.14/3,.2/4,.27/5,.32/6,.40/7,.07/8,.15/9,.2/10,.28
11,.33/12,.41/13,.07/14,.14/15,.20/16,.28/17,.33/18,.41
*
     7 FUNCTION    PF$DELRT,D2            TEST CELL ENTRANCE PATH
6,80/18,79
*
    11 FUNCTION    PF$DELRT,D2            TEST CELL EXIT PATH
6,78/18,77
*
    12 FUNCTION    PF(PCT),D18            //INITIAL %
1,100/2,50/3,33/4,25/5,20/6,16/7,14/8,12/9,11/10,10/11,9/12,8/14,7/17,6
20,5/25,4/33,3/50,2
*
    13 FUNCTION    PF(PCT),D39            //SECONDARY %
12,0/13,200/14,0/15,200/16,400/17,0/18,199/19,399/20,0/21,133/22,199/23,333
24,499/25,0/26,125/27,143/28,200/29,250/30,332/31,500/32,1000/33,0/34,105
35,111/36,125/37,143/38,167/39,181/40,199/41,221/42,249/43,333/44,399/45,499
46,667/47,999/48,1249/49,2499/50,0
*
 TLOC1     FUNCTION   PF(LCTR),L6
,PRO110/,PRO120/,PRO130/,PRO140/,PRO150/,PRO160
*
 TLOC2     FUNCTION   PF(LCTR),M6         //TECHNICIAN LOCATION #2
,PF$LOC1/,PF$LOC2/,PF$LOC3/,PF$LOC4/,PF$LOC5/,PF$LOC6
*
 PROCQ     FUNCTION   PF(LOC1),E6         //STORAGE DIRECT
1,S(18)/2,S(RECR1)/3,S(TSTLCOUT)/4,S(CSTRMCNT)/5,S(FNTRMCNT)/6,S(PAINTCNT)
*
          LET        &PCLR(1)='F3'
          LET        &PCLR(2)='F7'
          LET        &PCLR(3)='F4'
          LET        &PCLR(4)='F11'
          LET        &PCLR(5)='F2'
          LET        &PCLR(6)='F9'
*
          LET        &ITOCHAR(1)='1'
          LET        &ITOCHAR(2)='2'
          LET        &ITOCHAR(3)='3'
          LET        &ITOCHAR(4)='4'
          LET        &ITOCHAR(5)='5'
          LET        &ITOCHAR(6)='6'
          LET        &ITOCHAR(7)='7'
          LET        &ITOCHAR(8)='8'
          LET        &ITOCHAR(9)='9'
          LET        &ITOCHAR(10)='10'
          LET        &ITOCHAR(11)='11'
          LET        &ITOCHAR(12)='12'
          LET        &ITOCHAR(13)='13'
          LET        &ITOCHAR(14)='14'
          LET        &ITOCHAR(15)='15'
          LET        &ITOCHAR(16)='16'
          LET        &ITOCHAR(17)='17'
          LET        &ITOCHAR(18)='18'
```

```
         LET        &ITOCHAR(19)='19'
         LET        &ITOCHAR(20)='20'
         LET        &ITOCHAR(21)='21'
         LET        &ITOCHAR(22)='22'
         LET        &ITOCHAR(23)='23'
         LET        &ITOCHAR(24)='24'
*
* CONTROL STATEMENT PLUGS-INS
*
         INSERT     <CNTLDEF1.GPS>
         INSERT     <PROOF.MAC>         //ANIMATION MACROS
*
* TECHNICIAN TO ELEMENT MATCH-UP MACRO
*
 FNDTCH  STARTMACRO #A
         ALTERUCH E APOOL,1,CLOC$PF,PF$CLOC,#A,PF$CLOC,PRO100     //PASS LOC
         ALTERUCH E APOOL,1,CYCLE$PL,PL$CYCLE,CLOC$PF,PF$CLOC  //PASS CYCLE
         BLET       PL(CMPEST)=PL$CYCLE+AC1          //ESTIMATE COMPLETION
         UNLINK     APOOL,TCH100,1,CLOC$PF,PF$CLOC   //GET TECH
         ENDMACRO
*
*********************************************************************
* Simulation Timer Module
* Written by G. Rehn
* 6/29/98
* Version 01
*-------------------------------------------------------------
* VARIABLES:      PF(CTR)=       Segment Pointer
*             PF(MOD)=       Module #
*             PF(OPER1)=     1st Operator in Range
*             PF(OPERL)=     Last Operator in Range
*                 PF(INDX)=      INDEX POINTER
*                 PF(JNDX)=      INDEX POINTER
*                 &SD=           # Simulation Days
*             &MSHIFT(10)=   Initial Shift/Module
*                 &MODID(10)=    Module Identifier
*             &OPXID(200)=   Operator ID Index
*             &COPR(200)=    Operator Color (Current)/Index
*                 &SALOW =       Start-up Allowance
*                 &CALOW =       Clean-up Allowance
*                 &EFMIN(10) =   # Effective Mins/ Day
*                 &OPHRS(10) =   # Total Hours
*                 &OPSFT(10) =   # Operating Shifts/Module
*                 &OPAS(100)=    Input Translation from Excel
*                 &DFTOP(100)=   Default Operation Description
*                 &ACNOOP(10) =  Accumulated Out of Operation Time
*                 &CLKS =     Simulation Start Time
*                 &AMPM(2)=   AM/PM START INDICATOR
*                 LS(MORN)=   MORNING SWITCH LC-MORNING/LS-AFTERNOON
*                 &PE= PAINT PURGE START
*                 &PS= PAINT START-UP TIME
*                 Matrix HPS=    Hours/Shift (Halfword)
*              Row=   Module
*              Cols 1-96=     Action in 15 Min. Increments
*                 Matrix TCH1,TCHL = First & Last Technicians (Operators)
*                      Row=      Module
*                      Col=      Shift (1,2,3)
*********************************************************************
* TIMER CONTROL STATEMENTS
*************************************************************
          INTEGER    &OPXID(200)             //Operator ID Index
          VCHAR*9    &COPR(200)              //Operator Color (Current)/Index
          INTEGER    &OPSFT(10)              //# Operating Shifts/Module
          INTEGER    &MSHIFT(10)             //Initial Shift / Module
          REAL       &EFMIN(10)              //# Effective Mins/ Day
          REAL       &OPHRS(10)              //# Total Hours
          VCHAR*2    &OPAS(100)              //Input Translation from Excel
          INTEGER    &DFTOP(100)             //Default Operation
          REAL       &ACNOOP(10)             //Accumulated Out of Operation Time
```

```
         VCHAR*2    &AMPM(2)                       //AM/PM START INDICATOR
         INTEGER    &SD                        //# Simulation Days
         REAL       &SALOW                     //Start-up Allowance
         REAL       &CALOW                     //Clean-up Allowance
         REAL       &CLKS                      //Simulation Start Time
         REAL       &CLKPTH                    //Clock Path
         INTEGER    &PS,&PE                    //PAINT PURGE/START TIME
*
 TMDIR   FUNCTION   MH(HPS,PF(MOD),PF(CTR)),D5
-2,TMRBEG/-1,TMREND/3,TMRADV/15,TMRBRK/99,TMRSWG
*
 PCNVRT  FUNCTION   PF(CTR),E3                 //POINTER CONVERT
0,96/96,PFSCTR/97,1
*
 ADJDL   FVARIABLE  MP$WAIT$PL-(&ACNOOP(PF$MOD)-PL$ACBRK)  //DELAY ADJUSTMENT
 DISFT   BVARIABLE  (&DFTOP(PF$CTR)>=0)AND(&DFTOP(PF$CTR)<=3)
 TISFT   BVARIABLE  (MH(HPS,&I,PF$CTR)>0)AND(MH(HPS,&I,PF$CTR)<=3)
 LMITS   BVARIABLE  (PF$CLOC=&SVAR)AND(PF$OPNUM>0)  //SAME STA. SEARCH
*
 HPS     MATRIX     MH,10,100                  //HOURS PER SHIFT DESCRIPTION
 TCH1    MATRIX     MH,10,3                    //FIRST TECHNICIAN/MODULE
 TCHL    MATRIX     MH,10,3                    //LAST TECHNICIAN/MODULE
*PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
*
* VARIABLE DEFINITIONS (SEE EQU'S)
* BOOLEAN VARIABLE DEFINITIONS (ALL DIRECTLY ADDRESSED)
*
*
* FUNCTION DEFINITIONS (SEE EQU'S)
*
* MATRIX DEFINITIONS
*
  LINEUP MATRIX     MX,600,5                   //ASSEMBLY LINEUP
  PROD   MATRIX     ML,100,10                  //PRODUCTION ATTAINED
*
* VARIABLE DEFINITION
*
 ANYOP   FVARIABLE  PF$LOC1+PF$LOC2+PF$LOC3+PF$LOC4+PF$LOC5+PF$LOC6
 TBULT   FVARIABLE  (ML(TECHBD,&I,&J)/ML(TECHBD,&I,7))*FRV(&I)/10.0
*--------------------------------
* DATA READ LOGIC - INPUT SCENARIO
*--------------------------------
         GENERATE   ,,,1,,32PF,7PL           //SEED XACT
         BGETLIST   FILE=INFILE,&TESTID       //TEST #
         BGETLIST   FILE=INFILE,&TESTDSCR //DESCRIPTION
         BGETLIST   FILE=INFILE,&DUM         //SKIP LINE
         BGETLIST   FILE=INFILE,&PRODVOL(1),&JHSPD(1) //PRODUCTION-JHSPD/SHIFT
         BGETLIST   FILE=INFILE,&PRODVOL(2),&JHSPD(2) //PRODUCTION-JHSPD/SHIFT
         BGETLIST   FILE=INFILE,&PRODVOL(3),&JHSPD(3) //PRODUCTION-JHSPD/SHIFT
         BLET       &ASMMAX=&JHSPD(1)        //ASSEMBLY MAXIMUM
         TEST G     &JHSPD(2),&ASMMAX,*+2
         BLET       &ASMMAX=&JHSPD(2)         //ASSEMBLY MAXIMUM
         TEST G     &JHSPD(3),&ASMMAX,*+2
         BLET       &ASMMAX=&JHSPD(3)        //ASSEMBLY MAXIMUM
         BGETLIST   FILE=INFILE,&RUNDAYS     //#SIMULATION DAYS
         BGETLIST   FILE=INFILE,&DUM         //SKIP
         BGETLIST   FILE=INFILE,&CLKS        //STARTING SIMULATION TIME
         BGETLIST   FILE=INFILE,&SALOW       //START-UP ALLOWANCE
         BGETLIST   FILE=INFILE,&CALOW       //CLEAN-UP ALLOWANCE
         BGETLIST   FILE=INFILE,&LBCTMAIN //#LOAD BARS IN MAIN SYSTEM
         BGETLIST   FILE=INFILE,&LBCTJHK   //#LOAD BARS IN J-HOOK SYSTEM
         BGETLIST   FILE=INFILE,&MAX         //MAX# IN TEST CELL LOOP
         BGETLIST   FILE=INFILE,&CHOK        //EXIT TEST LIMIT
         BSTORAGE   S(TSTLCOUT),&CHOK        //SET STORAGE
         BGETLIST   FILE=INFILE,&CHOK        //MAX. CUSTOM TRIM LIMIT
         BSTORAGE   S(CSTRMCNT),&CHOK        //SET STORAGE
         BGETLIST   FILE=INFILE,&CHOK        //MAX. FINAL TRIM LIMIT
         BSTORAGE   S(FNTRMCNT),&CHOK        //SET STORAGE
         BGETLIST   FILE=INFILE,&CHOK        //MAX. PAINT LIMIT
```

```
          BSTORAGE   S(PAINTCNT),&CHOK      //SET STORAGE
          BGETLIST   FILE=INFILE,&BBLIM     //MAX. BB LIMIT
          BGETLIST   FILE=INFILE,&JHKCO     //JHOOK CHANGEOVER
          BGETLIST   FILE=INFILE,&JHKUL     //UNLOAD JHOOK LINE
          BGETLIST   FILE=INFILE,&DUM       //SKIP LINE
          BGETLIST   FILE=INFILE,&LEAKTST(1),&LEAKTST(2) //1ST&2ND LEAK TEST TIME
          BGETLIST   FILE=INFILE,&LEAKRJ(1),&LEAKRJ(2)   //1ST&2ND LEAK REJECT%
          BGETLIST   FILE=INFILE,&LEAKRPR(1),&LEAKRPR(2) //1ST&2ND LEAK REPAIR TIME
          BGETLIST   FILE=INFILE,&DUM       //SKIP LINE
          BGETLIST   FILE=INFILE,&HRPRTIM  //HEAVY 569 REPAIR
          BGETLIST   FILE=INFILE,&LRPRRJ,&LRPRTIM //569 LIGHT REJECT & REPAIR TIME
          BGETLIST   FILE=INFILE,&CRPRRJ,&CRPRTIM //569 CELL REJECT & REPAIR TIME
          BGETLIST   FILE=INFILE,&INSPCT    //571 INSPECTION
          BCLOSE     INFILE
*-------------------
* CUSTOM TRIM
*-------------------
          BGETLIST   FILE=CSTRM,(&ECLASC(&J),&J=1,14)
          BLET       &I=0
 CUS000   BLET       &I=&I+1               //NEXT STATION
          BGETLIST   FILE=CSTRM,END=CUS999,(ML(CUSTRM,&J,&I),&J=1,14)
          BLET       &CSTLS=&I             //SAVE FOR LAST STATIO
          TRANSFER   ,CUS000
 CUS999   BCLOSE     CSTRM
*-------------------
* FINAL TRIM
*-------------------
          BGETLIST   FILE=FNTRM,(&EFAM(&J),&J=1,14)
          BGETLIST   FILE=FNTRM,&DUM
          BLET       &I=0
 FNL000   BLET       &I=&I+1               //NEXT STATION
          BGETLIST   FILE=FNTRM,END=FNL999,&DUM,(ML(FNLTRM,&J,&I),&J=1,14)
          TEST NE    &DUM,'Kit',FNL000
          BLET       &FNTLS=&I             //SAVE FOR LAST STATIO
          TRANSFER   ,FNL000
 FNL999   BCLOSE     FNTRM
*-------------------
* STARTING INVENTORY
*-------------------
          BGETLIST   FILE=INVEN,&DUM       //SKIP LINE
          BLET       &I=0                  //ZERO OUT
 INV000   BLET       &I=&I+1               //BUMP
          BGETLIST   FILE=INVEN,END=INV100,&DUM,&DUM2,&J,&SIN568(&I),_
                     &SIN569(&I),&SIN570(&I),&SIN572(&I),&J,_
                     &CUSTMR(&I),&DUM1,&ECLR(&I)
          TEST NE    &CUSTMR(&I),&PRVCUS,INV005
          BLET       PF(LOC1)=&I
          SPLIT      1,KEY000
          BLET       &PRVCUS=&CUSTMR(&I)
          BLET       PF(LOC1)=0
 INV005   BLET       PF(LCTR)=LEN(&DUM) //LAST CHAR IN &DUM
 INV010   TEST NE    SSG(&DUM,PF$LCTR,1),'-',INV020 //FIND '-'
          LOOP       LCTR$PF,INV010
 INV020   BLET       PF(LCTR)=PF(LCTR)+1
          BLET       &PARTNO(&I)=SSG(&DUM,PF$LCTR) //TRUNCATE
          BLET       PF(LCTR)=20
 INV030   TEST E     &DUM2,&ECLASC(PF$LCTR),INV040 //MATCH?
          BLET       &ECLASI(&I)=PF$LCTR          //SAVE PTR VALUE
          TRANSFER   ,INV000
 INV040   LOOP       LCTR$PF,INV030
          TRANSFER   ,INV000
 INV100   BLET       &I=&I-1
          BLET       &NOMDLS=&I              //SAVE # MODELS
          BCLOSE     INVEN
*
          BLET       PF(LCTR)=100            //LOOPER
          BLET       PF(INDX)=0              //INDEX
          BLET       PF(JNDX)=0              //COUNTER
          BLET       &DUM=''                 //SET TO NULL
```

```
 INV110   BLET       PF(INDX)=PF(INDX)+1    //BUMP
          TEST NE    &CUSTMR(PF$INDX),'',INV120 //NULL CUSTOMER
          TEST NE    &CUSTMR(PF$INDX),&DUM,INV120
          BLET       PF(JNDX)=PF(JNDX)+1
          BLET       &CUSTID(PF$JNDX)=&CUSTMR(PF$INDX)
          BLET       &DUM=&CUSTMR(PF$INDX)
 INV120   LOOP       LCTR$PF,INV110
*-----------
* GET LINEUP
*-----------
          BGETLIST   FILE=ALINEUP,&DUM
          BLET       &I=0
 SIP000   BLET       &I=&I+1                //BUMP
          BGETLIST   FILE=ALINEUP,END=SIP100,&DUM1,MX(SHIPS,&I,2),MX(SHIPS,&I,4)
          BLET       &K=MX(SHIPS,&I,4)-1000  //SAVE TRUCK
          BLET       MX(SHIPS,&I,4)=&K
          BLET       &TRKLD(&K)=&TRKLD(&K)+MX(SHIPS,&I,2)
          BLET       PF(LCTR)=&NOMDLS       //SEARCH FOR MODEL ID
 SIP030   TEST NE    &PARTNO(PF$LCTR),&DUM1,SIP040
          LOOP       LCTR$PF,SIP030         //KEEP LOOKING
          BPUTPIC    &DUM1,&I
ENGINE: * AT LINE **** NOT FOUND CORRECT IN LINEUP OR BEGINV
          TRANSFER   ,SIP000
 SIP040   BLET       MX(SHIPS,&I,1)=PF(LCTR)  //MODEL
          BLET       &DUM1=&CUSTMR(PF$LCTR)
          BLET       PF(LCTR)=50
 SIP050   TEST NE    &DUM1,&CUSTID(PF$LCTR),SIP060
          LOOP       LCTR$PF,SIP050
 SIP060   BLET       MX(SHIPS,&I,5)=PF(LCTR)  //CUSTOMER ID
          TRANSFER   ,SIP000                //GO AGAIN
 SIP100   BCLOSE     ALINEUP
*----------------------------
* GET 568 CYCLE TIMES
*----------------------------
          BGETLIST   FILE=DPT568,&DUM     //SKIP LINE
          BGETLIST   FILE=DPT568,&DUM     //SKIP LINE
          BGETLIST   FILE=DPT568,&DUM     //SKIP LINE
  CYL000  BGETLIST   FILE=DPT568,END=CYL090,&DUM,(&RPASS(&I),&I=1,3)
          BLET       PF(LCTR)=&NOMDLS
  CYL010  TEST NE    &DUM,&PARTNO(PF$LCTR),CYL020  //PART# SEARCH
          LOOP       LCTR$PF,CYL010                //CHECK MATCH
*          BPUTPIC    FILE=OUT,&DUM
* * IN CYL568 NOT FOUND
          TRANSFER   ,CYL000
  CYL020  BLET       &I=PF(LCTR)            //SAVE PART#
          BLET       &BTRIM(&I)=&RPASS(1)  //BLUEBIRD TRIM TIME
          BLET       &CTRIM(&I)=&RPASS(2)  //COMPRESSOR TRIM TIME
          BLET       &COPTN(&I)=&RPASS(3)  //COMPRESSOR OPTION %
          TRANSFER   ,CYL000
  CYL090  BCLOSE     DPT568
*----------------------------
* GET 569 CYCLE TIMES
*----------------------------
          BGETLIST   FILE=DPT569,&DUM     //SKIP LINE
          BGETLIST   FILE=DPT569,&DUM     //SKIP LINE
  CYL100  BGETLIST   FILE=DPT569,END=CYL190,&DUM,(&RPASS(&I),&I=1,8)
          BLET       PF(LCTR)=&NOMDLS
  CYL110  TEST NE    &DUM,&PARTNO(PF$LCTR),CYL120  //PART# SEARCH
          LOOP       LCTR$PF,CYL110                //CHECK MATCH
*          BPUTPIC    FILE=OUT,&DUM
* * IN CYL569 NOT FOUND
          TRANSFER   ,CYL100
  CYL120  BLET       &I=PF(LCTR)            //SAVE PART#
          BLET       &CTEST(&I)=&RPASS(1)  //CELL TEST TIME
          BLET       &HOOK(&I)=&RPASS(2)   //HOOK-UP TIME
          BLET       &UNHK(&I)=&RPASS(3)   //UNHOOK TIME
          BLET       &RHOOK(&I)=&RPASS(4)  //HOOK R-TIME
          BLET       &RTORK(&I)=&RPASS(5)  //RETORQUE TIME
          BLET       &TCRTE(&I)=&RPASS(6)  //CELL ROUTING
```

```
         BLET        &TRJT1(&I)=&RPASS(7)   //1ST TIME REJECT %
         BLET        &TRJT2(&I)=&RPASS(8)   //2ND TIME REJECT %
         TRANSFER    ,CYL100
 CYL190  BCLOSE      DPT569
*----------------------------
* GET 570/571 CYCLE TIMES
*----------------------------
         BGETLIST    FILE=DPT570,&DUM     //SKIP LINE
         BGETLIST    FILE=DPT570,&DUM     //SKIP LINE
         BGETLIST    FILE=DPT570,&DUM     //SKIP LINE
 CYL200  BGETLIST    FILE=DPT570,END=CYL290,&DUM,(&RPASS(&I),&I=1,2),_
                     &J,(&RPASS(&I),&I=3,7)
         BLET        PF(LCTR)=&NOMDLS
 CYL210  TEST NE     &DUM,&PARTNO(PF$LCTR),CYL220   //PART# SEARCH
         LOOP        LCTR$PF,CYL210                 //CHECK MATCH
*         BPUTPIC     FILE=OUT,&DUM
* * IN CYL570 NOT FOUND
         TRANSFER    ,CYL200
 CYL220  BLET        &I=PF(LCTR)            //SAVE PART#
         BLET        &TRIM(&I)=&RPASS(1)    //TRIM TIME
         BLET        &TXFR(&I)=&RPASS(2)    //PAINT TRANSFER TIME
         BLET        &BLOWO(&I)=&RPASS(3)   //BLOW-OFF TIME
         BLET        &MASK(&I)=&RPASS(4)    //MASK TIME
         BLET        &PCOAT(&I)=&RPASS(5)   //PRIME COAT TIME
         BLET        &TCOAT(&I)=&RPASS(6)   //TOP COAT TIME
         BLET        &TRM571(&I)=&RPASS(7)   //FINAL TRIM TIME
         TRANSFER    ,CYL200
 CYL290  BCLOSE      DPT570
*----------------------------
* GET 572 CYCLE TIMES
*----------------------------
         BLET        &I=0
         BGETLIST    FILE=DPT572,&DOCK        //#DOCKS
         BSTORAGE    S$DOCKS,&DOCK
         BGETLIST    FILE=DPT572,&RECTRKS     //#RECEIVING SHIPMENTS
         BGETLIST    FILE=DPT572,&DUM         //SKIP LINE
 CYL300  BGETLIST    FILE=DPT572,END=CYL390,&DUM,&RPASS(1),&DUM1
         TEST E      &DUM1,'min/load',CYL300 //PART# SEARCH
         BLET        &I=&I+1
         BLET        &STECH(&I)=&DUM          //SHIPPING TECH
         BLET        &SHPTIM(&I)=&RPASS(1)    //MASK/BLOW-OFF TIME
         TRANSFER    ,CYL300
 CYL390  BCLOSE      DPT572
*----------------------------
* GET PAINT PARAMETERS
*----------------------------
         BGETLIST    FILE=DPT571,&LBCTPNT     //PAINT LOAD BARS
         BGETLIST    FILE=DPT571,&TIMEOVEN    //OVEN TIMER/LOAD BAR
         BGETLIST    FILE=DPT571,&FLASH       //PAINT FLASH TIME/STOP
         BGETLIST    FILE=DPT571,&COOL        //PAINT COOLDOWN/STOP
         BGETLIST    FILE=DPT571,&PNTFSP      //PAINT FAST SPEED
         BGETLIST    FILE=DPT571,&PNTSSP      //PAINT PROCESS SPEED
SPSPD    MACRO       PNT2,&PNTSSP
         BGETLIST    FILE=DPT571,&PCMAX       //PAINT PROCESS CHAIN MAX
         BSTORAGE    S$SPNT2,&PCMAX           //WASHER LIMIT
         BCLOSE      DPT571
*----------------------------
* READ IN DOWN TIME SCENARIOS
*----------------------------
         BGETLIST    FILE=DWNTIM,&DUM
         BLET        PF(INDX)=0
DTS000   BLET        PF(INDX)=PF(INDX)+1       //BUMP INDEX VALUE
         BGETLIST    FILE=DWNTIM,END=DTS100,&DUM,PF(JNDX),&DELAY1(PF$INDX),_
                     &MTBF(PF$INDX),&DTIM(PF$INDX),PF(LCTR)
         SPLIT       1,DWT000
         TRANSFER    ,DTS000
DTS100   BCLOSE      DWNTIM
****************************************************************************
* Simulation Timer Module
```

```
* Written by G. Rehn
* 6/29/98
* Version 01
*******************************************************************
* Input Operation Data
*
 CTM000   BLET        &M=&CLKS/15+1             //STARTING SEGMENT
          TEST E      &M,97,*+2                 //BEYOND DAY'S END
          BLET        &M=1
          BGETLIST    FILE=OPDAT,&DUM,(&OPAS(&J),&J=1,96)
          BLET        &I=0
 CTM010   BLET        &I=&I+1
          TEST E      &OPAS(&I),'B',*+3
          BLET        &DFTOP(&I)=15
          TRANSFER    ,CTM020
          TEST E      &OPAS(&I),'F',*+3
          BLET        &DFTOP(&I)=5
          TRANSFER    ,CTM020
          TEST E      &OPAS(&I),'C',*+3
          BLET        &DFTOP(&I)=10
          TRANSFER    ,CTM020
          TEST E      &OPAS(&I),'A',*+3
          BLET        &DFTOP(&I)=-2
          TRANSFER    ,CTM020
          TEST E      &OPAS(&I),'E',*+3
          BLET        &DFTOP(&I)=-1
          TRANSFER    ,CTM020
          BLET        &DFTOP(&I)=CHARSTOI(&OPAS(&I))
 CTM020   TEST E      &I,96,CTM010
*
          BLET        PF(CTR)=&M
          TEST E      &DFTOP(&M),-2,CTM040     //@SHIFT START?
 CTM030   BLET        PF(CTR)=PF(CTR)+1         //SEARCH FORWARD
          BLET        PF(CTR)=FN(PCNVRT)         //POINTER CONVERT
          TEST NE     BV(DISFT),1,CTM060        //FOUND INITIAL?
          TEST NE     &DFTOP(PF(CTR)),-1,CTM050  //@END?;LOOK BACKWARDS
          TRANSFER    ,CTM030                   //SEARCH FORWARD EVERYTHING ELSE
*
 CTM040   BLET        PF(CTR)=FN(PCNVRT)            //POINTER CONVERT
          TEST NE     BV(DISFT),1,CTM060            //FOUND INITIAL SHIFT?
          TEST NE     &DFTOP(PF(CTR)),-2,CTM030  //@START?;LOOK FORWARD
 CTM050   BLET        PF(CTR)=PF(CTR)-1                 //REDUCE
          TRANSFER    ,CTM040
 CTM060   BLET        &N=&DFTOP(PF(CTR))        //STARTING SHIFT
*
 CTM070   BLET        &I=0                      //RESET
 CTM080   BLET        &I=&I+1
          BGETLIST    FILE=OPDAT,END=CTM190,&MODID(&I),(&OPAS(&J),&J=1,96),&EFMIN(&I),_
                      &OPHRS(&I),&OPSFT(&I),&WDAYS(&I),&PERF(&I)
          TEST G      &EFMIN(&I),0,CTM180       //MODULE IN PLAY?
          TEST G      &WDAYS(&I),&WEEK,*+2      //WORK DAYS>WEEK?
          BLET        &WEEK=&WDAYS(&I)          //YES;NEW WEEK DEFINITION
          TEST E      &OPAS(1),'D',CTM100       //DEFAULT?
          BLET        &J=0
 CTM090   BLET        &J=&J+1                   //BUMP POINTER
          BLET        MH(HPS,&I,&J)=&DFTOP(&J)
          TEST E      &J,96,CTM090
          BLET        &MSHIFT(&I)=&N            //TAG INITIAL SHIFT
          TRANSFER    ,CTM180
*
 CTM100   BLET        &J=0
 CTM110   BLET        &J=&J+1
          TEST E      &OPAS(&J),'B',*+3
          BLET        MH(HPS,&I,&J)=15
          TRANSFER    ,CTM120
          TEST E      &OPAS(&J),'F',*+3
          BLET        MH(HPS,&I,&J)=5
          TRANSFER    ,CTM120
          TEST E      &OPAS(&J),'C',*+3
```

```
           BLET       MH(HPS,&I,&J)=10
           TRANSFER   ,CTM120
           TEST E     &OPAS(&J),'A',*+3
           BLET       MH(HPS,&I,&J)=-2
           TRANSFER   ,CTM120
           TEST E     &OPAS(&J),'E',*+3
           BLET       MH(HPS,&I,&J)=-1
           TRANSFER   ,CTM120
           TEST E     &OPAS(&J),'S',*+3        //INDICATES SWING OPERATION
           BLET       MH(HPS,&I,&J)=99
           TRANSFER   ,CTM120
           BLET       MH(HPS,&I,&J)=CHARSTOI(&OPAS(&J))
 CTM120    TEST E     &J,96,CTM110
*
           BLET       PF(CTR)=&M
           TEST E     MH(HPS,&I,PF(CTR)),-2,CTM140 //@SHIFT START?
 CTM130    BLET       PF(CTR)=PF(CTR)+1       //SEARCH FORWARD
           BLET       PF(CTR)=FN(PCNVRT)      //POINTER CONVERT
           TEST NE    BV(TISFT),1,CTM160      //FOUND INITIAL?
           TEST NE    MH(HPS,&I,PF(CTR)),-1,CTM150  //@END?'LOOK BACKWARDS
           TRANSFER   ,CTM130                 //SEARCH FORWARD EVERYTHING ELSE
*
 CTM140    BLET       PF(CTR)=FN(PCNVRT)              //POINTER CONVERT
           TEST NE    BV(TISFT),1,CTM160              //FOUND INITIAL SHIFT?
           TEST NE    MH(HPS,&I,PF(CTR)),-2,CTM130    //@START?;LOOK FORWARD
 CTM150    BLET       PF(CTR)=PF(CTR)-1               //REDUCE
           TRANSFER   ,CTM140
 CTM160    BLET       &MSHIFT(&I)=MH(HPS,&I,PF(CTR)) //TAG INITIAL SHIFT
*
 CTM180    TEST E     &I,10,CTM080           //FINISH READ
 CTM190    BCLOSE     OPDAT
***********************************************************************
* READ IN TECHNICIAN DATA
*
           BLET       PF(PLOC)=0             //ZERO OUT FOR SWING ID
           BGETLIST   FILE=TECHS,&DUM
 TIN000    BGETLIST   FILE=TECHS,END=DIN000,ERR=DIN000,PF(TECHN),&TCHNM(PF$TECHN),_
                      &DUM,&DUM1,PF(SHFT),(&CPASS(&J),&J=1,6)
           TEST NE    &TCHNM(PF$TECHN),'0',DIN000
           TEST E     &DUM1,'Y',TIN000                //TECH IN PLAY?
           BLET       PF(MOD)=10                      //MOD SEARCH
 TIN010    TEST NE    &DUM,&MODID(PF$MOD),TIN020      //MODULE MATCH
           LOOP       MOD$PF,TIN010                   //KEEP LOOKING
           TRANSFER   ,DIN000
 TIN020    BLET       PF(LCTR)=6
 TIN025    TEST NE    &CPASS(PF$LCTR),'0',TIN060
           BLET       PF(INDX)=0
 TIN030    BLET       PF(INDX)=PF(INDX)+1
           TEST NE    &CPASS(PF$LCTR),&SNAME(PF$INDX),TIN040
           TRANSFER   ,TIN030
 TIN040    BLET       PF(PF$LCTR+20)=PF(INDX)
 TIN060    LOOP       LCTR$PF,TIN025
 TIN070    BLET       &K=V(ANYOP)                     //SUM OF ALL OPERATIONS
           TEST G     &K,0,TIN000                     //IF ZERO; NO TECH
           BLET       &TLAST=PF$TECHN
           BLET       &TCC=&TCC+1                     //#TECHNICIAN
           BLET       MX(TCHASN,PF$TECHN,1)=PF(MOD)   //SAVE ASSIGNMENTS
           BLET       MX(TCHASN,PF$TECHN,2)=PF(SHFT)  //SAVE ASSIGNMENTS
           BLET       MX(TCHASN,PF$TECHN,3)=PF(LOC1)  //SAVE ASSIGNMENTS
           BLET       MX(TCHASN,PF$TECHN,4)=PF(LOC2)  //SAVE ASSIGNMENTS
           BLET       MX(TCHASN,PF$TECHN,5)=PF(LOC3)  //SAVE ASSIGNMENTS
           BLET       MX(TCHASN,PF$TECHN,6)=PF(LOC4)  //SAVE ASSIGNMENTS
           BLET       MX(TCHASN,PF$TECHN,7)=PF(LOC5)  //SAVE ASSIGNMENTS
           BLET       MX(TCHASN,PF$TECHN,8)=PF(LOC6)  //SAVE ASSIGNMENTS
 TIN080    TEST E     MH(TCH1,PF$MOD,PF$SHFT),0,*+2   //ANY VALUE HERE?
           BLET       MH(TCH1,PF$MOD,PF$SHFT)=PF(TECHN)+TCHNS //NO;MUST BE FIRST
           BLET       MH(TCHL,PF$MOD,PF$SHFT)=PF(TECHN)+TCHNS //CURRENT=LAST
           TEST E     &MODID(PF$MOD),'569S',TIN090    //SWING SHIFT?
           BLET       PF(PLOC)=PF(PLOC)+1             //BUMP COUNTER
```

```
TIN090  SPLIT      1,TCH000                      //CREATE XACT
        TEST E     PF(PLOC),2,*+2                //SECOND?
        BLET       PF(PLOC)=0                    //YES;RESET
        PRIORITY   -1,YIELD                      //XACT GET THERE
        PRIORITY   0
        BLET       PF(LOC1)=0                    //ZERO OUT FOR NEXT READ
        BLET       PF(LOC2)=0                    //ZERO OUT FOR NEXT READ
        BLET       PF(LOC3)=0                    //ZERO OUT FOR NEXT READ
        BLET       PF(LOC4)=0                    //ZERO OUT FOR NEXT READ
        BLET       PF(LOC5)=0                    //ZERO OUT FOR NEXT READ
        BLET       PF(LOC6)=0                    //ZERO OUT FOR NEXT READ
        TRANSFER   ,TIN000                       //LOOP AGAIN
*---------------------------------------------------------------------------
* DONE INPUTING - INITILIZE SYSTEM/CREATE REMAINING ACTIVE ENTITIES
*---------------------------------------------------------------------------
DIN000  BCLOSE     TECHS
WRITE   MACRO      TESTID,&TESTID
WRITE   MACRO      TESTDSCR,&TESTDSCR
        BLET       PF(LCTR)=&LBCTMAIN      //TOTAL # LOAD BARS IN SYSTEM
        TRANSFER   ,DIN130
*
* 572 INVENTORY
*
        BLET       PF(ENGINE)=99           //START 572 INITILIZATION
DIN010  TEST G     &SIN572(PF$ENGINE),0,DIN020 //ANY OF THIS ENGINE?
        BLET       &INV572(PF$ENGINE)=&SIN572(PF$ENGINE) //YES;INIT
        BLET       &INV572(100)=&INV572(100)+&SIN572(PF$ENGINE)
        SPLIT      &SIN572(PF$ENGINE),FIN060          //ENGINE TO FINISHED
        ENTER      TOTALQ,&SIN572(PF$ENGINE)
        PRIORITY   -1,YIELD
        PRIORITY   0
DIN020  LOOP       ENGINE$PF,DIN010        //KEEP LOOPING
WRITE   MACRO      IV572,&INV572(100)      //INITILIZE #
BARG    MACRO      IVB,RIGHT,&INV572(100)
        TRANSFER   ,DIN130
*
*
* 570 INVENTORY
*
        BLET       &DUM='570'
        TRANSFER   SBR,FNDMOD,SUBR$PF
        BLET       PF(ENGINE)=0              //ENGINES
        BLET       PF(PTR)=50                //570 ENGINES ON FLOOR
DIN030  BLET       PF(ENGINE)=PF(ENGINE)+1
        BLET       PF(CTR)=&SIN570(PF$ENGINE) //#AVAILABLE
        TEST G     PF(CTR),0,DIN060          //> 0?
        TEST G     PF(PTR),0,DIN045          //FINISHED W/ FLOOR?
        BLET       PF(PTR)=PF(PTR)-PF(CTR)   //REDUCE FLOOR COUNT
DIN040  SPLIT      1,ISP000                  //TO INSPECT
        BLET       &INPROC=&INPROC+1         //COUNT IN PROCESS
        ENTER      EWIPQ
        ENTER      TOTALQ
        ADVANCE    .1
        LOOP       CTR$PF,DIN040
        TRANSFER   ,DIN060
ISP000  TERMINATE
*
DIN045  BLET       PF(LCTR)=PF(LCTR)-PF(CTR) //CONSUME LOAD BARS
DIN050  BLET       &INPROC=&INPROC+1         //COUNT IN PROCESS
        ENTER      EWIPQ
        ENTER      TOTALQ
        GATE SNF   SOUT
        SPLIT      1,ITT000                  //CREATE ENGINE
        ADVANCE    .1                        //CLEARANCE
        LOOP       CTR$PF,DIN050
DIN060  TEST GE    PF$ENGINE,99,DIN030      //MORE ENGINES
*
* 569 INVENTORY
*
```

```
        BLET      &DUM='569'
        TRANSFER  SBR,FNDMOD,SUBR$PF
        BLET      PF(ENGINE)=0                //ENGINES
DIN070  BLET      PF(ENGINE)=PF(ENGINE)+1
        BLET      PF(CTR)=&SIN569(PF$ENGINE)  //#AVAILABLE
        TEST G    PF(CTR),0,DIN090            //> 0?
        BLET      PF(LCTR)=PF(LCTR)-PF(CTR)
DIN080  BLET      &INPROC=&INPROC+1           //COUNT IN PROCESS
        ENTER     EWIPQ
        ENTER     TOTALQ
        GATE SNF  SP1
        SPLIT     1,ITC000                    //CREATE ENGINE
        ADVANCE   .1                          //CLEARANCE
        LOOP      CTR$PF,DIN080
DIN090  TEST GE   PF$ENGINE,99,DIN070            //MORE ENGINES
*
* 568 INVENTORY
*
        BLET      &DUM='568'
        TRANSFER  SBR,FNDMOD,SUBR$PF
        BLET      PF(ENGINE)=0                //ENGINES
DIN100  BLET      PF(ENGINE)=PF(ENGINE)+1
        BLET      PF(CTR)=&SIN568(PF$ENGINE)  //#AVAILABLE
        TEST G    PF(CTR),0,DIN120            //> 0?
        BLET      PF(LCTR)=PF(LCTR)-PF(CTR)
        BLET      PF(RJCT)=0                  //ZERO OUT REJECT INDICATOR
        TEST E    &CTRIM(PF$ENGINE),0,DIN110 //COMPRESSOR ENGINE?
        TEST E    &BTRIM(PF$ENGINE),0,DIN110 //NO;BLUEBIRD?
        BLET      PF(RJCT)=1                  //MUST BE REJECT
DIN110  BLET      &INPROC=&INPROC+1           //COUNT IN PROCESS
        ENTER     EWIPQ
        ENTER     TOTALQ
        GATE LC   260
        SPLIT     1,ITR000                    //CREATE ENGINE
        ADVANCE   .1                          //CLEARANCE
        LOOP      CTR$PF,DIN110
DIN120  TEST GE   PF$ENGINE,99,DIN100            //MORE ENGINES
*
DIN130  SPLIT     1,CNV000                    //START MAIN DELIVERY CONV.
        PRIORITY  -1,YIELD
        PRIORITY  0
        SPLIT     1,LIN000                    //START ASSEMBLY LINE
        PRIORITY  -1,YIELD
        PRIORITY  0
        GATE LS   DINIT
        BLET      PF(MOD)=-1
        SPLIT     10,TMR000,MOD$PF            //CREATE MODULE CONTROLS
        PRIORITY  -1,YIELD
        PRIORITY  0
*------------------------------------------------------------
* MAIN XACT EXECUTES CLOCK
* CLOCK MOVEMENT
* DAY STARTS AT       &CLKS
*------------------------------------------------------------
CLK000  BLET      &AMPM(1)='AM'            //INITIALIZE AM/PM VAR
        BLET      &AMPM(2)='PM'            //INITIALIZE AM/PM VAR
        BLET      PF1=&CLKS/60             //#HOURS INITIAL OFFSET
        BLET      PF2=&CLKS@60             //#MINS. INITIAL OFFSET
        BLET      PF3=&CLKS/15             //#CLOCK LOOPS
        BLET      PL1=PF1@12               //MODULUS OF 12
        BLET      PL2=PF2                  //MAKE REAL #
        TEST L    PF3,48,CLK005            //START IN MORN OR AFTERNOON?
        LOGIC C   MORN                     //YES; MORNING
        BLET      PL3=(48-PF3)*15          //TIME AM/PM SWITCH
        TRANSFER  ,CLK010
CLK005  LOGIC S   MORN                     //NO;AFTERNOON
        BLET      PL3=(96-PF3)*15
CLK010  BLET      PF4=LS(MORN)+1           //AM/PM POINTER
WRITE   MACRO     DST,&AMPM(PF4)           //AM/PM INDICATOR
```

```
         BLET       PL1=(PL1/12+PL2/720)*130.85 //HOUR HAND OFFSET
         BLET       PL2=(PL2/60)*130.85         //MIN. HAND OFFSET
 PLONAT  MACRO      'MHND',TYMM,PL2          //INITIAL SET MINUTE HAND
 PLONAT  MACRO      'HHND',TYMH,PL1          //INITIAL SET MINUTE HAND
         ADVANCE    PL3                      //INITIAL TIME TO AM/PM SWITCH
 CLK020  LOGIC I    MORN                     //INVERT AM/PM
         BLET       PF4=LS(MORN)+1           //AM/PM POINTER
 WRITE   MACRO      DST,&AMPM(PF4)           //AM/PM INDICATOR
         ADVANCE    720                      //NEXT 12 HRS
         TRANSFER   ,CLK020
**********************************************************************
* MODULE OPERATION CONTROL
**********************************************************************
 TMR000  TEST G     &EFMIN(PF$MOD),0,TMRSTP  //MOD IN OPERATION?
         BLET       PF(SHFT)=&MSHIFT(PF$MOD) //YES;GET INITIAL SHIFT
         TEST NE    PF(MOD),1,JOP000
         TEST NE    PF(SHFT),0,TMRADV        //ACTIVE SHIFT?
         BLET       PF(OPER1)=MH(TCH1,PF$MOD,PF$SHFT) //FIRST FACILITY
         BLET       PF(OPERL)=MH(TCHL,PF$MOD,PF$SHFT) //LAST FACILITY
         FUNAVAIL   PF(OPER1)-PF(OPERL)      //SHUT EVERYONE OFF
         BLET       PF(CTR)=&CLKS/15+1       //STARTING SEGMENT
         BLET       PF(CTR)=FN(PCNVRT)
         BLET       PF(PTR)=MH(HPS,PF(MOD),PF(CTR)) //CURRENT SEGMENT VALUE
         TEST G     PF(PTR),0,TMRINT         //CHECK FOR START/STOP
         TEST LE    PF(PTR),3,TMRINT         //IN OPERATION?
         FAVAIL     PF(OPER1)-PF(OPERL)      //PUT IN PLAY
         TRANSFER   SBR,FACLR,SUBR$PF        //CHANGE OPER COLORS
 TMRADV  ADVANCE    15                       //TIME ADVANCE
 TMR010  BLET       PF(CTR)=PF(CTR)+1        //BUMP SEGMENT
         BLET       PF(CTR)=FN(PCNVRT)       //YES;RESET
         TRANSFER   ,FN(TMDIR)               //PROCEED
*
 TMRINT  TEST NE    PF(PTR),-2,TMRBEG        //@ START SHIFT
         FAVAIL     PF(OPER1)-PF(OPERL)      //PUT IN PLAY
         TRANSFER   SBR,FACLR,SUBR$PF        //CHANGE TECH COLORS
         TRANSFER   ,FN(TMDIR)               //PROCEED
*
* START OF SHIFT
*
 TMRBEG  TEST E     &SDAY@&WEEK,0,*+2  //END OF WEEK?
         TEST E     &WDAYS(PF$MOD),&WEEK,TMRWKE      //YES;WORK THE WEEKEND?
         BLET       &ACNOOP(PF$MOD)=&ACNOOP(PF$MOD)+&SALOW      //START-UP
         BLET       &ACNOOP(PF$MOD)=&ACNOOP(PF$MOD)+MP$WAIT$PL //OFFTIME
         BLET       PF(PTR)=MH(HPS,PF(MOD),PF(CTR)) //CURRENT SEGMENT VALUE
         ADVANCE    &SALOW                    //DO STARTUP ALLOWANCE
         BLET       PF(SHFT)=PF(CTR)          //CURRENT POSITION
 TMR015  BLET       PF(SHFT)=PF(SHFT)+1       //BUMP POINTER
         TEST E     PF(SHFT),97,*+2           //END OF ROAD?
         BLET       PF(SHFT)=1                //YES;REST TO 1
         TEST G     MH(HPS,PF$MOD,PF$SHFT),0,TMR015 //IN OPERATION?
         TEST LE    MH(HPS,PF$MOD,PF$SHFT),3,TMR015 //NOT A BREAK?
         BLET       PF(SHFT)=MH(HPS,PF$MOD,PF$SHFT) //FOUND NEXT SHIFT
         TEST NE    PF(MOD),1,JOP010          //564 MODULE
         BLET       PF(OPER1)=MH(TCH1,PF$MOD,PF$SHFT) //FIRST FACILITY
         BLET       PF(OPERL)=MH(TCHL,PF$MOD,PF$SHFT) //LAST FACILITY
         FAVAIL     PF(OPER1)-PF(OPERL)       //START-UP
         TRANSFER   SBR,FACLR,SUBR$PF         //CHANGE TECH COLOR
         BLET       &MD=PF(MOD)               //SAVE MODULE
         BLET       &SF=PF(SHFT)              //SAVE SHIFT
         UNLINK     IPOOL,TMR030,ALL,BV$SFTCO //SHIFT CHANGE-OVER
 TMR020  ADVANCE    15-&SALOW                 //PROCEED
         TRANSFER   ,TMR010
*
 TMR030  TEST E     &MODID(PF$MOD),'569',TMR040
         SPLIT      1,ATS000                  //DETERMINE ACTIVE CELLS
         TRANSFER   ,TCH160
*
 TMR040  TEST E     &MODID(PF$MOD),'569S',TCH160
         TEST NE    PF(PLOC),2,TCH400
```

```
          SPLIT      1,ATS000                   //DETERMINE ACTIVE CELLS
          TRANSFER   ,TCH160
*
 TMRBRK   FUNAVAIL   PF(OPER1)-PF(OPERL)          //OPERATOR BREAK
          TRANSFER   SBR,FUNCLR,SUBRSPF         //CHANGE TECH COLOR
 TMR050   TEST E     MH(HPS,PF(MOD),PF(CTR)),15,SBREK //MIN TMRBRK?
          BLET       &ACNOOP(PF$MOD)=&ACNOOP(PF$MOD)+15
          ADVANCE    15                         //DO 15 MIN. TMRBRK
          BLET       PF(CTR)=PF(CTR)+1          //BUMP SEGMENT
          BLET       PF(CTR)=FN(PCNVRT)         //YES;RESET
          TEST NE    MH(HPS,PF(MOD),PF(CTR)),99,TMR055 //STILL ON BREAK?
          TEST LE    MH(HPS,PF(MOD),PF(CTR)),3,TMR050 //STILL ON BREAK?
 TMR055   FAVAIL     PF(OPER1)-PF(OPERL)          //NO;BACK IN OPERATION
          TRANSFER   SBR,FACLR,SUBRSPF
          TRANSFER   ,FN(TMDIR)
*
 SBREK    ADVANCE    MH(HPS,PF(MOD),PF(CTR))  //SHORT TMRBRK
          BLET       &ACNOOP(PF$MOD)=&ACNOOP(PF$MOD)+MH(HPS,PF(MOD),PF(CTR))
          FAVAIL     PF(OPER1)-PF(OPERL)         //BACK IN OPERATION
          TRANSFER   SBR,FACLR,SUBRSPF
          ADVANCE    15-MH(HPS,PF(MOD),PF(CTR)) //RESUME
          TRANSFER   ,TMR010
*
 TMREND   TEST E     &MODID(PF$MOD),'569',*+2
          UNLINK     ACELLS,ATS100,ALL        //ACTIVE CELLS
          ADVANCE    15-&CALOW                //CLEAN-UP ALLOWANCE
          TEST NE    PF(MOD),1,JOP130         //564 CONTROL?
          UNLINK     PDLAY,TMR100,ALL,MODSPF,PFSMOD  //REORDER WAITING XACTS
          UNLINK     NOTCH,TMR100,ALL,MODSPF,PFSMOD  //REORDER WAITING XACTS
          LOGIC C    MATCH
          TEST NE    &MODID(PF$MOD),'569S',TMR091
          BLET       PF(LCTR)=PF(OPERL)-PF(OPER1)+1  //# OPERATORS
          BLET       PF$JNDX=PF(OPER1)               //STARTING OPR. INDEX
 TMR080   BLET       PF$INDX=&OPXID(PF$JNDX-TCHNS)   //GET OBJECT ID
          TEST G     PF(INDX),0,TMR085
SCOLOR    MACRO      PF$INDX,'BAC'
          PREEMPT    PF(JNDX),,TMR200,CYCLE$PL     //INTERRUPT TECH IN ACTION
          PRIORITY   -1,YIELD
          PRIORITY   0
          RETURN     PF(JNDX)
 TMR085   BLET       PF$JNDX=PF$JNDX+1        //BUMP POINTER
          LOOP       LCTRSPF,TMR080           //CONTINUE
          UNLINK     APOOL,TMR300,ALL,MODSPF,PFSMOD
          UNLINK     HOLD,TMR150,ALL          //RELEASE HELD XACTS
 TMR090   FUNAVAIL   PF(OPER1)-PF(OPERL)      //OPERATOR TMRBRK
          ADVANCE    &CALOW                   //DO CLEAN-UP
          BLET       &ACNOOP(PFSMOD)=&ACNOOP(PFSMOD)+&CALOW
          MARK       WAITSPL                  //COLLECT STOPPAGE TIME
          BLET       PF(PTR)=0
          TRANSFER   ,TMR010
*
 TMR091   BLET       PF(JNDX)=PF(OPERL)
          BLET       PF(LCTR)=1
          TRANSFER   ,TMR080
*
* SHIFT CHANGEOVER LOGIC
*
 TMR100   LINK       HOLD,FIFO                //STAGE TEMPORARILY
 TMR150   LINK       NOTCH,FIFO               //REORDER
*
 TMR200   ALTERUCH E INPRO,1,CYCLESPL,PLSCYCLE,CLOCSPF,PFSCLOC
          ALTERUCH E INPRO,1,OPNUMSPF,PFSOPNUM,CLOCSPF,PFSCLOC
          REMOVE     ATECHS
          RELEASE    PF(TECHN)+TCHNS        //NO;RELEASE
          BLET       PL(CMPEST)=0
          UNLINK     INPRO,TMR150,1,CLOCSPF,PFSCLOC
 PLON     MACRO      XID1,TECHSTG
 TMR300   LINK       IPOOL,FIFO
*
```

```
 TMRSWG    UNLINK E    APOOL,TCH500,1,TECHN$PF,(PF$OPER1-TCHNS),TMR400
           TRANSFER    ,TMRADV
 TMR400    ALTER E     ATECHS,1,OPER1$PF,-99,TECHN$PF,(PF$OPER1-TCHNS)
           TRANSFER    ,TMRADV
*
* WEEKEND STOPPAGE
*
 TMRWKE    BLET        PF1=&SDAY
           ADVANCE     1440
           TEST NE     PF1,&SDAY
           TRANSFER    ,TMRBEG
*
* DETERMINE ACTIVE CELLS
*
 ATS000    BLET        PF(LCTR)=2                //CHECK 1ST 2 ASNS ONLY
 ATS010    BLET        PF(DELRT)=PF(20+PF$LCTR)  //POINT TO POSSIBLE STA
           TEST G      PF(DELRT),40,ATS020       //TEST CELL STATION
           TEST LE     PF(DELRT),63,ATS020       //MAX. TEST CELL
           SPLIT       1,ATS050
 ATS020    LOOP        LCTR$PF,ATS010
 ATS030    TERMINATE
*
 ATS050    GATE LC     PF(DELRT),ATS030          //ALREADY ACTIVE?
           LOGIC S     PF(DELRT)                 //NO;NOW IS
           GATE LS     81,*+2
           LOGIC C     81
           GATE LS     87,*+2
           LOGIC C     87
           BLET        PF(DELRT)=PF(DELRT)-40    //ADJUST POINTER
           LINK        ACELLS,FIFO               //ON ACTIVE CHAIN
 ATS100    LOGIC C     PF(DELRT)+40              //RESET TO INACTIVE
           TERMINATE
*
* TECHNICIAN COLOR SUBROUTINES
*
 FACLR     TEST NE     PF(MOD),1,JOP120                //564?
           BLET        PF(LCTR)=PF(OPERL)-PF(OPER1)+1 //# OPERATORS
           BLET        PF$JNDX=PF(OPER1)                //STARTING OPR. INDEX
 CLR010    BLET        PF$INDX=&OPXID(PF$JNDX-TCHNS)   //GET OBJECT ID
           TEST G      PF(INDX),0,CLR015
SCOLOR     MACRO       PF$INDX,&COPR(PF$JNDX-TCHNS)
 CLR015    BLET        PF$JNDX=PF$JNDX+1               //BUMP POINTER
           LOOP        LCTR$PF,CLR010                  //CONTINUE
           TRANSFER    ,PF(SUBR)+1                     //RETURN
*
 FUNCLR    TEST NE     PF(MOD),1,JOP110                //564?
           BLET        PF(LCTR)=PF(OPERL)-PF(OPER1)+1 //# OPERATORS
           BLET        PF$JNDX=PF(OPER1)                //STARTING OPR. INDEX
 CLR020    BLET        PF$INDX=&OPXID(PF$JNDX-TCHNS)   //GET OBJECT ID
           TEST G      PF(INDX),0,CLR025
SCOLOR     MACRO       PF$INDX,'LAYOUT'
 CLR025    BLET        PF$JNDX=PF$JNDX+1               //BUMP POINTER
           LOOP        LCTR$PF,CLR020                  //CONTINUE
           TRANSFER    ,PF(SUBR)+1                     //RETURN
*
 TMRSTP    TERMINATE                                   //INACTIVE MODULE
*
* KEY OBJECT CREATION
*
 KEY000    BLET        &KEYCNT=&KEYCNT+1
 CREATE    MACRO       KEY,XID1
 WRITEO    MACRO       KEYID,XID1,&CUSTMR(PF$LOC1)
 SCOLOR    MACRO       XID1,&ECLR(PF$LOC1)
           BLET        PF(LOC2)=250-(10*&KEYCNT)
 PLACEAT   MACRO       XID1,0,PF$LOC2
           TERMINATE
*
* 564 SPECIAL CONTROL
*
```

```
JOP000   BLET        PF(OPER1)=200                  //FIXED 1ST
         BLET        PF(OPERL)=260                  //FIXED LAST
         BLET        PF(CTR)=&CLKS/15+1             //STARTING SEGMENT
         BLET        PF(CTR)=FN(PCNVRT)
         BLET        PF(PTR)=MH(HPS,PF(MOD),PF(CTR)) //CURRENT SEGMENT VALUE
         FUNAVAIL    PF(OPER1)-PF(OPERL)
SPSPD    MACRO       JHK6,0
SPSPD    MACRO       JHK7,0
         BLET        &SSP=&JHSPD(1)                 //INITIAL SPEED
         TEST NE     PF(PTR),0,TMRADV               //OFFSHIFT START
         TEST LE     PF(PTR),3,FN(TMDIR)            //ON BREAK START
         TEST NE     PF(PTR),-2,TMRBEG              //@BEGINNING SHIFT START
         FAVAIL      PF(OPER1)-PF(OPERL)            //ALL ELSE IN PLAY
         BLET        &SSP=&JHSPD(PF$SHFT)           //SET SPEED
SPSPD    MACRO       JHK6,&SSP
SPSPD    MACRO       JHK7,&SSP
         TRANSFER    ,FN(TMDIR)
*                                                   //LOOP PARAMTER
JOP010   BLET        PF(LCTR)=260
         FAVAIL      PF(OPER1)-PF(OPERL)        //PUT IN PLAY
JOP020   TEST GE     PF(LCTR),212,JOP030
         PREEMPT     PF(LCTR),,JOP100,CYCLE$PL      //PREEMPT & SAVE CYCLE
         LOOP        LCTR$PF,JOP020
JOP030   PRIORITY    -1,YIELD
         PRIORITY    0
         BLET        PF(LCTR)=260                   //RETURN CONTROL
JOP040   TEST GE     PF(LCTR),212,JOP050
         RETURN      PF(LCTR)
         LOOP        LCTR$PF,JOP040
JOP050   ALTERUCH NE 212,ALL,SHFT$PF,PF$SHFT,SHFT$PF,PF$SHFT
         BLET        &CYADJ=&SSP/&JHSPD(PF$SHFT) //ADJUST TO NEW LS
         UNLINK      212,JOP060,ALL
         BLET        &SSP=&JHSPD(PF$SHFT)
SPSPD    MACRO       JHK6,&SSP
SPSPD    MACRO       JHK7,&SSP
         TRANSFER    ,TMR020                        //RETURN
*
JOP060   BLET        PL(CYCLE)=PL(CYCLE)*&CYADJ  //ADJUST TIME
         TEST E      PL(ITIME),1,JHK035            //DIRECT ACCORDING TO STATUS
         TRANSFER    ,JHK031
*
JOP100   LINK        212,FIFO
*
JOP110   ADVANCE     0
SPSPD    MACRO       JHK6,0
SPSPD    MACRO       JHK7,0
         TRANSFER    ,PF(SUBR)+1
*
JOP120   ADVANCE     0
SPSPD    MACRO       JHK6,&SSP
SPSPD    MACRO       JHK7,&SSP
         TRANSFER    ,PF(SUBR)+1
*
JOP130   ADVANCE     0
SPSPD    MACRO       JHK6,0
SPSPD    MACRO       JHK7,0
         TRANSFER    ,TMR090
*
* INITILIZATION STATUS
*
ITT000   ENTER       SOUT
CREATE   MACRO       LBR,XID1
SCOLOR   MACRO       XID1,&ECLR(PF$ENGINE)
PLON     MACRO       XID1,OUT
*         TRANSFER    ,ITT100
*
ITC000   ENTER       SP1
CREATE   MACRO       LBR,XID1
SCOLOR   MACRO       XID1,&ECLR(PF$ENGINE)
```

```
PLON      MACRO      XID1,P1
*          TRANSFER   ,CNV100
*
ITR000    LOGIC S    260
          ENTER      260
          ENTER      SPO
CREATE    MACRO      LBR,XID1
          TEST NE    PF(ENGINE),0,ITR010
SCOLOR    MACRO      XID1,&ECLR(PF$ENGINE)
*          TRANSFER   ,RPH001
ITR010    ADVANCE    0
SCOLOR    MACRO      XID1,'WHITE'
*          TRANSFER   ,RPH001
*
ITE000    ENTER      PF(CVSEC)
CREATE    MACRO      LBR,XID1
PLON3     MACRO      XID1,P,PF(CVSEC)
*          TRANSFER   ,EMP050
*------------------------------------------------------------
* CREATION OF ACTIVE TECHNICIAN XACTS
*--------------------------------------------
TCH000    BLET       &OPXID(PF$TECHN)=XID1 //SAVE XACT#
CREATE    MACRO      TECH,XID1
WRITEO    MACRO      TID,XID1,PF(TECHN)
PLON      MACRO      XID1,TECHSTG
          TEST E     PF(SHFT),&MSHIFT(PF$MOD),TCH010
          BLET       &COPR(PF$TECHN)='LAY'
SCOLOR    MACRO      XID1,'LAY'
          TEST NE    PF(PLOC),2,TCH400     //2ND ASSIGNMENT
          SPLIT      1,ATS000              //ACTIVE CELL LOGIC
          LINK       APOOL,FIFO            //PLACE IN ACTIVE POOL
TCH010    BLET       &COPR(PF$TECHN)='BAC'
SCOLOR    MACRO      XID1,'BAC'
PLON      MACRO      XID1,TECHSTG
          LINK       IPOOL,FIFO           //PLACE IN INACTIVE POOL
TCH100    SEIZE      PF(TECHN)+TCHNS      //GRAB OPERATOR
          BLET       PF(OPNUM)=-1         //ASSIGNED
          JOIN       ATECHS               //IN ACTIVE GROUP
          TEST E     PL(CMPEST),0,*+2
          BLET       PL(CMPEST)=PL(CYCLE)+AC1
          MARK       WAITSPL
          BLET       PL(ACMBRK)=&ACNOOP(PF$MOD)
          BLET       &COPR(PF$TECHN)='GREEN' //SET CURRENT COLOR
SCOLOR    MACRO      XID1,'GREEN'            //IN OPERATION
PLON3     MACRO      XID1,STA,PF(CLOC)
TCH110    TEST NE    PF(CLOC),PNTTC,TCH600   //NEW PAINT PROCESS?
          ADVANCE    PL(CYCLE)               //WORK ELEMENT
SCOLOR    MACRO      XID1,'WHITE'
TCH115    BLET       &COPR(PF$TECHN)='WHITE'
          TEST NE    PL(CMPEST),-1,TCH120  //HELPER DOESN'T ADJUST COUNT
          BLET       &TECHC(PF$CLOC)=&TECHC(PF$CLOC)-1
          UNLINK     INPRO,PRO220,1,CLOC$PF,PF$CLOC     //FREE ELEMENT
          PRIORITY   -1,YIELD
          PRIORITY   0
TCH120    RELEASE    PF(TECHN)+TCHNS       //NO;RELEASE
          BLET       PL(CMPEST)=0          //RESET HELPER INDICATOR
          REMOVE     ATECHS
*
          BLET       PL$WAIT=MP$WAIT$PL-(&ACNOOP(PF$MOD)-PL$ACMBRK) //OP TIME
          BLET       PF(NOOPR)=26
TCH130    TEST NE    PF(CLOC),PF(PF$NOOPR),TCH140
          LOOP       NOOPR$PF,TCH130
TCH140    BLET       PF(NOOPR)=PF(NOOPR)-20
          BLET       ML(TECHBD,PF$TECHN,PF$NOOPR)=ML(TECHBD,PF$TECHN,PF$NOOPR)+PL$WAIT
          BLET       ML(TECHBD,PF$TECHN,7)=ML(TECHBD,PF$TECHN,7)+PL$WAIT
          TEST NE    PF(OPER1),-99,TCH500  //TAGGED TO MOVE?
*
TCH160    TEST NE    CH(NOTCH),0,TCH300   //NO;ANY DELINQUENT UNITS?
          GATE LC    MATCH
```

```
         LOGIC S     MATCH
         BLET        &LOC(1)=PF$LOC1
         BLET        &LOC(2)=PF$LOC2
         BLET        &LOC(3)=PF$LOC3
         BLET        &LOC(4)=PF$LOC4
         BLET        &LOC(5)=PF$LOC5
         BLET        &LOC(6)=PF$LOC6
*         UNLINK      NOTCH,PRO305,1,BV$DLAY1,,TCH320
         BLET        PF(LCTR)=6
TCH161   TEST G      &LOC(PF$LCTR),0,TCH162                      //NON 0 LOC?
         UNLINK      NOTCH,PRO305,1,CLOC$PF,&LOC(PF$LCTR),TCH162 //FIND MATCH LOC
         BLET        PL(CMPEST)=0                                //ONE DISCOVERED
         TRANSFER    ,TCH163                                     //GET OUT OF LOOP
TCH162   LOOP        LCTR$PF,TCH161
         TRANSFER    ,TCH320
TCH163   GATE LC     MATCH
         UNLINK      PDLAY,PRO310,1,CLOC$PF,PF$LOC1,TCH170
         BLET        PF(OPNUM)=PF(LOC1)*1000
         LINK        APOOL,FIFO
TCH170   UNLINK      PDLAY,PRO310,1,CLOC$PF,PF$LOC2,TCH180
         BLET        PF(OPNUM)=PF(LOC2)*1000
         LINK        APOOL,FIFO
TCH180   UNLINK      PDLAY,PRO310,1,CLOC$PF,PF$LOC3,TCH190
         BLET        PF(OPNUM)=PF(LOC3)*1000
         LINK        APOOL,FIFO
TCH190   UNLINK      PDLAY,PRO310,1,CLOC$PF,PF$LOC4,TCH200
         BLET        PF(OPNUM)=PF(LOC4)*1000
         LINK        APOOL,FIFO
TCH200   UNLINK      PDLAY,PRO310,1,CLOC$PF,PF$LOC5,TCH210
         BLET        PF(OPNUM)=PF(LOC5)*1000
         LINK        APOOL,FIFO
TCH210   UNLINK      PDLAY,PRO310,1,CLOC$PF,PF$LOC6,TCH300
         BLET        PF(OPNUM)=PF(LOC6)*1000
TCH300   BLET        PL(CMPEST)=0                //ZERO OUT HELPER ID
         TEST NE     &MODID(PF$MOD),'570',TCH550 //570 HELPS
         TEST NE     &MODID(PF$MOD),'571',TCH550
TCH310   LINK        APOOL,FIFO                  //BACK IN TECH POOL
*
TCH320   LOGIC C     MATCH                       //DELAY NOT FOUND
         TRANSFER    ,TCH300                     //GO BACK ON POOL
*
TCH400   LINK        SWING,FIFO                  //STAGE 2ND OPS
TCH410   SPLIT       1,ATS000                    //PUT INTO PLAY
         TRANSFER    ,TCH160                     //GO LOOK FOR WORK
*
TCH500   UNLINK      SWING,TCH410,1,PLOC$PF,2 //RELEASE ALTER EGO
SCOLOR   MACRO       XID1,'BAC'
PLON     MACRO       XID1,TECHSTG
         BLET        PF(OPER1)=0              //RESET TAG
         BLET        PF(LCTR)=6               //SEARCH ASSIGNMENTS
TCH510   BLET        PF(DELRT)=PF(PF$LCTR+20) //FIND ASSIGNMENT
         TEST G      PF(DELRT),40,TCH520      //CHECK FOR CELLS & RTQ
         TEST LE     PF(DELRT),63,TCH520
         UNLINK      ACELLS,ATS100,1,DELRT$PF,(PF$DELRT-40)
TCH520   LOOP        LCTR$PF,TCH510
         LINK        IPOOL,FIFO               //ORIGINAL GOES INACTIVE
*
* TECHNICIAN HELPING LOGIC
*
TCH550   LOGIC C     SMSTA
         BLET        PF(LCTR)=0             //ZERO FOR SEARCH
TCH560   BLET        PF(LCTR)=PF(LCTR)+1
         BLET        PL(CMPEST)=0           //ZERO OUT
         TEST LE     PF(LCTR),6,TCH310      //END OF SEARCH?
         BLET        &SVAR=FN(TLOC2)        //NO;GET ELEMENT ASSIST#
         TEST G      &SVAR,0,TCH560         //NOT ASSIGNED HERE?
         SCAN E      ATECHS,CLOC$PF,&SVAR,TECHN$PF,CTR$PF,TCH560 //GET TECH#
         SCAN E      ATECHS,CLOC$PF,&SVAR,CMPEST$PL,CMPEST$PL
         TEST G      PL(CMPEST),0,TCH560
```

```
         PREEMPT    PF(CTR)+TCHNS,,TCH110,CYCLE$PL                    //DELAY TECH
         SCAN E     ATECHS,CLOC$PF,&SVAR,CYCLE$PL,CYCLE$PL //REMAINING CYCLE
         SCAN E     ATECHS,CLOC$PF,&SVAR,CLOC$PF,CLOC$PF    //GROUP #
         BLET       PL(CYCLE)=PL(CYCLE)/2.0                 //ADJUST
         ALTER      ATECHS,1,CYCLE$PL,PL$CYCLE,CLOC$PF,&SVAR //PASS CYCLE TIME
         ALTER      ATECHS,1,CMPEST$PL,0,CLOC$PF,&SVAR      //PASS CYCLE TIME
         RETURN     PF(CTR)+TCHNS
         BLET       PL(CMPEST)=-1
         BLET       PF(CLOC)=&SVAR
         TRANSFER   ,TCH100
*
* NEW PAINT PROCESS
*
 TCH600  BLET       PF(LCTR)=3              //3 LOOPS
 TCH610  ADVANCE    0
 PLON    MACRO      XID1,PTPASS
         ADVANCE    PL(CYCLE)               //LOOP PAST TWO LOADS
         LOOP       LCTR$PF,TCH610
 PLON3   MACRO      XID1,STA,PF(CLOC)
 SCOLOR  MACRO      XID1,'WHITE'
         TRANSFER   ,TCH115
*-------------------
* PROCESS SUROUTINES
*-------------------
 PRO000  BLET       PF(LCTR)=0              //LOOP COUNTER TO 0
 PRO100  BLET       PF(LCTR)=PF(LCTR)+1 //BUMP
         TEST LE    PF(LCTR),6,PRO300       //END OF SEARCH?
         TRANSFER   ,FN(TLOC1)
*
 PRO110  ADVANCE    0
 FNDTCH  MACRO      LOC1$PF
         TRANSFER   ,PRO200
*
 PRO120  ADVANCE    0
 FNDTCH  MACRO      LOC2$PF
         TRANSFER   ,PRO200
*
 PRO130  ADVANCE    0
 FNDTCH  MACRO      LOC3$PF
         TRANSFER   ,PRO200
*
 PRO140  ADVANCE    0
 FNDTCH  MACRO      LOC4$PF
         TRANSFER   ,PRO200
*
 PRO150  ADVANCE    0
 FNDTCH  MACRO      LOC5$PF
         TRANSFER   ,PRO200
*
 PRO160  ADVANCE    0
 FNDTCH  MACRO      LOC6$PF
 PRO200  BLET       &TECHC(PF$CLOC)=&TECHC(PF$CLOC)+1       //FOUND TECH
 PRO210  LINK       INPRO,FIFO                              //IN PROCESS
 PRO220  TRANSFER   ,PF(SUBR)+1
*
 PRO300  MARK       WAIT$PL                 //NO TECHS - COLLECT WAIT
         BLET       PL(ACMBRK)=&ACNOOP(PF$MOD)  //ACCUM BREAK TIME
         LINK       NOTCH,FIFO              //NO TECH CHAIN
 PRO305  LOGIC C    MATCH                   //1ST DELAY FOUND/FREE MATCH
* PRO305  SPLIT      1,NDL000
         LINK       PDLAY,FIFO              //AWAIT 2ND CALL
 PRO310  ALTERUCH E APOOL,1,CLOC$PF,PF$CLOC,OPNUM$PF,PF$CLOC*1000 //PASS ID
         ALTERUCH E APOOL,1,CYCLE$PL,PL$CYCLE,OPNUM$PF,PF$CLOC*1000 //PASS CYCLE
*         ALTERUCH E APOOL,1,OPNUM$PF,PF$OPNUM,CLOC$PF,PF$CLOC     //PASS OPNUM
         BLET       PL(CMPEST)=PL$CYCLE+AC1         //ESTIMATE COMPLETION
         UNLINK     APOOL,TCH100,1,OPNUM$PF,PF$CLOC*1000   //GET TECH
         BLET       PF(OPNUM)=-1                    //STOP PICKUP
         TRANSFER   ,PRO200
*-----------------------------------------------------------------------
```

```
* ASSEMBLY LAUNCH
*-----------------------------------------------------------------
 LIN000   SPLIT      1,CLB000                     //CREATE JHOOK LOAD BARS
          BLET       &J=0                         //FOR SHIP SCHEDULE POINTER
          BLET       &SSP=&JHSPD(1)
 SPSPD    MACRO      JHK6,&SSP
 SPSPD    MACRO      JHK7,&SSP
* READ SCHEDULE LINEUP IN
          BLET       PF(CVSEC)=1                  //SET STARTING POINT
 LIN010   BLET       PF(SEQNM)=PF(SEQNM)+1        //BUMP SCHEDULE
          TEST E     MX(SHIPS,PF$SEQNM,1),0,*+2   //SCHEDULE EOF?
*          BLET       PF(SEQNM)=1                  //YES;RESET
          TERMINATE
          BLET       PF(ENGINE)=MX(SHIPS,PF$SEQNM,1) //GET ENGINE
          BLET       PF(TSEQN)=MX(SHIPS,PF$SEQNM,4)  //TRUCK#
          BLET       PF(LCTR)=MX(SHIPS,PF$SEQNM,2)   //# IN RUN
 LIN020   BLET       PF(SSEQN)=PF(SSEQN)+1           //GRAND SEQ.
          GATE SE    200                             //1ST OPEN
          SPLIT      1,BLK000                        //YES;CREATE BLOCK
          GATE SNE   200                              //WAIT FOR IT
          LOOP       LCTR$PF,LIN020                  //CONTINUE W/ RUN
          TRANSFER   ,LIN010                         //GET NEXT RUN
*
* BLOCK LINE - DEPT. 566
*
 BLK000   ENTER      200
          LINK       ASMLD,FIFO                   //AWAIT ASSEMBLY LOAD
 LDBLK    LEAVE      200
          TERMINATE
*---------------------------------------------------------------
* J-HOOK ASSEMBLY LINE
*---------------------------------------------------------------
 CLB000   BLET       PF(LCTR)=&LBCTJHK            //J-HOOK LOAD BARS
          BLET       PF(CVSEC)=9                  //STARTING POINT
 CLB010   GATE LC    PF(CVSEC)+JHOOK              //FIRST OPEN?
          SPLIT      1,JHK000                     //CREATE CARRIER
          ADVANCE    .5                           //DELAY
          LOOP       LCTR$PF,CLB010
          TERMINATE
*
 JHK000   GATE LC    PF(CVSEC)+JHOOK              //CLEARANCE SECTION
          LOGIC S    PF(CVSEC)+JHOOK
          ENTER      PF(CVSEC)+JHOOK              //ZONE
 CREATE   MACRO      JHLB,XID1
 WRITEO   MACRO      JID,XID1,'EMPTY'
 SCOLOR   MACRO      XID1,'WHITE'
 PLON3    MACRO      XID1,JHK,PF(CVSEC)
 JHK010   ADVANCE    8.0/&FSP                     //CLEAR LOAD BAR
          LOGIC C    PF(CVSEC)+JHOOK
          ADVANCE    (&APATH(PF$CVSEC)-8.0)/&FSP  //TRAVEL
          TEST LE    PF(CVSEC),5,JHK020           //STILL ON FAST TRACK
          SEIZE      PF(CVSEC)+JHOOK              //YES;GRAB STATION
          ADVANCE    PL(CYCLE)*&ASMMAX/&SSP       //PROPORTION
          RELEASE    PF(CVSEC)+JHOOK
 JHK020   BLET       PF(PLOC)=PF(CVSEC)           //UPDATE
          BLET       PF(CVSEC)=PF(CVSEC)+1        //BUMP
          TEST NE    PF(CVSEC),10,JHK060          //END OF CONV.
          TEST NE    PF(CVSEC),8,JHK050           //@END OF JHOOK?
          GATE LC    PF(CVSEC)+JHOOK              //CLEARANCE SECTION
          LOGIC S    PF(CVSEC)+JHOOK              //CLEARANCE SECTION
          ENTER      PF(CVSEC)+JHOOK              //ZONE
          LEAVE      PF(PLOC)+JHOOK
 PLON3    MACRO      XID1,JHK,PF(CVSEC)
          TEST G     PF(CVSEC),5,JHK010           //FAST TRACK?
          TEST NE    PF(CVSEC),6,JHK040           //AT SLOW CHAIN?
          TEST NE    PF(CVSEC),9,JHK010           //FAST RETURN CHECK
 JHK030   BLET       PL(CYCLE)=8.0/&SSP           //CLEARANCE @ SLOW
          BLET       PL(ITIME)=1
 JHK031   ADVANCE    PL(CYCLE)
```

```
         LOGIC C     PF(CVSEC)+JHOOK
         BLET        PL(CYCLE)=(&APATH(PF$CVSEC)-8.0)/&SSP    //TRAVEL @ SLOW
         BLET        PL(ITIME)=0
JHK035   ADVANCE     PL(CYCLE)
         TRANSFER    ,JHK020                      //KEEP LOOKING
*
JHK040   GATE LC     212
         SELECT NU   PTR$PF,212,260               //SELECT A FACILITY
         SEIZE       PF(PTR)                      //GRAB IT
JHKTOT   BLET        PL(CYCLE)=14.0/&SSP          //CLEARANCE @ SLOW
         BLET        PL(ITIME)=1
         TEST NE     &EFAM(&ECLASI(PF$ENGINE)),&PRVENG,JHK031
JHKCHG   ADVANCE     &JHKCO
         BLET        &JHKCOTIM=&JHKCOTIM+&JHKCO
         BLET        &PRVENG=&EFAM(&ECLASI(PF$ENGINE))
         TRANSFER    ,JHK031
*
* TRANSFER TO MAIN CONVEYOR
*
JHK050   GATE LS     (PF(CVSEC)+JHOOK),JHK055     //END OF LINE STOPPED?
SPSPD    MACRO       JHK6,0
SPSPD    MACRO       JHK7,0
         LOGIC S     212
         FUNAVAIL    212-260                      //STOP SLOW TRACK
         GATE LC     PF(CVSEC)+JHOOK
SPSPD    MACRO       JHK6,&SSP
SPSPD    MACRO       JHK7,&SSP
         FAVAIL      212-260                      //STOP SLOW TRACK
JHK054   LOGIC C     212
JHK055   GATE LC     PF(CVSEC)+JHOOK              //CLEARANCE SECTION
         LOGIC S     PF(CVSEC)+JHOOK
         ENTER       PF(CVSEC)+JHOOK              //ZONE
         LEAVE       PF(PLOC)+JHOOK
         RELEASE     PF(PTR)
PLON3    MACRO       XID1,JHK,PF(CVSEC)
         ADVANCE     &APATH(PF$CVSEC)/&FSP        //TRAVEL
         GATE LC     451                          //J-HOOK DELAY?
         SEIZE       210                          //JHOOK UNLOADER
         ADVANCE     &JHKUL/&PERF(1)              //UNLOAD TIME
         RELEASE     210
         BLET        &INPROC=&INPROC+1            //COUNT ENGINE IN PROCESS
         BLET        &EPROD(1)=&EPROD(1)+1        //COUNT ENGINE RATE
         BLET        &PRORATE(1)=&PRORATE(1)+1
BARG     MACRO       RT1,TOP,&PRORATE(1)
         ENTER       EWIPQ
         ENTER       TOTALQ
         MARK        LAPTIM$PL                    //START TIMING
         LOGIC S     ASMUL                        //SIGNAL UNLOAD
         LINK        ASMUL,FIFO                   //AWAIT INTERFACE
ULASM    LOGIC C     PF(CVSEC)+JHOOK
SCOLOR   MACRO       XID1,'WHITE'
WRITEO   MACRO       JID,XID1,'EMPTY'
         TRANSFER    ,JHK020
*
JHK060   BLET        PF(CVSEC)=1                  //AT FIRST
         GATE LC     PF(CVSEC)+JHOOK              //CLEARANCE SECTION
         LOGIC S     PF(CVSEC)+JHOOK
         ENTER       PF(CVSEC)+JHOOK              //ZONE
         LEAVE       PF(PLOC)+JHOOK
PLON3    MACRO       XID1,JHK,PF(CVSEC)
         ADVANCE     &APATH(PF$CVSEC)/&FSP        //TRAVEL
         GATE SNE    200                          //BLOCK THERE?
         SCANUCH G   ASMLD,SSEQN$PF,0,SSEQN$PF,SSEQN$PF
         SCANUCH E   ASMLD,SSEQN$PF,PF$SSEQN,ENGINE$PF,ENGINE$PF
         SCANUCH E   ASMLD,SSEQN$PF,PF$SSEQN,TSEQN$PF,TSEQN$PF
         SCANUCH E   ASMLD,SSEQN$PF,PF$SSEQN,SEQNM$PF,SEQNM$PF
         BLET        PL(CYCLE)=440.0/&PRODVOL(1)/&PERF(1)
         SEIZE       PF(CVSEC)+JHOOK
         ADVANCE     PL(CYCLE)*&ASMMAX/&SSP       //UNLOAD TIME
```

```
          RELEASE    PF(CVSEC)+JHOOK
          UNLINK     ASMLD,LDBLK,1                  //GRAB BLOCK
 SCOLOR   MACRO      XID1,'GREEN'
 WRITEO   MACRO      JID,XID1,&PARTNO(PF$ENGINE)
          LOGIC C    PF(CVSEC)+JHOOK
          TRANSFER   ,JHK020
*----------------------------------------
* Initialize empty load bars in main loop.
*----------------------------------------
 CNV000   GATE SNF   18                            //ZONE FULL?
          SPLIT      1,CNV010                        //NO;CREATE
          ADVANCE    .12                             //CLEAR
          LOOP       LCTR$PF,CNV000
          TERMINATE
*
 CNV010   ENTER      18
CREATE    MACRO      LBR,XID1
 WRITEO   MACRO      LBRID,XID1,'EMPTY'
SCOLOR    MACRO      XID1,'WHITE'
 BARG     MACRO      PQ1,TOP,100.0*S(18)/(S(18)+R(18))
*
GOPF1     ADVANCE    0
PLON      MACRO      XID1,BB18
          ADVANCE    11.22
GOPF2     LINK       18,FIFO,GOPF2A
GOPF2A    SEIZE      SPF2
          ENTER      SPF2
PLON      MACRO      XID1,PF2
          ADVANCE    .1
          LEAVE      18
 BARG     MACRO      PQ1,TOP,100.0*S(18)/(S(18)+R(18))
          ADVANCE    .94
GOPF3     RELEASE    SPF2
          UNLINK     18,GOPF2A,1
PLON      MACRO      XID1,PF3
          ADVANCE    .90
*
          ENTER      SPL
          LEAVE      SPF2
PLON      MACRO      XID1,PL
          ADVANCE    .17
*
* Now wait for a raw engine to be ready to be transferred.
* Wait on switch, while matching engine is transferred.
*
          GATE LC    SPF1
          GATE LS    ASMUL,BLU100         //GO TEST BLUBIRD IF NO J-HOOK
  GOPF4A  GATE LS    ASMUL                //AWAIT JHOOK ENGINE?
          SCANUCH G  ASMUL,SSEQN$PF,0,SSEQN$PF,SSEQN$PF   //GET GRAND SEQ#
          SCANUCH E  ASMUL,SSEQN$PF,PF$SSEQN,ENGINE$PF,ENGINE$PF //GET ENGINE#
          SCANUCH E  ASMUL,SSEQN$PF,PF$SSEQN,LAPTIM$PL,LAPTIM$PL
          SCANUCH E  ASMUL,SSEQN$PF,PF$SSEQN,TSEQN$PF,TSEQN$PF
          SCANUCH E  ASMUL,SSEQN$PF,PF$SSEQN,SEQNM$PF,SEQNM$PF
          UNLINK     ASMUL,ULASM,1        //RELEASE
          LOGIC C    ASMUL                //AWAIT NEXT ENGINE
 WRITEO   MACRO      LBRID,XID1,&PARTNO(PF$ENGINE)
SCOLOR    MACRO      XID1,&ECLR(PF$ENGINE)
*
* MAIN DELIVERY CONVEYOR
*
 GOPF5    SEIZE      SPO
          ENTER      SPO
PLON      MACRO      XID1,PO
          ADVANCE    .1
          RELEASE    SPO
          LEAVE      SPL
          LOGIC C    SPF1
          ADVANCE    .55
          LINK       211,FIFO,CNV020
```

```
CNV020  SEIZE      211                        //LEAK TESTER
        BLET       PF(RJCT)=0                 //ZERO REJECT PARM
        BLET       PF(PCODE)=0                //ZERO OUT PROCESS CODE
        TEST NE    PF(DELRT),568,RPH000       //COMING FROM FLOOR
        BLET       PF(DELRT)=0                //ZERO OUT
        TEST G    &BTRIM(PF$ENGINE),0,CNV030  //BLUEBIRD?
        BLET       PF(PCODE)=PNTSYS           //YES;DIRECT TO PAINT
        TRANSFER   ,CNV050                    //PROCEED
CNV030  TEST G    &CTRIM(PF$ENGINE),0,CNV040  //CARB TRIM
        BLET      &OPCNT(PF$ENGINE)=&OPCNT(PF$ENGINE)+1
        BLET       PF(CTR)=&OPCNT(PF$ENGINE)  //SAVE
        BLET       PF(PCT)=&COPTN(PF$ENGINE)*100
        TRANSFER   SBR,RPCT00,SUBR$PF         //DETERMINE %
        BLET       PF(DELRT)=PF(RJCT)
        BLET       PF(RJCT)=0
CNV040  ADVANCE    &LEAKTST(1)/&PERF(1)       //NO LEAK TEST
        BLET       &GBCNT(1)=&GBCNT(1)+1      //REJECT COUNT
        BLET       PF(CTR)=&GBCNT(1)          //SAVE COUNT
        BLET       PF(PCT)=&LEAKRJ(1)*100     //SAVE PCT
        TRANSFER   SBR,RPCT00,SUBR$PF         //DETERMINE REJECT
        TEST E     PF(RJCT),0,RPH000          //PASS TEST?
        TEST E     PF(DELRT),0,RPH000         //YES;CARB TRIM JOB?
        BLET       PF(PCODE)=CLTEST           //REST TO TEST
CNV050  BLET       PF(CVSEC)=1                //STARTING SECTION
        GATE LC    PF(CVSEC)                  //ZONE CLEAR
        LOGIC S    PF(CVSEC)                  //SHUT OFF
        BLET       &DUM='569'                 //GOING 569
        TRANSFER   SBR,FNDMOD,SUBR$PF         //FIND MOD #
        ENTER      PF(CVSEC)                  //MERGE ZONE
PLON3   MACRO      XID1,MBB,PF(CVSEC)
        ADVANCE    ML(CSECT,PF$CVSEC,12)      //MERGE ZONE
        LOGIC C    PF(CVSEC)                  //CLEARANCE
        LEAVE      SPO                        //FREE PREVIOUS
        RELEASE    211
        UNLINK     211,CNV020,1
PLON3   MACRO      XID1,BB,PF(CVSEC)
        TRANSFER   ,BBD060
*-------------------------------------------
* BACKBONE DELIVERY CONVEYOR
*-------------------------------------------
BBD000  GATE LC    PF(CVSEC)                  //SWITCH CLEAR
        LOGIC S    PF(CVSEC)                  //1@TIME
        TEST NE    &BBMRG(PF$CVSEC),1,BBD050  //MERGE ZONE?
        ENTER      PF(CVSEC)                  //NO;GET ZONE
PLON3   MACRO      XID1,BB,PF(CVSEC)          //GET ON PATH
        ADVANCE    .12                        //CLEARANCE ZONE
        TEST E     PF(CVSEC),18,*+2
        TRANSFER   SBF,BBD090,SUBR$PF
        LOGIC C    PF(CVSEC)                  //OPEN CLEARANCE
        LEAVE      PF(PLOC)                   //FREE PREVIOUS
        TEST E     PF(PLOC),3,*+2
        UNLINK     PF(PLOC),BBD000,1
        TEST E     PF(PLOC),33,*+2
        UNLINK     PF(PLOC),TLC3320,1
        ADVANCE    ML(CSECT,PF$CVSEC,1)-.12   //ZONE
BBD010  TEST NE    PF(CVSEC),1,BBD0100        //GO TO RC?
        TEST NE    PF(CVSEC),3,BBD0300        //ALL TEST CELLS
        TEST NE    PF(CVSEC),4,BBD0400        //ALL TEST CELLS
        TEST NE    PF(CVSEC),5,BBD0500        //TEST CELLS 7-18
*       TEST NE    PF(CVSEC),9,BBD0900        //SPECIAL
*       TEST NE    PF(CVSEC),11,BBD1100       //SPECIAL
        TEST NE    PF(CVSEC),13,TLC000        //EXIT TEST CELL LOOP?
        TEST NE    PF(CVSEC),16,BBD1600       //BACKBONE LIMIT CHECK
        TEST NE    PF(CVSEC),17,BBD1700       //EXIT FOR PAINT, TRIM, EMPTIES
        TEST NE    PF(CVSEC),18,GOPF2         //EMPTY LOAD BAR RETURN
BBD020  BLET       PF(PLOC)=PF(CVSEC)         //BUMP PREVIOUS LOC
        BLET       PF(CVSEC)=PF(CVSEC)+1      //NEXT ZONE
        TRANSFER   ,BBD000
*
```

```
BBD050   ENTER      PF(CVSEC)                   //MERGE ZONE
PLON3    MACRO      XID1,BBM,PF(CVSEC)
         ADVANCE    ML(CSECT,PF$CVSEC,11)       //MERGE ZONE
         TEST E     PF(CVSEC),18,*+2
         TRANSFER   SBR,BBD090,SUBR$PF
BBD055   LOGIC C    PF(CVSEC)                   //CLEARANCE
         LEAVE      PF(PLOC)                    //FREE PREVIOUS
         TEST E     PF(PLOC),3,*+2
         UNLINK     PF(PLOC),BBD000,1
         TEST E     PF(PLOC),33,*+2
         UNLINK     PF(PLOC),TLC3320,1
PLON3    MACRO      XID1,BB,PF(CVSEC)
BBD060   ADVANCE    ML(CSECT.PF$CVSEC.1)        //ZONE TRAVEL
         TRANSFER   ,BBD010
*
BBD090   LOGIC C    BBSWT
         UNLINK     PF(PLOC),BBD1710,1
         LEAVE      BACKBCNT
BARG     MACRO      PQ1,TOP,100.0*S(18)/(S(18)+R(18))
         TEST L     S(BACKBCNT),&BBLIM,PF(SUBR)+1
         LOGIC C    BACKBCNT
         TRANSFER   ,PF(SUBR)+1
*
BBD0100  ENTER      BACKUP
         GATE LC    RECR1                       //GRAB SWITCH
         LOGIC S    RECR1
         GATE SNF   RECR1O
         TEST E     BV(NOBKUP),1,*+2            //BACKUP CONDITION?
         UNLINK     BACKUP,RCL1005,1            //YES;RELEASE TO ATTIC
         TEST E     PF(PCODE),CLTEST,BBD0110
         GATE SE    RECR1,RCL1000               //ANYTHING IN ATTIC?
BBD0105  GATE LC    COUNT,BBD0120
BBD0106  BLET       XF(COUNT)=XF(COUNT)+1
         TEST GE    XF(COUNT),&MAX,*+2
         LOGIC S    COUNT
         JOIN       GCLTEST
BBD0110  GATE LC    PF(CVSEC)+1                //UPDATE
         LEAVE      BACKUP
         LOGIC C    RECR1
         TRANSFER   ,BBD020
*
BBD0120  GATE SE    RECR1,RCL1000               //GO RECIRC IF ATTIC NOT EMPTY
         GATE SNF   BACKUP,RCL1000              //GO ATTIC IF BACKUP FULL
         GATE SNF   BACKUP
         GATE LC    COUNT
         TRANSFER   SIM,BBD0105,RCL1000         //RETEST IF TRUE/GO ATTIC IF NOT
*
BBD0300  BLET       PF(PLOC)=PF(CVSEC)          //BUMP PREVIOUS LOC
         BLET       PF(CVSEC)=PF(CVSEC)+1       //NEXT ZONE
         LINK       PF(PLOC),FIFO,BBD000
         TRANSFER   ,BBD000
*
BBD0400  TEST E     PF(PCODE),CLTEST,BBD020     //TEST CELL CODE?
         SCAN MIN   GCLTEST,TSEQN$PF,,TSEQN$PF,DELRT$PF,CEL006 //FIND LOWEST TRK GRID#
         TEST LE    PF$TSEQN,PF$DELRT,BBD020    //AM I LOWEST?
         TRANSFER   ,CEL006
*
BBD0500  TEST E     PF(PCODE),CLTEST,BBD020     //TEST CELL CODE?
         GATE SNF   36,BBD020                   //ZONE CLEAR?
         SCAN MIN   GCLTEST,TSEQN$PF,,TSEQN$PF,DELRT$PF,CEL013 //FIND LOWEST TRK GRID#
         TEST LE    PF$TSEQN,PF$DELRT,BBD020    //AM I LOWEST?
         TRANSFER   ,CEL013
*
BBD1600  LINK       BACKBCNT,FIFO,BBD1610       //ACCUMULATE BEHIND STOP
BBD1610  GATE LC    BACKBCNT                    //STOP OPEN?
         ENTER      BACKBCNT                    //GRAB ZONE
         TEST GE    S(BACKBCNT),&BBLIM,BBD1620
         LOGIC S    BACKBCNT
BBD1620  ADVANCE    .01
```

```
         UNLINK     BACKBCNT,BBD1610,1
         TRANSFER   ,BBD020
*
BBD1700  LINK       PF(CVSEC),FIFO,BBD1710        //ACCUMULATE BEHIND STOPS
BBD1710  GATE LC    BBSWT
         LOGIC S    BBSWT
         TEST NE    PF(PCODE),0,BBD020            //EMPTY?
         GATE SNF   SSTG0                         //NO;STAGE POSITION OPEN
         GATE LC    SSTG0                         //YES;INDEX IN
         LOGIC S    SSTG0                         //ONE AT TIME
         ENTER      SSTG0                         //ENTER ZONE
PLON     MACRO      XID1,STG0
         ADVANCE    .12                           //INDEX IN
         LOGIC C    BBSWT
         UNLINK     PF(CVSEC),BBD1710,1
         LOGIC C    SSTG0
         LEAVE      PF(CVSEC)                     //LEAVE PREVIOUS ZONE
         LEAVE      BACKBCNT                      //LEAVE BACKBONE
         TEST L     S(BACKBCNT),&BBLIM,*+2        //LESS THAN CHOKE LIMIT?
         LOGIC C    BACKBCNT                      //YES; FREE ZONE
         ADVANCE    .40                           //INDEX
         LINK       SSTG0,FIFO,BBD1730
BBD1730  TEST NE    PF(PCODE),PNTSYS,PST000       //DESTINED TO PAINT
         TRANSFER   ,TRM000
*
* TEST CELL LEG
*
TLC000   BLET       PF(PLOC)=PF(CVSEC)            //KEEP PREV
         BLET       PF(CVSEC)=31
         TEST G     PF(ENGINE),0,TLC010           //EMPTY?
         TEST G     PF(PCODE),10,TLC010           //10.5/12.5?
         BLET       PF(PCODE)=PF(PCODE)-10        //YES;NOW AVAILABLE
         JOIN       PF(PCODE)                     //IN GROUP
TLC010   GATE LC    PF(CVSEC)                     //SWITCH CLEAR
         LOGIC S    PF(CVSEC)                     //1@TIME
         TEST NE    &BBMRG(PF$CVSEC),1,TLC050     //MERGE ZONE?
         ENTER      PF(CVSEC)                     //NO;GET ZONE
PLON3    MACRO      XID1,TL,PF$CVSEC-30           //GET ON PATH
         ADVANCE    .13                           //CLEARANCE ZONE
         LOGIC C    PF(CVSEC)                     //OPEN CLEARANCE
         LEAVE      PF(PLOC)                      //FREE PREVIOUS
         TEST E     PF(PLOC),32,*+2
         UNLINK     PF(PLOC),TLC3205,1
         ADVANCE    ML(CSECT,PF$CVSEC,1)-.13      //ZONE
TLC020   TEST NE    PF(CVSEC),32,TLC3200          //GO TO RETORQ?
         TEST NE    PF(CVSEC),33,TLC3300          //EXIT TL OR REPAIRS
TLC030   BLET       PF(PLOC)=PF(CVSEC)            //BUMP PREVIOUS LOC
         BLET       PF(CVSEC)=PF(CVSEC)+1         //NEXT ZONE
         TRANSFER   ,TLC010
*
TLC050   ENTER      PF(CVSEC)                     //MERGE ZONE
PLON3    MACRO      XID1,TLM,PF$CVSEC-30
         ADVANCE    ML(CSECT,PF$CVSEC,11)         //MERGE ZONE
TLC055   LOGIC C    PF(CVSEC)                     //CLEARANCE
         LEAVE      PF(PLOC)                      //FREE PREVIOUS
         TEST E     PF(PLOC),32,*+2
         UNLINK     PF(PLOC),TLC3205,1
PLON3    MACRO      XID1,TL,PF$CVSEC-30
TLC060   ADVANCE    ML(CSECT,PF$CVSEC,1)          //ZONE TRAVEL
         TRANSFER   ,TLC020
*
TLC3200  LINK       PF(CVSEC),FIFO,TLC3205
TLC3205  GATE SNF   RTORKQ,TLC030                 //ZONE FULL?
         BLET       PF(PLOC)=PF(CVSEC)            //SAVE PREVIOUS LOCATION
         TEST NE    PF(ENGINE),0,RTQ000           //EMPTY EXITS
         TEST G     PF(PCODE),REPAIRS,TLC030      //REPAIRS STAY ON LOOP?
         TEST L     PF(PCODE),10,TLC030           //1ST PASS 10.5/12.5 STAY ON
         SCAN MIN   PF$PCODE,TSEQN$PF,,TSEQN$PF,DELRT$PF,TLC3210
         TEST LE    PF$TSEQN,PF$DELRT,TLC030
```

```
TLC3210  GATE SNF   (294+PF$PCODE),TLC030
         ENTER      (294+PF$PCODE)
BARG4    MACRO      PQ,(PF$PCODE-3),TOP,100.0*S(294+PF$PCODE)/(S(294+PF$PCODE)+_
                     R(294+PF$PCODE))
         TRANSFER   ,RTQ000
*
TLC3300  BLET       PF(PLOC)=PF(CVSEC)           //SAVE PREVIOUS LOCATION
         TEST E     PF(PCODE),REPAIRS,TLC3310    //REPAIR PC?
         GATE SNF   REPRQ,TLC3310
         SCAN MIN   GREPAIRS,TSEQN$PF,,TSEQN$PF,DELRT$PF,RPR000
         TEST LE    PF$TSEQN,PF$DELRT,TLC3310
         REMOVE     GREPAIRS
         TRANSFER   ,RPR000
*
TLC3310  ADVANCE    .08
         LINK       PF(PLOC),FIFO,TLC3320
TLC3320  BLET       PF(CVSEC)=3                  //CONV. SECTION
         GATE LC    PF(CVSEC)
         LOGIC S    PF(CVSEC)
         ENTER      PF(CVSEC)
PLON     MACRO      XID1,MBB3
         ADVANCE    .14
         TRANSFER   ,BBD055
*-----------------------------------------
* BLUEBIRD AND REAR PTO FLOOR QUEUE
*-----------------------------------------
BLU000   ADVANCE    5                      //5 MIN DELIVERY ASSUMED
         QUEUE      BLUBFQ                 //FLOOR QUEUE
BARG     MACRO      BBFG,RIGHT,Q(BLUBFQ)
WRITE    MACRO      BBFM,Q(BLUBFQ)
         LOGIC S    BLUBFQ                 //INDICATE HERE
         LINK       BLUBFQ,FIFO
BLU010   TEST E     CH(BLUBFQ),0,*+2        //EMPTY YET?
         LOGIC C    BLUBFQ                 //BLUEBIRD NOT THERE?
         DEPART     BLUBFQ
BARG     MACRO      BBFG,RIGHT,Q(BLUBFQ)
WRITE    MACRO      BBFM,Q(BLUBFQ)
         TERMINATE
*
BLU100   GATE LS    BLUBFQ,GOPF4A          //BLUEBIRD THERE?
         TEST E     BV(BBMTR),1,GOPF4A     //YES;CONVEYOR FULL?
         SCANUCH G  BLUBFQ,SSEQN$PF,0,SSEQN$PF,SSEQN$PF    //GET GRAND SEQ#
         SCANUCH E  BLUBFQ,SSEQN$PF,PF$SSEQN,ENGINE$PF,ENGINE$PF //GET ENGINE#
         SCANUCH E  BLUBFQ,SSEQN$PF,PF$SSEQN,LAPTIM$PL,LAPTIM$PL
         SCANUCH E  BLUBFQ,SSEQN$PF,PF$SSEQN,TSEQN$PF,TSEQN$PF
         SCANUCH E  BLUBFQ,SSEQN$PF,PF$SSEQN,SEQNM$PF,SEQNM$PF
         UNLINK     BLUBFQ,BLU010,1         //RELEASE
SCOLOR   MACRO      XID1,&ECLR(PF$ENGINE)
WRITEO   MACRO      LBRID,XID1,&PARTNO(PF$ENGINE)
         BLET       PF(DELRT)=568           //TAG TO GO TO 568
         TRANSFER   ,GOPF5
*-----------------------------------------
* 568 REPAIR AND TRIM
*-----------------------------------------
RPH000   BLET       &DUM='568'            //NEW MODULE
         TRANSFER   SBR,FNDMOD,SUBR$PF    //FIND CORRESPONDING #
         GATE LC    260                   //CLEARING ZONE
         LOGIC S    260                   //STOP
         RELEASE    211                   //FREE TEST ZONE
         UNLINK     211,CNV020,1
         TEST E     PF(RJCT),1,RPH100     //NEED REPAIR?
SCOLOR   MACRO      XID1,'RED'
         SELECT SNF CLOC$PF,261,263       //OPEN REPAIR SPUR
RPH002   ENTER      PF(CLOC)
PLON3    MACRO      XID1,RS,PF(CLOC)-260
         ADVANCE    .28                   //CLEARING
         LOGIC C    260                   //CLEARED
         LEAVE      SPO
         ADVANCE    1.0                    //TO SPUR
```

```
         BLET      PL(CYCLE)=&LEAKRPR(1) //REPAIR TIME
         TRANSFER  SBR,PRO000,SUBR$PF    //PROCESS
SCOLOR   MACRO     XID1,&ECLR(PF$ENGINE)
         GATE LC   LEAKQ                 //EXIT CLEAR?
         LOGIC S   LEAKQ                 //YES;TIE UP
         LEAVE     PF(CLOC)
PLON3    MACRO     XID1,RF,PF(CLOC)-260
         ADVANCE   .41+.20*(263-PF$CLOC)
RPH005   ENTER     LEAKQ
         LOGIC C   LEAKQ
PLON     MACRO     XID1,LEAKTQ
         ADVANCE   1.38
         LINK      LEAKQ,FIFO,RPH010
RPH010   ENTER     LKTST2               //LEAK TEST ZONE
PLON     MACRO     XID1,LKT2            //PATH
         ADVANCE   .14
         LEAVE     LEAKQ
         BLET      PF(CLOC)=LKTEST2     //NEW STATION
         BLET      PL(CYCLE)=&LEAKTST(2)
         TRANSFER  SBR,PRO000,SUBR$PF   //PROCESS
         BLET      PF(PCODE)=CLTEST     //CELL TEST IS NEXT
         BLET      PF(DELRT)=0          //ZERO OUT DEL. ROUTE
         BLET      PF(PLOC)=270         //LEAK TEST #2
         BLET      PF(CVSEC)=19         //SET ZONE
         GATE LC   PF(CVSEC)            //MERGE CLEAR
         LOGIC S   PF(CVSEC)            //YES; TIE UP
         ENTER     PF(CVSEC)
         BLET      &DUM='569'
         TRANSFER  SBR,FNDMOD,SUBR$PF
         UNLINK    LEAKQ,RPH010,1
PLON     MACRO     XID1,BB19
         ADVANCE   .12
         BLET      &EPROD(2)=&EPROD(2)+1    //COUNT ENGINE IN PROCESS
         LOGIC C   PF(CVSEC)
         LEAVE     PF(PLOC)
         ADVANCE   .92                  //CLEAR
         BLET      PF(PLOC)=PF(CVSEC)
         BLET      PF(CVSEC)=1
         TRANSFER  ,BBD000              //BACK TO MAIN
*--------------------------------------------
* BLUEBIRD & COMPRESSOR TRIM
*--------------------------------------------
RPH100   ADVANCE   0
PLON     MACRO     XID1,BCTRM0
         ADVANCE   .28                  //CLEARANCE
         LOGIC C   260                  //CLEAR ZONE
         ENTER     271
         LEAVE     SPO
RPH110   ADVANCE   0
PLON     MACRO     XID1,BCTRM1
         ADVANCE   .79                  //TIME
         BLET      PF(CLOC)=271         //START OF TRIM LINE
         GATE LC   PF(CLOC)             //@ ASM STATION
         LOGIC S   PF(CLOC)
         BLET      PL(CYCLE)=&CTRIM(PF$ENGINE)/6/&PERF(PF$MOD)
         TEST E    PL(CYCLE),0,*+2      //NOT CARB?
         BLET      PL(CYCLE)=&BTRIM(PF$ENGINE)/6/&PERF(PF$MOD)
RPH120   TRANSFER  SBR,PRO000,SUBR$PF   //PROCESS
         BLET      PF(PLOC)=PF(CLOC)    //SAVE PREVIOUS
         BLET      PF(CLOC)=PF(CLOC)+1  //BUMP LOCATION
         TEST NE   PF(CLOC),277,RPH150  //END OF LINE?
         ENTER     PF(CLOC)
PLON3    MACRO     XID1,BCTRM,PF(CLOC)-270
         ADVANCE   .15                  //MOVE INTO NEXT
         LOGIC C   PF(PLOC)
         LEAVE     PF(PLOC)
         GATE LC   PF(CLOC)             //STATION CLEAR
         LOGIC S   PF(CLOC)             //TIE UP
         TRANSFER  ,RPH120
```

```
*
RPH150   GATE LC    PF(CLOC)
         LOGIC S    PF(CLOC)
 PLON    MACRO      XID1,RBC
         ADVANCE    .14
         LOGIC C    PF(PLOC)
         LEAVE      PF(PLOC)              //EXIT LINE
         ADVANCE    .14
         GATE LC    LEAKQ
         LOGIC S    LEAKQ
         LOGIC C    PF(CLOC)
         TRANSFER   ,RPH005
*
RPH200   BLET       PF(PLOC)=PF(CVSEC)    //KEEP SEGMENT
         GATE SF    271,RPH210            //QUEUE FULL?
         BLET       PF(CVSEC)=1           //YES;RESET CONV. SEC
         TRANSFER   ,BBD000               //BACK TO BACKBONE
RPH210   ENTER      271                   //GET SEGMENT
         GATE LC    260                   //SEGMENT ZONE OPEN
         LOGIC S    260
 PLON    MACRO      XID1,BCTRMM
         ADVANCE    .1
         LEAVE      PF(PLOC)              //FREE BB
         ADVANCE    .19                   //SWITCH IN
         LOGIC C    260
         TRANSFER   ,RPH110
*----------------------------
* TEST RECIRCULATOR #1
*----------------------------
RCL1000  TEST G     &ATHEAD,0,RCL1005
         TEST L     PF(TSEQN),&ATHEAD,RCL1005   //LOWER TRUCK #
         LINK       BACKUP,FIFO
RCL1005  BLET       PF(PCODE)=RCRQ1
         ENTER      RECR1                       //GRAB RC
BARG     MACRO      PQ2,TOP,100.0*S(RECR1)/(S(RECR1)+R(RECR1))
         BLET       PF(PCODE)=CLTEST
PLON     MACRO      XID1,RC11
         ADVANCE    .12                         //CLEAR SWITCH
         LEAVE      PF(CVSEC)                   //EXIT BB
         LEAVE      BACKUP                      //EXIT BACKUP
         LOGIC C    RECR1                       //CLEAR ENTRY
         ADVANCE    7.53                        //UP THE VERTICAL
         TEST E     CH(RECR1),0,*+2
         BLET       &ATHEAD=PF(TSEQN)           //AT HEAD OF LINK
         LINK       RECR1,FIFO
RCL1010  LINK       RECR1O,FIFO,RCL1020         //YES;PUT BACK
RCL1020  ENTER      RECR1O
         LEAVE      RECR1
BARG     MACRO      PQ2,TOP,100.0*S(RECR1)/(S(RECR1)+R(RECR1))
PLON     MACRO      XID1,RC12
         BLET       &ATHEAD=0
         SCANUCH G  RECR1O,TSEQN$PF,0,TSEQN$PF,DELRT$PF,RCL1030
         TRANSFER   ,RCL1040
RCL1030  SCANUCH G  RECR1,TSEQN$PF,0,TSEQN$PF,DELRT$PF,RCL1050
RCL1040  BLET       &ATHEAD=PF(DELRT)
RCL1050  BLET       PF(DELRT)=0
         ADVANCE    1.3
         BLET       PF(CVSEC)=2
         GATE LC    PF(CVSEC)
         LOGIC S    PF(CVSEC)
         ENTER      PF(CVSEC)
         BLET       XF(COUNT)=XF(COUNT)+1
         TEST GE    XF(COUNT),&MAX,*+2
         LOGIC S    COUNT
PLON     MACRO      XID1,MBB2
         ADVANCE    .14
         LEAVE      RECR1O
         UNLINK     RECR1O,RCL1020,1
PLON     MACRO      XID1,BB2
```

```
         LOGIC C     PF(CVSEC)
         ADVANCE     ML(CSECT,PF$CVSEC,1)         //ON BACKBONE
         JOIN        GCLTEST
         TRANSFER    ,BBD010
*------------------------------
* TEST CELL LOGIC
*------------------------------
CEL006   GATE SNF    34,CEL012
         SCANUCH LE  ACELLS,DELRT$PF,6,,,CEL012 //ANY ACTIVE HERE?
         TEST G      &TCRTE(PF$ENGINE),0,CEL010 //YES;ANY CELL WORK?
         TEST LE     &TCRTE(PF$ENGINE),6,CEL012 //NO;RIGHT RANGE?
         BLET        PF(LOC1)=1                  //YES;CLEARANCE INDICATOR
CEL010   ENTER       34                   1ST BANK CELL
         LEAVE       PF$CVSEC
         REMOVE      GCLTEST
CEL011   BLET        PF(CVSEC)=34
PLON     MACRO       XID1,P34
         ADVANCE     .33
         LINK        21,FIFO,FIRST
*
CEL012   GATE SNF    35,BBD020
         SCANUCH G   ACELLS,DELRT$PF,6,,,BBD020 //ANY ACTIVE HERE?
         TEST G      &TCRTE(PF$ENGINE),0,CEL020  //YES;ANY CELL WORK?
         TEST G      &TCRTE(PF$ENGINE),6,BBD020 //NO;RIGHT RANGE?
         BLET        PF(LOC1)=7
CEL020   ENTER       35                   //CAPTURE EXIT STORAGE
         LEAVE       PF$CVSEC             //LEAVE ZONE
         REMOVE      GCLTEST
         BLET        PF(CVSEC)=35
PLON     MACRO       XID1,P35
         ADVANCE     .98                  //INDEX
         TRANSFER    ,CEL040
*
CEL013   GATE SNF    36,BBD020
         SCANUCH G   ACELLS,DELRT$PF,6,,,BBD020 //ANY ACTIVE HERE?
         TEST G      &TCRTE(PF$ENGINE),0,CEL030  //YES;ANY CELL WORK?
         TEST G      &TCRTE(PF$ENGINE),6,BBD020 //NO;RIGHT RANGE?
         BLET        PF(LOC1)=7
CEL030   ENTER       36                   //CAPTURE EXIT STORAGE
         LEAVE       PF$CVSEC             //LEAVE ZONE
         REMOVE      GCLTEST
         BLET        PF(CVSEC)=36
PLON     MACRO       XID1,P36
         ADVANCE     .62                  //INDEX
*
* TEST CELL LOGIC
*
CEL040   PRIORITY    10
         LINK        22,FIFO,FIRST
FIRST    GATE LC     PF$CVSEC
         LOGIC S     PF$CVSEC
         TEST G      &TCRTE(PF$ENGINE),0,TSC050  //GENERAL ASSIGNMENT?
         BLET        PF(DELRT)=&TCRTE(PF$ENGINE) //NO SPECIFIC
TSC000   GATE FS     PF(DELRT)+80         //GRAB CELL?
         GATE LS     PF(DELRT)+40         //ACTIVE?
         TRANSFER    SIM,TSC100,TSC000    //NO;PROCEED
TSC050   TEST E      PF(CVSEC),34,TSC080  //NOT;SPECIFIC 1ST DAY RANGE
         BLET        PF(LOC1)=1           //1ST CELL
         BLET        PF(LOC2)=6           //LAST CELL
         TRANSFER    ,TSC090              //YES;PROCEED
TSC080   BLET        PF(LOC1)=7           //1ST CELL
         BLET        PF(LOC2)=18          //LAST CELL
TSC090   SELECT E    DELRT$PF,PF$LOC1,PF$LOC2,1,BV
         TEST E      PF(DELRT),0,TSC100   //FOUND HOME?
         LOGIC S     PF(LOC1)+80          //NO
         GATE LC     PF(LOC1)+80          //AWAIT OPENING
         TRANSFER    ,TSC090
TSC100   SEIZE       PF(DELRT)+80         //ASSIGN CELL
         SEIZE       FN7                  //PATH WAY
```

```
          LOGIC C    PF$CVSEC
          UNLINK     FN2,FIRST,1           //TAKE NEXT
          LEAVE      PF$CVSEC              //CLEAR ZONE
          TEST NE    PF(CVSEC),34,FBAYS
PLON3     MACRO      XID1,FR,PF$CVSEC
          ADVANCE    .12
 FBAYS    BLET       XF(COUNT)=XF(COUNT)-1
          TEST L     XF(COUNT),&MAX,TSC102
          LOGIC C    COUNT
          UNLINK     BACKUP,BBD0106,1,,,TSC101
          TRANSFER   ,TSC102
 TSC101   UNLINK     RECR1,RCL1010,1
 TSC102   ADVANCE    0
PLON3     MACRO      XID1,IN,PF(DELRT)
          ADVANCE    FN4
          RELEASE    FN7
          BLET       PF(CLOC)=PF(DELRT)+40        //LOCATION - DO HOOK-UP
          BLET       PL(CYCLE)=(&HOOK(PF$ENGINE)+&RHOOK(PF$ENGINE))/&PERF(PF$MOD)
          TRANSFER   SBR,PRO000,SUBR$PF           //HOOK UP
          GATE LC    PF$DELRT+451                 //CELL DELAY UNDERWAY
*
* CELL DELAY TEST
*
          BLET       &GBCNT(2)=&GBCNT(2)+1        //REJECT COUNT
          BLET       PF(CTR)=&GBCNT(2)            //SAVE COUNT
          BLET       PF(PCT)=&CRPRRJ*100          //SAVE PCT
          TRANSFER   SBR,RPCT00,SUBR$PF           //DETERMINE REJECT
          TEST NE    PF(RJCT),0,TSC105            //NEED CELL REPAIR?
 SCOLOR   MACRO      XID1,'RED'
          ADVANCE    &CRPRTIM                     YES;DOWNTIME
SCOLOR    MACRO      XID1,&ECLR(PF$ENGINE)
 TSC105   SEIZE      PF$DELRT+100                 RUN TIME STATISTICS
          ADVANCE    &CTEST(PF$ENGINE)-&RHOOK(PF$ENGINE)/&PERF(PF$MOD) //TEST TIME
          BLET       ML(PROD,100,8)=ML(PROD,100,8)+1
          RELEASE    PF$DELRT+100                 RUN TIME STATISTICS
* MAJOR REPAIR TEST 1ST
*
          TEST L     PF$RETEST,2,TSC130           2ND TEST?
          BLET       PF(RETEST)=PF(RETEST)+1      //BUMP COUNT
          TEST E     PF(RETEST),1,TSC110
          BLET       &ENGC1(PF$ENGINE)=&ENGC1(PF$ENGINE)+1 //REJECT COUNT
          BLET       PF(CTR)=&ENGC1(PF$ENGINE)             //SAVE COUNT
          BLET       PF(PCT)=&TRJT1(PF$ENGINE)*100         //SAVE PCT
          TRANSFER   SBR,RPCT00,SUBR$PF           //DETERMINE REJECT
          TRANSFER   ,TSC120
* MAJOR REPAIR TEST 2ND
*
 TSC110   BLET       &ENGC2(PF$ENGINE)=&ENGC2(PF$ENGINE)+1 //REJECT COUNT
          BLET       PF(CTR)=&ENGC2(PF$ENGINE)    //SAVE COUNT
          BLET       PF(PCT)=&TRJT2(PF$ENGINE)*100   //SAVE PCT
          TRANSFER   SBR,RPCT00,SUBR$PF           //DETERMINE REJECT
 TSC120   TEST NE    PF(RJCT),1,TSC200            //NEED CELL REPAIR?
          BLET       PF(RETEST)=0                 //NO;ELIMINATE RETEST NEED
* MINOR REPAIR TEST
*
          BLET       &GBCNT(3)=&GBCNT(3)+1        //REJECT COUNT
          BLET       PF(CTR)=&GBCNT(3)            //SAVE COUNT
          BLET       PF(PCT)=&LRPRRJ*100          //SAVE PCT
          TRANSFER   SBR,RPCT00,SUBR$PF           //DETERMINE REJECT
          TEST E     PF(RJCT),0,TSC200            //NEED CELL REPAIR?
 TSC130   BLET       PF(RETEST)=0                 //CLEAR PARMS
 TSC200   MSAVEVALUE PROD+,PF$ENGINE,PF$RETEST+3,1,ML COLLECT TEST
          MSAVEVALUE PROD+,100,PF$RETEST+3,1,ML   //STATS
          TEST E     PF(RJCT),1,TSC210            //LAST PHASE
SCOLOR    MACRO      XID1,'RED'
 TSC210   BLET       PL(CYCLE)=&UNHK(PF$ENGINE)/&PERF(PF$MOD)
          TRANSFER   SBR,PRO000,SUBR$PF       //HOOK UP
          SEIZE      FN11                     //EXIT PATH CLEAR
 PLON3    MACRO      XID1,OUTB,PF(DELRT)
```

```
          ADVANCE    .14
          RELEASE    PF(DELRT)+80            //RELEASE CELL
          LOGIC C    PF(LOC1)+80             //SIGNAL OPENING
PLON3     MACRO      XID1,OUT,PF(DELRT)
          ADVANCE    FN5
          BLET       PF(CVSEC)=FN3           //32 OR 33
          BLET       PF(PCODE)=CSTRIM        //DEFAULT TO CUSTOM TRIM
          TEST E     &TCOAT(PF$ENGINE),0,*+2 //ANY PAINT STD?
          BLET       PF(PCODE)=FNTRIM        //NO;PC=FINAL TRIM
          TEST E     PF(RJCT),1,TSC215          //REJECT?
          BLET       PF(PCODE)=REPAIRS       //YES;ROUTE TO REPAIRS
          TRANSFER   ,TSC220
 TSC215   ENTER      TSTLCOUT
 BARG     MACRO      PQ3,TOP,100.0*S(TSTLCOUT)/(S(TSTLCOUT)+R(TSTLCOUT))
          TEST E     BV(ENG105),1,TSC220     //10.5/12.5'S?
          BLET       PF(PCODE)=PF(PCODE)+10  //YES;RESET PROCESS CODE
 TSC220   GATE LC    80                      //METER
          LOGIC S    80                      //CLOSE OFF
          GATE LC    PF(CVSEC)               //CLEARANCE
          LOGIC S    PF(CVSEC)               //ZONE CLEAR
          ENTER      PF(CVSEC)
 PLON3    MACRO      XID1,MTL,(PF$CVSEC-30)
          ADVANCE    ML(CSECT,PF$CVSEC,12)
          RELEASE    FN11                    //EXIT
          LOGIC C    80
          LOGIC C    PF(CVSEC)
          TEST L     PF(PCODE),10,*+2        //10.5-12.5 DON'T JOIN HERE
          JOIN       PF(PCODE)               //JOIN GROUP
 PLON3    MACRO      XID1,TL,PF$CVSEC-30
          TRANSFER   ,TLC060                 //ENTER CONV. LOOP
*---------------------------
* RETORQUE AREA
*---------------------------
 RTQ000   GATE LC    RTORKQ                   //RETORQUE QUEUE
          LOGIC S    RTORKQ
          ENTER      RTORKQ
          REMOVE     PF(PCODE)
          TEST NE    BV(RTQBYP),1,RTQ010          //EMPTY OR PAINT?
          LEAVE      TSTLCOUT
 BARG     MACRO      PQ3,TOP,100.0*S(TSTLCOUT)/(S(TSTLCOUT)+R(TSTLCOUT))
 RTQ010   ADVANCE    0
 PLON     MACRO      XID1,RTQ0
          ADVANCE    .12
          LEAVE      PF(PLOC)                 //FREE RETORQUE INPUT
          LOGIC C    RTORKQ
          UNLINK     PF(PLOC),TLC3205,1
          ADVANCE    .75
          BLET       PF(PLOC)=37
          LINK       RTORKQ,FIFO,RTQ050
 RTQ050   TEST NE    BV(RTQBYP),1,RTQ100      //BYPASS?
          SELECT LC  CVSEC$PF,38,39           //PICK PATH
          LOGIC S    PF(CVSEC)
          TEST NE    BV(RTQUL),1,*+2
          UNLINK     RTORKQ,RTQ050,1
PLON3     MACRO      XID1,RTQ,(PF$CVSEC-37)   //IN TRAVEL
          ADVANCE    .15                      //CONV. TRAVEL
          LEAVE      PF(PLOC)                 //STOP 20 REOPENED
          ADVANCE    ML(CSECT,PF$CVSEC,1)     //QUEUE POSITION
          ENTER      PF(CVSEC)
PLON3     MACRO      XID1,RTS,(PF$CVSEC-37)
          ADVANCE    .14                      //MOVE IN
          LOGIC C    PF(CVSEC)
          UNLINK     RTORKQ,RTQ050,1          //MAKE OPENING
          BLET       PF(CLOC)=PF(CVSEC)+23    //STATION ID 61/62
          BLET       PL(CYCLE)=&RTORK(PF$ENGINE)/&PERF(PF$MOD)/2.0
          TRANSFER   SBR,PRO000,SUBR$PF       //PROCESS
SCOLOR    MACRO      XID1,&ECLR(PF$ENGINE)
          BLET       PF(PLOC)=PF(CVSEC)       //SAVE LOCATION
          BLET       PF(CVSEC)=40             //BUMP LOCATION
```

```
          GATE LC    PF(CVSEC)
          LOGIC S    PF(CVSEC)
          ENTER      PF(CVSEC)
          ADVANCE    .12
          LOGIC C    PF(CVSEC)
          LEAVE      PF(PLOC)
          ADVANCE    ML(CSECT,PF$PLOC,11)-.12 //REMAINING TRAVEL
          BLET       PF(PLOC)=PF(CVSEC)
          BLET       PF(CVSEC)=15
          BLET       &EPROD(3)=&EPROD(3)+1    //COUNT ENGINE IN PROCESS
          BLET       &PRORATE(2)=&PRORATE(2)+1     //COUNT ENGINE RATE
 BARG     MACRO      RT2,TOP,&PRORATE(2)
          BLET       &DUM='570'
          TRANSFER   SBR,FNDMOD,SUBR$PF
          GATE LC    PF(CVSEC)
          LOGIC S    PF(CVSEC)
          ENTER      PF(CVSEC)
 PLON3    MACRO      XID1,MBB,PF(CVSEC)
          ADVANCE    ML(CSECT,PF$CVSEC,12)    //MERGE ZONE
          TRANSFER   ,BBD055                  //RETURN TO BB
*
 RTQ100   BLET       PF(CVSEC)=14             //BACKBONE BYPASS
          GATE LC    PF(CVSEC)
          LOGIC S    PF(CVSEC)
          ENTER      PF(CVSEC)
 PLON     MACRO      XID1,BB14
          ADVANCE    .12                      //CLEARANCE
          LOGIC C    PF(CVSEC)
          LEAVE      PF(PLOC)
          UNLINK     RTORKQ,RTQ050,1          //TAKE NEXT
          ADVANCE    ML(CSECT,PF$CVSEC,1)-.12
          TRANSFER   ,BBD020
*---------------------------
* 569 REPAIRS
*---------------------------
 RPR000   GATE LC    REPRQ               //REPAIR QUEUE
          LOGIC S    REPRQ
          ENTER      REPRQ
 PLON     MACRO      XID1,RPQ
          ADVANCE    .12
          LEAVE      PF(PLOC)
          LOGIC C    REPRQ
          ADVANCE    .66
 RPR005   SELECT NU  CVSEC$PF,119,122    //OPEN SPUR
          TEST E     PF(CVSEC),0,RPR010
          LINK       34,FIFO             //AWAIT
 RPR010   SEIZE      PF(CVSEC)           //GET SPUR
          ENTER      EREPR               //EXIT PATH
 PLON3    MACRO      XID1,IN,PF(CVSEC)
          ADVANCE    .1                  //CLEARANCE
          LEAVE      REPRQ               //FREE QUEUE
          ADVANCE    .2                  //REMAINING PATH
          LEAVE      EREPR               //CLEAR PATH
          BLET       PF(CLOC)=PF(CVSEC)-118+265 //LOCATION
          TEST E     PF(RETEST),0,RPR050 //MAJOR REPAIR?
          BLET       PL(CYCLE)=&LRPRTIM  //TIME
          BLET       PF(PCODE)=CSTRIM        //DEFAULT TO CUSTOM TRIM
          TEST E     &TCOAT(PF$ENGINE),0,*+2 //ANY PAINT STD?
          BLET       PF(PCODE)=FNTRIM        //NO;PC=FINAL TRIM
          TRANSFER   ,RPR100
 RPR050   BLET       PL(CYCLE)=&HRPRTIM  //HEAVY REPAIR
          BLET       PF(PCODE)=CLTEST
 RPR100   TRANSFER   SBR,PRO000,SUBR$PF  //PROCESS TIME
          BLET       PF(RJCT)=0          //NOT REJECT
SCOLOR    MACRO      XID1,&ECLR(PF$ENGINE)
          ENTER      EREPR                //PATH CLEAR?
PLON3     MACRO      XID1,OUTB,PF(CVSEC)  //YES
          ADVANCE    .09                  //BACKOUT
          RELEASE    PF(CVSEC)            //FREE SPUR
```

```
           UNLINK      34,RPR005,1              //TAKE NEXT
PLON3      MACRO       XID1,OUT,PF(CVSEC)       //OUTBOUND PATH
           ADVANCE     .3
           TEST NE     PF(PCODE),CLTEST,RPR110
 XXX002    ENTER       TSTLCOUT
 BARG      MACRO       PQ3,TOP,100.0*S(TSTLCOUT)/(S(TSTLCOUT)+R(TSTLCOUT))
 RPR110    BLET        PF(PLOC)=259             //EXIT PATH
           BLET        PF(CVSEC)=4              //BACKBONE ENTRY
           GATE LC     PF(CVSEC)
           LOGIC S     PF(CVSEC)
           ENTER       PF(CVSEC)
           TEST E      PF(PCODE),CLTEST,RPR150
           BLET        XF(COUNT)=XF(COUNT)+1
           TEST GE     XF(COUNT),&MAX,*+2       //COUNT OUT
           LOGIC S     COUNT
 RPR150    JOIN        PF(PCODE)                     //JOIN NEXT GROUP
 PLON3     MACRO       XID1,MBB,PF(CVSEC)
           ADVANCE     ML(CSECT,PF$CVSEC,12)         //MERGE ZONE
           TRANSFER    ,BBD055                       //RETURN TO BB
*---------------------------
* TRIM CONVEYOR
*---------------------------
 TRM000    GATE LC     STRMI                         //INPUT ZONE
           LOGIC S     STRMI                         //SHUT OFF ZONE
           ENTER       STRMI                         //ENTER ZONE
 PLON      MACRO       XID1,TRMI
           ADVANCE     .12                           //INDEX IN
           LOGIC C     STRMI                         //CLEAR
           LEAVE       SSTG0                         //FREE STAGE
           UNLINK      SSTG0,BBD1730,1
           ADVANCE     1.87
           BLET        PF(CVSEC)=PF(PCODE)*20        //TRIM ZONE
           LINK        STRMI,FIFO,TRM020
 TRM020    ENTER       PF(CVSEC)
           ENTER       STRM1
 PLON      MACRO       XID1,TRM1
           ADVANCE     .23
           LEAVE       STRMI
           LEAVE       STRM1
           UNLINK      STRMI,TRM020,1
 TRM030    TEST E      PF(PCODE),CSTRIM,FNT000       //CUSTOM OR FINAL
*---------------------------
* CUSTOM TRIM
*---------------------------
 PLON      MACRO       XID1,CTC
           ADVANCE     2.15
           BLET        PF(PLOC)=PF(CVSEC)
           BLET        PF(INDX)=1
           BLET        PF(CVSEC)=PF(CVSEC)+1
           LINK        PF(CVSEC),FIFO,CST000
 CST000    ENTER       PF(CVSEC)
 PLON3     MACRO       XID1,CT,PF(INDX)
           ADVANCE     .12
           LEAVE       PF(PLOC)
           LEAVE       CSTRMCNT
 BARG      MACRO       PQ4,TOP,100.0*S(CSTRMCNT)/(S(CSTRMCNT)+R(CSTRMCNT))
           ADVANCE     .15
 CST010    TEST LE     PF(INDX),&CSTLS,CST020
           BLET        PF(CLOC)=CTRIM1-1+PF(INDX)  //POSITION
           BLET        PL(CYCLE)=ML(CUSTRM,&ECLASI(PF$ENGINE),PF$INDX)/&PERF(PF$MOD)
           TEST G      PL(CYCLE),0,CST020
           TRANSFER    SBR,PRO000,SUBR$PF
 CST020    BLET        PF(PLOC)=PF(CVSEC)
           BLET        PF(INDX)=PF(INDX)+1
           BLET        PF(CVSEC)=PF(CVSEC)+1
           ENTER       PF(CVSEC)
           LEAVE       PF(PLOC)
 PLON3     MACRO       XID1,CT,PF(INDX)
           TEST E      PF$INDX,2,CST030
```

```
         UNLINK      PF$PLOC,CST000,1
CST030   TEST NE     PF(CVSEC),149,CST110
         ADVANCE     .27
         TRANSFER    ,CST010
*
CST110   ADVANCE     .93
         BLET        PF(PCODE)=PNTSYS
         TEST G      &BTRIM(PF$ENGINE),0,*+2
         BLET        PF(PCODE)=FNTRIM
         LINK        PF(CVSEC),FIFO,CST120
CST120   ENTER       (294+PF$PCODE)
BARG4    MACRO       PQ,(PF$PCODE-3),TOP,100.0*S(294+PF$PCODE)/(S(294+PF$PCODE)+_
                     R(294+PF$PCODE))
         BLET        &PRORATE(3)=&PRORATE(3)+1      //COUNT ENGINE IN PROCESS
BARG     MACRO       RT3,TOP,&PRORATE(3)
         GATE LC     TRMOUT
         LOGIC S     TRMOUT
         ENTER       TRMOUT
PLON     MACRO       XID1,TRMOM
         ADVANCE     .12
         LEAVE       PF(CVSEC)
         BLET        &EPROD(4)=&EPROD(4)-1     //COUNT ENGINE IN PROCESS
         UNLINK      PF(CVSEC),CST120,1
TRM100   LOGIC C     TRMOUT
PLON     MACRO       XID1,TRMO
         ADVANCE     1.39
         BLET        PF(PLOC)=158           //EXIT PATH
         BLET        PF(CVSEC)=17           //BACKBONE ENTRY
         GATE LC     PF(CVSEC)
         LOGIC S     PF(CVSEC)
         ENTER       PF(CVSEC)
         ENTER       BACKBCNT                   //BACKBONE ZONE
         TEST GE     S(BACKBCNT),&BBLIM,*+2     //AT CHOKE LIMIT
         LOGIC S     BACKBCNT                   //CLOSE OFF ZONE
PLON3    MACRO       XID1,MBB,PF(CVSEC)
         ADVANCE     ML(CSECT,PF$CVSEC,12) //MERGE ZONE
         TRANSFER    ,BBD055
*---------------------------
* FINAL TRIM
*---------------------------
FNT000   BLET        &DUM='571'
         TRANSFER    SBR,FNDMOD,SUBR$PF
PLON     MACRO       XID1,FT0
         ADVANCE     1.58
         BLET        PF(PLOC)=PF(CVSEC)
         BLET        PF(INDX)=1
         BLET        PF(CVSEC)=PF(CVSEC)+1
         LINK        PF(CVSEC),FIFO,FNT010
FNT010   ENTER       PF(CVSEC)
PLON3    MACRO       XID1,FT,PF(INDX)
         ADVANCE     .12
         LEAVE       PF(PLOC)
         LEAVE       FNTRMCNT
BARG     MACRO       PQ5,TOP,100.0*S(FNTRMCNT)/(S(FNTRMCNT)+R(FNTRMCNT))
         ADVANCE     .15
FNT020   TEST NE     PF(DELRT),99,FNT030
         TEST LE     PF(INDX),&FNTLS,FNT030
         BLET        PF(CLOC)=FTRIM1-1+PF(INDX)  //POSITION
         BLET        PL(CYCLE)=ML(FNLTRM,&ECLASI(PF$ENGINE),PF$INDX)/&PERF(PF$MOD)
         TEST G      PL(CYCLE).0,FNT030
         TRANSFER    SBR,PRO000,SUBR$PF
FNT030   BLET        PF(PLOC)=PF(CVSEC)
         BLET        PF(INDX)=PF(INDX)+1
         BLET        PF(CVSEC)=PF(CVSEC)+1
         ENTER       PF(CVSEC)
         LEAVE       PF(PLOC)
PLON3    MACRO       XID1,FT,PF(INDX)
         TEST E      PF$INDX,2,FNT040
         UNLINK      PF$PLOC,FNT010,1
```

```
 FNT040   TEST NE    PF(CVSEC),169,FNT110
          ADVANCE    .27
          TRANSFER   ,FNT020
*
 FNT110   ADVANCE    .13
          BLET       PF(CLOC)=FTRIM1-1+PF(INDX) //POSITION
          BLET       PL(CYCLE)=ML(FNLTRM,&ECLASI(PF$ENGINE),PF$INDX)/&PERF(PF$MOD)
          TEST G     PL(CYCLE),0,FNT120
          TRANSFER   SBR,PRO000,SUBR$PF
 FNT120   SPLIT      1,FIN000                          //SEND TO INSPECTION
 SCOLOR   MACRO      XID1,'WHITE'
 WRITEO   MACRO      LBRID,XID1,'EMPTY'
          BLET       &PRORATE(4)=&PRORATE(4)+1      //COUNT ENGINE IN PROCESS
 BARG     MACRO      RT4,TOP,&PRORATE(4)
          BLET       PF(ENGINE)=0
          BLET       PF(PCODE)=0
          BLET       PF(DELRT)=0
          ENTER      TRMOUT
          GATE LC    TRMOUT
          LOGIC S    TRMOUT
 PLON     MACRO      XID1,MTRMO
          ADVANCE    .23
          LEAVE      PF(CVSEC)
          TRANSFER   ,TRM100
************************************************************
* PAINT SYSTEM
************************************************************
 PST000   GATE LC    SPNT1                      //ENTER PAINT
          LOGIC S    SPNT1
          ENTER      SPNT1
          BLET       &DUM='570P'
          TRANSFER   SBR,FNDMOD,SUBR$PF
PLON      MACRO      XID1,PNT1
          ADVANCE    .12                        //CLEAR LIMIT
          LEAVE      SSTG0
          LOGIC C    SPNT1
          UNLINK     SSTG0,BBD1730,1
          ADVANCE    2.35
*
* Washer process chain.
*
          LINK       SPNT1,FIFO,PST010
 PST010   GATE LC    SPNT2                     //TRANSITION TO PROCESS CHAIN
          LOGIC S    SPNT2
PLON      MACRO      XID1,PXFR
          ADVANCE    .13
          ENTER      SPNT2
          LEAVE      SPNT1                     //FREE INPUT QUEUE
          LEAVE      PAINTCNT
 BARG     MACRO      PQ6,TOP,100.0*S(PAINTCNT)/(S(PAINTCNT)+R(PAINTCNT))
PLON      MACRO      XID1,PNT2
          ADVANCE    8.02/&PNTSSP              //TIME TO INDEX INTO PAINT
          LOGIC C    SPNT2
          UNLINK     SPNT1,PST010,1
          ADVANCE    49.70/&PNTSSP             //REMAINING TRAVEL
          GATE LC    SPNT3                     //TRANSITION TO PROCESS CHAIN
          LOGIC S    SPNT3
          ENTER      SPNT3
PLON      MACRO      XID1,PNT3
          ADVANCE    .12
          LEAVE      SPNT2
          LOGIC C    SPNT3
          ADVANCE    .25                       //TIME TO INDEX INTO PAINT
          GATE LC    SPNT4                     //TRANSITION TO PROCESS CHAIN
          LOGIC S    SPNT4
          BLET       PF(CLOC)=PMASK
          BLET       PL(CYCLE)=(&MASK(PF$ENGINE)+&BLOWO(PF$ENGINE))/&PERF(PF$MOD)
          TRANSFER   SBR,PRO000,SUBR$PF
          SPLIT      1,PST100                  //THROUGH PREP
```

```
          LOGIC C     SPNT4
          LINK        SPNT3,FIFO
 PST020   ENTER       SPNT4
          LEAVE       SPNT3
PLON      MACRO       XID1,PNT4
          ADVANCE     .20                       //TIME TO INDEX INTO PAINT
          BLET        PF(CLOC)=PPRIM
          GATE SF     SPNT5,PST030              //SECOND STOP OPEN?
          BLET        PF(CLOC)=0                //NO TAG AS SECOND LOAD BAR
          LINK        SPNT4,FIFO
 PST030   ENTER       SPNT5                     //Enter initial section of oven.
          LEAVE       SPNT4
PLON      MACRO       XID1,PNT5
          ADVANCE     .14                       //Delay for path time.
          TEST NE     PF(CLOC),0,PST040
          BLET        PF(CLOC)=PNTTC
          BLET        PL(CYCLE)=5.0
          TRANSFER    SBR,PRO000,SUBR$PF
          GATE LC     464                      //PAINT DELAY UNDERWAY
          UNLINK      SPNT4,PST030,1
 PST040   MARK        DELAY$PL                 //START FLASHOFF TIME
          ENTER       SPNT6
          LEAVE       SPNT5
          TEST E      PF(CLOC),0,*+2
          SPLIT       1,PST100
PLON      MACRO       XID1,PNT6
          ADVANCE     1.5                      //TRAVEL TIME
          LINK        SPNT6,FIFO,PST042
 PST042   GATE LC     SPNT6                    //CAPTURE EXIT STOP
          LOGIC S     SPNT6
          BLET        PL(CYCLE)=&FLASH-MP$DELAY$PL //AWAIT FLASH-OFF
          TEST G      PL(CYCLE),0,*+2
          ADVANCE     PL(CYCLE)                //AWAIT FLASH-OFF
*
* Oven.
*
          ENTER       SPNT7
          LEAVE       SPNT6
          LOGIC C     SPNT6
          UNLINK      SPNT6,PST042,1
          MARK        DELAY$PL                 //START OVEN TIME
PLON      MACRO       XID1,PNT7
          ADVANCE     .69
          LINK        SPNT7,FIFO,PST044
 PST044   GATE LC     SPNT7                    //CAPTURE EXIT STOP
          LOGIC S     SPNT7
          BLET        PL(CYCLE)=&TIMEOVEN-MP$DELAY$PL //AWAIT BAKE
          TEST G      PL(CYCLE),0,*+2
          ADVANCE     PL(CYCLE)                //AWAIT OVEN TIME
          ENTER       SPNT8
          LEAVE       SPNT7
          LOGIC C     SPNT7
          UNLINK      SPNT7,PST044,1
          MARK        DELAY$PL                 //START COOLDOWN
PLON      MACRO       XID1,PNT8                //COOL DOWN ZONE
          ADVANCE     2.45
          BLET        &EPROD(7)=&EPROD(7)+1    //COUNT ENGINE IN PROCESS
          BLET        &PRORATE(5)=&PRORATE(5)+1      //COUNT ENGINE IN PROCESS
 BARG     MACRO       RT5,TOP,&PRORATE(5)
          GATE LC     SPNT8                    //CAPTURE EXIT STOP
          LOGIC S     SPNT8
          BLET        PL(CYCLE)=&COOL-MP$DELAY$PL //AWAIT COOLDOWN
          TEST G      PL(CYCLE),0,*+2
          ADVANCE     PL(CYCLE)                //AWAIT COOLING
          LOGIC C     SPNT8
          BLET        PF(PCODE)=FNTRIM         //PC=FINAL TRIM
          TEST G      &BTRIM(PF$ENGINE),0,*+2
          BLET        PF(DELRT)=99             //TAG AS BLUEBIRD
          BLET        PF(CVSEC)=PF(PCODE)*20   //BACKBONE ENTRY
```

```
         LINK      SPNT8,FIFO,PST050
PST050   ENTER     (294+PF$PCODE)
BARG4    MACRO     PQ,(PF$PCODE-3),TOP,100.0*S(294+PF$PCODE)/(S(294+PF$PCODE)+_
                   R(294+PF$PCODE))
         ENTER     PF(CVSEC)
         ENTER     STRM1
PLON     MACRO     XID1,MTRM1
         ADVANCE   .55                     //MERGE ZONE
         LEAVE     SPNT8
         LEAVE     STRM1
         UNLINK    SPNT8,PST050,1
         TRANSFER  ,TRM030                 //RETURN TO BB
*
PST100   TEST E    BV(PBATCH),1,PST120
         BLET      PF(LCTR)=2
PST110   UNLINK    SPNT3,PST020,1
         ADVANCE   .1
         LOOP      LCTR$PF,PST110
PST120   TERMINATE
*
* FINISHED ENGINE DATA
*
FIN000   TEST NE   PF(DELRT),99,BLU000     //BLUEBIRD ENGINE?
         BLET      &OFLDENGS=&OFLDENGS+1
         BLET      ML(PROD,PF$ENGINE,2)=ML(PROD,PF$ENGINE,2)+1
         BLET      ML(PROD,100,2)=ML(PROD,100,2)+1
         BLET      &FTRMENGS=&FTRMENGS+1
         BLET      ML(PROD,PF$ENGINE,2)=ML(PROD,PF$ENGINE,2)+1
         BLET      ML(PROD,100,2)=ML(PROD,100,2)+1
         BLET      &EPROD(5)=&EPROD(5)+1           //COUNT ENGINE IN PROCESS
         BLET      &INV572(PF$ENGINE)=&INV572(PF$ENGINE)+1 //COUNT IN
         BLET      &INV572(100)=&INV572(100)+1     //COUNT IN TOTAL
         BLET      &INPROC=&INPROC-1               //COUNT OUT
         LEAVE     EWIPQ
WRITE    MACRO     INP,&INPROC
BARG     MACRO     IPB,RIGHT,&INPROC
WRITE    MACRO     IV572,&INV572(100)
BARG     MACRO     IVB,RIGHT,&INV572(100)
         TEST G    PL(LAPTIM),0,FIN010             //NON-INITIAL LOAD?
* COLLECT TIME BY ENGINE
         BLET      ML(PROTIME,PF$ENGINE,1)=ML(PROTIME,PF$ENGINE,1)+MP$LAPTIM$PL
         BLET      ML(PROTIME,PF$ENGINE,2)=ML(PROTIME,PF$ENGINE,2)+1
         BLET
ML(PROTIME,PF$ENGINE,3)=ML(PROTIME,PF$ENGINE,1)/ML(PROTIME,PF$ENGINE,2)
         TEST E    ML(PROTIME,PF$ENGINE,2),1,*+3
         BLET      ML(PROTIME,PF$ENGINE,4)=MP$LAPTIM$PL        //MAX
         BLET      ML(PROTIME,PF$ENGINE,5)=MP$LAPTIM$PL        //MIN
         TEST G    MP$LAPTIM$PL,ML(PROTIME,PF$ENGINE,4),*+2
         BLET      ML(PROTIME,PF$ENGINE,4)=MP$LAPTIM$PL        //MAX
         TEST L    MP$LAPTIM$PL,ML(PROTIME,PF$ENGINE,5),*+2
         BLET      ML(PROTIME,PF$ENGINE,5)=MP$LAPTIM$PL        //MIN
* COLLECT TIME IN TOTAL
         BLET      ML(PROTIME,100,1)=ML(PROTIME,100,1)+MP$LAPTIM$PL
         BLET      ML(PROTIME,100,2)=ML(PROTIME,100,2)+1
         BLET      ML(PROTIME,100,3)=ML(PROTIME,100,1)/ML(PROTIME,100,2)
         TEST E    ML(PROTIME,100,2),1,*+3
         BLET      ML(PROTIME,100,4)=MP$LAPTIM$PL        //MAX
         BLET      ML(PROTIME,100,5)=MP$LAPTIM$PL        //MIN
         TEST G    MP$LAPTIM$PL,ML(PROTIME,100,4),*+2
         BLET      ML(PROTIME,100,4)=MP$LAPTIM$PL        //MAX
         TEST L    MP$LAPTIM$PL,ML(PROTIME,100,5),*+2
         BLET      ML(PROTIME,100,5)=MP$LAPTIM$PL        //MIN
WRITE    MACRO     SYST,(ML(PROTIME,100,3)/60.0)
* PROCESS COUNT
FIN010   BLET      ML(ESYSPRF,1,2)=ML(ESYSPRF,1,2)+1
         TEST E    ML(ESYSPRF,1,2),1,*+2
         BLET      ML(ESYSPRF,1,5)=&INPROC         //MIN
         TEST L    &INPROC,ML(ESYSPRF,1,5),*+2
         BLET      ML(ESYSPRF,1,5)=&INPROC         //MIN
```

```
* FINISH SEQUENCE
          BLET       &FINORD=&FINORD+1                //FINISH ORDER
          BLET       ML(SEQVAR,PF$ENGINE,1)=ML(SEQVAR,PF$ENGINE,1)+V$FINSEQ
          BLET       ML(SEQVAR,PF$ENGINE,2)=ML(SEQVAR,PF$ENGINE,2)+1
          BLET       ML(SEQVAR,PF$ENGINE,3)=ML(SEQVAR,PF$ENGINE,1)/ML(SEQVAR,PF$ENGINE,2)
          TEST E     ML(SEQVAR,PF$ENGINE,2),1,*+3
          BLET       ML(SEQVAR,PF$ENGINE,4)=V$FINSEQ          //MAX
          BLET       ML(SEQVAR,PF$ENGINE,5)=V$FINSEQ          //MIN
          TEST G     V$FINSEQ,ML(SEQVAR,PF$ENGINE,4),*+2
          BLET       ML(SEQVAR,PF$ENGINE,4)=V$FINSEQ          //MAX
          TEST L     V$FINSEQ,ML(SEQVAR,PF$ENGINE,5),*+2
          BLET       ML(SEQVAR,PF$ENGINE,5)=V$FINSEQ          //MIN
* FINISH SEQUENCE IN TOTAL
          BLET       ML(SEQVAR,100,1)=ML(SEQVAR,100,1)+V$FINSEQ
          BLET       ML(SEQVAR,100,2)=ML(SEQVAR,100,2)+1
          BLET       ML(SEQVAR,100,3)=ML(SEQVAR,100,1)/ML(SEQVAR,100,2)
          TEST E     ML(SEQVAR,100,2),1,*+3
          BLET       ML(SEQVAR,100,4)=V$FINSEQ          //MAX
          BLET       ML(SEQVAR,100,5)=V$FINSEQ          //MIN
          TEST G     V$FINSEQ,ML(SEQVAR,100,4),*+2
          BLET       ML(SEQVAR,100,4)=V$FINSEQ          //MAX
          TEST L     V$FINSEQ,ML(SEQVAR,100,5),*+2
          BLET       ML(SEQVAR,100,5)=V$FINSEQ          //MIN
 FIN050   MARK       LAPTIM$PL                          //TIME IN TRUCK GRID
          BLET       PF(LOC2)=0                         //ZERO OUT
          SELECT E   LOC1$PF,51,300,PF$TSEQN,XH,FIN055
          JOIN       PF$LOC1
          TEST E     G(PF$LOC1),&TRKLD(PF$TSEQN),FIN060
          SPLIT      1,SHP000
          TRANSFER   ,FIN060
 FIN055   SELECT E   LOC1$PF,51,300,0,XH
          BLET       XH(PF$LOC1)=PF$TSEQN
          JOIN       PF$LOC1
          BLET       PF$LOC2=1
          QUEUE      TGRIDS
 FIN060   LINK       FINV,FIFO                        //GO INTO INVENTORY
 FIN080   LEAVE      TOTALQ
          REMOVE     PF$LOC1
          BLET       &INV572(PF$ENGINE)=&INV572(PF$ENGINE)-1
          BLET       &INV572(100)=&INV572(100)-1
* WAREHOUSE COUNT-MINIMUM
          BLET       ML(ESYSPRF,2,2)=ML(ESYSPRF,2,2)+1
          TEST E     ML(ESYSPRF,2,2),1,*+2
          BLET       ML(ESYSPRF,2,5)=CH(FINV)                   //MIN
          TEST L     CH(FINV),ML(ESYSPRF,2,5),*+2
          BLET       ML(ESYSPRF,2,5)=CH(FINV)                   //MIN
* TOTAL COUNT-MINIMUM
          BLET       ML(ESYSPRF,3,2)=ML(ESYSPRF,3,2)+1
          TEST E     ML(ESYSPRF,3,2),1,*+2
          BLET       ML(ESYSPRF,3,5)=S(TOTALQ)          //MIN
          TEST L     S(TOTALQ),ML(ESYSPRF,3,5),*+2
          BLET       ML(ESYSPRF,3,5)=S(TOTALQ)          //MIN
* COLLECT WAREHOUSE TIME BY ENGINE
 FIN090   TEST G     PL(LAPTIM),0,FIN095
          BLET       ML(WHSETIM,PF$ENGINE,1)=ML(WHSETIM,PF$ENGINE,1)+MP$LAPTIM$PL
          BLET       ML(WHSETIM,PF$ENGINE,2)=ML(WHSETIM,PF$ENGINE,2)+1
          BLET
ML(WHSETIM,PF$ENGINE,3)=ML(WHSETIM,PF$ENGINE,1)/ML(WHSETIM,PF$ENGINE,2)
          TEST E     ML(WHSETIM,PF$ENGINE,2),1,*+3
          BLET       ML(WHSETIM,PF$ENGINE,4)=MP$LAPTIM$PL         //MAX
          BLET       ML(WHSETIM,PF$ENGINE,5)=MP$LAPTIM$PL         //MIN
          TEST G     MP$LAPTIM$PL,ML(WHSETIM,PF$ENGINE,4),*+2
          BLET       ML(WHSETIM,PF$ENGINE,4)=MP$LAPTIM$PL         //MAX
          TEST L     MP$LAPTIM$PL,ML(WHSETIM,PF$ENGINE,5),*+2
          BLET       ML(WHSETIM,PF$ENGINE,5)=MP$LAPTIM$PL         //MIN
* COLLECT WAREHOUSE TIME IN TOTAL
          BLET       ML(WHSETIM,100,1)=ML(WHSETIM,100,1)+MP$LAPTIM$PL
          BLET       ML(WHSETIM,100,2)=ML(WHSETIM,100,2)+1
          BLET       ML(WHSETIM,100,3)=ML(WHSETIM,100,1)/ML(WHSETIM,100,2)
```

```
          TEST E     ML(WHSETIM,100,2),1,*+3
          BLET       ML(WHSETIM,100,4)=MP$LAPTIM$PL         //MAX
          BLET       ML(WHSETIM,100,5)=MP$LAPTIM$PL         //MIN
          TEST G     MP$LAPTIM$PL,ML(WHSETIM,100,4),*+2
          BLET       ML(WHSETIM,100,4)=MP$LAPTIM$PL         //MAX
          TEST L     MP$LAPTIM$PL,ML(WHSETIM,100,5),*+2
          BLET       ML(WHSETIM,100,5)=MP$LAPTIM$PL         //MIN
* DAILY SHIPMENTS
 FIN095   BLET       MH(DSHIPS,PF$ENGINE,&SDAY)=MH(DSHIPS,PF$ENGINE,&SDAY)+1
          BLET       MH(DSHIPS,PF$ENGINE,21)=MH(DSHIPS,PF$ENGINE,21)+1
          BLET       MH(DSHIPS,100,&SDAY)=MH(DSHIPS,100,&SDAY)+1
          BLET       MH(DSHIPS,100,21)=MH(DSHIPS,100,21)+1
 WRITE    MACRO      IV572,&INV572(100)
 BARG     MACRO      IVB,RIGHT,&INV572(100)
 WRITE    MACRO      SPE,MH(DSHIPS,100,21)
 BARG     MACRO      SPB,RIGHT,MH(DSHIPS,100,21)
          BLET       &EPROD(6)=&EPROD(6)+1             //COUNT ENGINE IN PROCESS
          TEST E     PF(LOC2),1,FIN100
          DEPART     TGRIDS
          BLET       PF(JNDX)=MX(SHIPS,PF$SEQNM,5)     //CUSTOMER#
          BLET       XH(PF$LOC1)=0
          BLET       MH(TSHIPS,PF$JNDX,&SDAY)=MH(TSHIPS,PF$JNDX,&SDAY)+1
          BLET       MH(TSHIPS,PF$JNDX,21)=MH(TSHIPS,PF$JNDX,21)+1
          BLET       MH(TSHIPS,100,&SDAY)=MH(TSHIPS,100,&SDAY)+1
          BLET       MH(TSHIPS,100,21)=MH(TSHIPS,100,21)+1
* COLLECT GRID TIME BY CUSTOMER
          BLET       ML(GRIDTIM,PF$JNDX,1)=ML(GRIDTIM,PF$JNDX,1)+MP$LAPTIM$PL
          BLET       ML(GRIDTIM,PF$JNDX,2)=ML(GRIDTIM,PF$JNDX,2)+1
          BLET       ML(GRIDTIM,PF$JNDX,3)=ML(GRIDTIM,PF$JNDX,1)/ML(GRIDTIM,PF$JNDX,2)
          TEST E     ML(GRIDTIM,PF$JNDX,2),1,*+3
          BLET       ML(GRIDTIM,PF$JNDX,4)=MP$LAPTIM$PL         //MAX
          BLET       ML(GRIDTIM,PF$JNDX,5)=MP$LAPTIM$PL         //MIN
          TEST G     MP$LAPTIM$PL,ML(GRIDTIM,PF$JNDX,4),*+2
          BLET       ML(GRIDTIM,PF$JNDX,4)=MP$LAPTIM$PL         //MAX
          TEST L     MP$LAPTIM$PL,ML(GRIDTIM,PF$JNDX,5),*+2
          BLET       ML(GRIDTIM,PF$JNDX,5)=MP$LAPTIM$PL         //MIN
* COLLECT GRID TIME IN TOTAL
          BLET       ML(GRIDTIM,100,1)=ML(GRIDTIM,100,1)+MP$LAPTIM$PL
          BLET       ML(GRIDTIM,100,2)=ML(GRIDTIM,100,2)+1
          BLET       ML(GRIDTIM,100,3)=ML(GRIDTIM,100,1)/ML(GRIDTIM,100,2)
          TEST E     ML(GRIDTIM,100,2),1,*+3
          BLET       ML(GRIDTIM,100,4)=MP$LAPTIM$PL         //MAX
          BLET       ML(GRIDTIM,100,5)=MP$LAPTIM$PL         //MIN
          TEST G     MP$LAPTIM$PL,ML(GRIDTIM,100,4),*+2
          BLET       ML(GRIDTIM,100,4)=MP$LAPTIM$PL         //MAX
          TEST L     MP$LAPTIM$PL,ML(GRIDTIM,100,5),*+2
          BLET       ML(GRIDTIM,100,5)=MP$LAPTIM$PL         //MIN
 FIN100   TERMINATE
*----------------------------------------------------
* SHIPPING SCHEDULE
*----------------------------------------------------
 SHP000   BLET       PF(DELRT)=0                      //STARTING TRUCK
          GATE LC    DINIT,SHP010                     //1ST ONE HERE?
          SPLIT      1,SIMC90                         //YES;CREATE SIMULATION CONTROL
          GATE LS    DINIT
 SHP010   BLET       PF(INDX)=0                       //RESET INDEX
          MARK       LAPTIM$PL
          BLET       PF(PTR)=0                        //ZERO OUT
          BLET       &DUM='572'
          TRANSFER   SBR,FNDMOD,SUBR$PF
          BLET       PF(CLOC)=SHIPR                   //SHIPPER LOCATES ENGINES
          BLET       PL(CYCLE)=&SHPTIM(1)
          TRANSFER   SBR,PRO000,SUBR$PF               //PROCESS TIME
          BLET       PF(CLOC)=ANALYST                 //ANALYSIS PRINTS TAGS
          BLET       PL(CYCLE)=&SHPTIM(2)
          TRANSFER   SBR,PRO000,SUBR$PF               //PROCESS TIME
          BLET       PF(CLOC)=SHIPR                   //SHIPPER TAGS ENGINES
          BLET       PL(CYCLE)=&SHPTIM(3)
          TRANSFER   SBR,PRO000,SUBR$PF               //PROCESS TIME
```

```
          ENTER      DOCKS                             //YES;OPEN DOCK DOOR?
          BLET       PF(CLOC)=TRCK572                  //TRUCKERS LOAD TRUCK
          BLET       PL(CYCLE)=&SHPTIM(4)
          TRANSFER   SBR,PRO000,SUBR$PF                //PROCESS TIME
          BLET       PF(CLOC)=CLERK                  //TRUCKERS LOAD TRUCK
          BLET       PL(CYCLE)=&SHPTIM(5)
          TRANSFER   SBR,PRO000,SUBR$PF                //PROCESS TIME
          LEAVE      DOCKS                             //TRUCK LEAVES DOCK
          UNLINK     FINV,FIN080,ALL,TSEQN$PF,PF$TSEQN
          BLET       XH(PF$LOC1)=0
* COLLECT TRUCK LOAD TIME BY CUSTOMER
          BLET       PF(JNDX)=MX(SHIPS,PF$SEQNM,5)      //CUSTOMER#
          BLET       ML(TRKLDTIM,PF$JNDX,1)=ML(TRKLDTIM,PF$JNDX,1)+MP$LAPTIM$PL
          BLET       ML(TRKLDTIM,PF$JNDX,2)=ML(TRKLDTIM,PF$JNDX,2)+1
          BLET       ML(TRKLDTIM,PF$JNDX,3)=ML(TRKLDTIM,PF$JNDX,1)/ML(TRKLDTIM,PF$JNDX,2)
          TEST E     ML(TRKLDTIM,PF$JNDX,2),1,*+3
          BLET       ML(TRKLDTIM,PF$JNDX,4)=MP$LAPTIM$PL           //MAX
          BLET       ML(TRKLDTIM,PF$JNDX,5)=MP$LAPTIM$PL           //MIN
          TEST G     MP$LAPTIM$PL,ML(TRKLDTIM,PF$JNDX,4),*+2
          BLET       ML(TRKLDTIM,PF$JNDX,4)=MP$LAPTIM$PL           //MAX
          TEST L     MP$LAPTIM$PL,ML(TRKLDTIM,PF$JNDX,5),*+2
          BLET       ML(TRKLDTIM,PF$JNDX,5)=MP$LAPTIM$PL           //MIN
* COLLECT TRUCK LOAD TIME IN TOTAL
          BLET       ML(TRKLDTIM,100,1)=ML(TRKLDTIM,100,1)+MP$LAPTIM$PL
          BLET       ML(TRKLDTIM,100,2)=ML(TRKLDTIM,100,2)+1
          BLET       ML(TRKLDTIM,100,3)=ML(TRKLDTIM,100,1)/ML(TRKLDTIM,100,2)
          TEST E     ML(TRKLDTIM,100,2),1,*+3
          BLET       ML(TRKLDTIM,100,4)=MP$LAPTIM$PL           //MAX
          BLET       ML(TRKLDTIM,100,5)=MP$LAPTIM$PL           //MIN
          TEST G     MP$LAPTIM$PL,ML(TRKLDTIM,100,4),*+2
          BLET       ML(TRKLDTIM,100,4)=MP$LAPTIM$PL           //MAX
          TEST L     MP$LAPTIM$PL,ML(TRKLDTIM,100,5),*+2
          BLET       ML(TRKLDTIM,100,5)=MP$LAPTIM$PL           //MIN
          TERMINATE
*
* REJECT % SUBROUTINE
*
 RPCT00   TEST LE    PF(PCT),50,RPCT50                //>100%
 RPCT10   TEST E     PF$CTR@FN12,0,RPCT70             //NO;REJECT?
          BLET       PF(RJCT)=1                       //YES;TAG
          TRANSFER   ,PF(SUBR)+1                      //RETURN
 RPCT50   TEST NE    PF(PCT),100,(PF$SUBR+1)          //100%
          TEST E     PF$CTR@2,0,RPCT60                //50% GET REJECT
 RPCT55   BLET       PF(RJCT)=1                       //YES;TAG
          TRANSFER   ,PF(SUBR)+1                      //RETURN
 RPCT60   BLET       PF(PCT)=PF(PCT)-50               //REDUCE ORIGINAL BY 50%
          TRANSFER   ,RPCT10
 RPCT70   TEST G     FN13,0,(PF$SUBR+1)               //2NDARY REJECT ADD?
          TEST E     PF$CTR@FN13,0,(PF$SUBR+1)        //NO;REJECT?
          BLET       PF(RJCT)=1                       //YES;TAG
          TRANSFER   ,PF(SUBR)+1                      //RETURN
*
* FIND MODULE # SUBROUTINE
*
 FNDMOD   BLET       PF(MOD)=0                        //ZERO OUT
 FNDM00   BLET       PF(MOD)=PF(MOD)+1                //BUMP
          TEST NE    &MODID(PF$MOD),&DUM,(PF$SUBR+1)  //RETURN IF MATCH
          TRANSFER   ,FNDM00
*----------------------------
* DOWN TIME LOGIC
*----------------------------
 DWT000   TEST NE    PF(JNDX),0,DWT100                //ANY DELAY TIME SPECIFIED?
          LINK       FAILR,FIFO                       //HOLD FAILURE PULSES
 DWT010   ADVANCE    &DELAY1(PF$INDX)                 //TIME UNTIL 1ST DELAY
 DWT020   LOGIC S    450+PF(INDX)                     //CREATE DELAY STOPPAGE
 SCOLOR3  MACRO      DOBJ,PF$INDX,'RED'
          ADVANCE    &DTIM(PF$INDX)                   //DELAY TIME
 SCOLOR3  MACRO      DOBJ,PF$INDX,'BAC'
          LOGIC C    450+PF(INDX)                     //REMOVE BLOCKAGE
```

```
         ADVANCE    &MTBF(PF$INDX)                  //MEAN TIME BETWEEN FAILURE
         LOOP       LCTR$PF,DWT020                  //CONTINUE W/ #DELAYS
DWT100   TERMINATE                                  //NUMBER COMPLETE
*
* DAY DEFINITION AND SIMULATION DURATION
*
SIMC00   BLET       &SDAY=1                         //STARTING DAY
         LOGIC S    DINIT                           //RELEASE REST
         TRANSFER   ,SIMC15
SIMC10   BPUTPIC    &SDAY                           //INDICATE MODEL STATUS
Simulating Production Day: **
WRITE    MACRO      DAY,&SDAY
         UNLINK     FAILR,DWT010,ALL,JNDX$PF,&SDAY  //FREE DELAYS
SIMC15   ADVANCE    1440                            //NEXT DAY
         SPLIT      1,SIMC99                        //CREATE TERMINATION PULSE
         PRIORITY   -1,YIELD                        //LET IT GET THER
         PRIORITY   0
         TEST G     N$SIMC10,0,SIMC10
         BLET       &SDAY=&SDAY+1                   //BUMP DAY
         TEST LE    &SDAY,&RUNDAYS,SIMC20
         TRANSFER   ,SIMC10                         //REPEAT
SIMC20   TERMINATE
*
* RATE COLLECTION/PLOTTING LOGIC
*
PLT000   BLET       PF(CLOC)=0
         BLET       PF(LCTR)=6
PLT010   BLET       PL(PF$LCTR)=0
         LOOP       LCTR$PF,PLT010
         BLET       PF(LOC1)=0
         SPLIT      5,PLT020,LOC1$PF
PLT020   BLET       PF(PLOC)=PF(CLOC)
         BLET       PF(CLOC)=PF(CLOC)+1
         TEST E     PF(CLOC),25,PLT030
         BLET       PF(CLOC)=1
         BLET       PF(PLOC)=0
PLT030   BLET       &NUM=&ITOCHAR(PF$CLOC)
         BLET       &NUM='#'||&NUM
         TEST NE    PF(LOC1),6,PLT040
PLOT4    MACRO      RTPLT,PF$LOC1,&NUM,PF$PLOC,PL(PF$LOC1),PF$CLOC,_
                    &PRORATE(PF$LOC1),&PCLR(PF$LOC1+1)
         TRANSFER   SBR,FLOW00,SUBR$PF
PLT040   ADVANCE    0
PLOT4    MACRO      PQPLT,PF$LOC1,&NUM,PF$PLOC,PL$CMPEST,PF$CLOC,_
                    FN$PROCQ,&PCLR(PF$LOC1)
         TRANSFER   SBR,KEYQ00,SUBR$PF
         BLET       PL(PF$LOC1)=&PRORATE(PF$LOC1)
         BLET       PL(CMPEST)=FN(PROCQ)
         BLET       &PRORATE(PF$LOC1)=0             //ZERO OUT
         TEST NE    PF(LOC1),6,PLT050
BARG4    MACRO      RT,PF$LOC1,TOP,&PRORATE(PF$LOC1)
PLT050   ADVANCE    60                              //WAIT NEXT HOUR
         TRANSFER   ,PLT020
*
* FLOW METER DATA COLLECTION
*
FLOW00   TEST G     &PRORATE(PF$LOC1),0,(PF$SUBR+1)
         BLET       ML(FLOWRT,PF$LOC1,1)=ML(FLOWRT,PF$LOC1,1)+&PRORATE(PF$LOC1)
         BLET       ML(FLOWRT,PF$LOC1,2)=ML(FLOWRT,PF$LOC1,2)+1
         BLET       ML(FLOWRT,PF$LOC1,3)=ML(FLOWRT,PF$LOC1,1)/ML(FLOWRT,PF$LOC1,2)
         TEST E     ML(FLOWRT,PF$LOC1,2),1,*+3
         BLET       ML(FLOWRT,PF$LOC1,4)=&PRORATE(PF$LOC1)          //MAX
         BLET       ML(FLOWRT,PF$LOC1,5)=&PRORATE(PF$LOC1)          //MIN
         TEST G     &PRORATE(PF$LOC1),ML(FLOWRT,PF$LOC1,4),*+2
         BLET       ML(FLOWRT,PF$LOC1,4)=&PRORATE(PF$LOC1)          //MAX
         TEST L     &PRORATE(PF$LOC1),ML(FLOWRT,PF$LOC1,5),*+2
         BLET       ML(FLOWRT,PF$LOC1,5)=&PRORATE(PF$LOC1)          //MIN
         TRANSFER   ,(PF$SUBR+1)
*
```

```
* CRITICAL QUEUE DATA COLLECTION
*
KEYQ00   BLET       ML(KEYQUE,PF$LOC1,1)=ML(KEYQUE,PF$LOC1,1)+FN(PROCQ)
         BLET       ML(KEYQUE,PF$LOC1,2)=ML(KEYQUE,PF$LOC1,2)+1
         BLET       ML(KEYQUE,PF$LOC1,3)=ML(KEYQUE,PF$LOC1,1)/ML(KEYQUE,PF$LOC1,2)
         TEST E     ML(KEYQUE,PF$LOC1,2),1,*+3
         BLET       ML(KEYQUE,PF$LOC1,4)=FN(PROCQ)         //MAX
         BLET       ML(KEYQUE,PF$LOC1,5)=FN(PROCQ)         //MIN
         TEST G     FN(PROCQ),ML(KEYQUE,PF$LOC1,4),*+2
         BLET       ML(KEYQUE,PF$LOC1,4)=FN(PROCQ)         //MAX
         TEST L     FN(PROCQ),ML(KEYQUE,PF$LOC1,5),*+2
         BLET       ML(KEYQUE,PF$LOC1,5)=FN(PROCQ)         //MIN
         TRANSFER   ,(PF$SUBR+1)
*
SIMC90   SPLIT      1,SIMC00                        //DONE INITIALIZING
         SPLIT      1,PLT000
SIMC99   TERMINATE  1
*
         START      2,NP
         RESET
         LET        &EPROD(1)=0
         LET        &EPROD(2)=0
         LET        &EPROD(3)=0
         LET        &EPROD(4)=0
         LET        &EPROD(5)=0
         LET        &EPROD(6)=0
         LET        &EPROD(7)=0
         LET        &EPROD(8)=0
         LET        &PRORATE(1)=0
         LET        &PRORATE(2)=0
         LET        &PRORATE(3)=0
         LET        &PRORATE(4)=0
         LET        &PRORATE(5)=0
         LET        &PRORATE(6)=0
         LET        &JHKCOTIM=0
         INITIAL    ML$PROD(1-100,2-9),0
         INITIAL    ML$ESYSPRF(1-10,1-5),0
         INITIAL    MH$DSHIPS(1-100,1-21),0
         INITIAL    MH$TSHIPS(1-100,1-21),0
         INITIAL    ML$FLOWRT(1-6,1-5),0
         INITIAL    ML$KEYQUE(1-6,1-5),0
         START      &RUNDAYS
         PUTPIC     &SDAY
 Simulation Completed!
*
         REPORT
         OUTPUT
*
      PUTPIC    FILE=OUT,LINES=5,CURDATE
-------------------------------------------------
 ENGINE WORKS TEST, TRIM, PAINT & SHIP SIMULATION
  ***********
-------------------------------------------------
  INPUT CONDITIONS:
      PUTPIC    FILE=OUT,LINES=10,(&TESTID,&TESTDSCR,_
                &PRODVOL(1),&PRODVOL(2),&PRODVOL(3),_
                &LBCTMAIN,&HRPRTIM,&LRPRTIM,&LRPRRJ*100.0,_
                &CRPRTIM,&CRPRRJ*100.0,&DOCK)
      TEST: *
      SCENARIO: *
      AVG. LINE RATE-1ST:       ***.* ENGINES/SHIFT
      AVG. LINE RATE-2ND:       ***.* ENGINES/SHIFT
      AVG. LINE RATE-3RD:       ***.* ENGINES/SHIFT
      # LOAD BARS - MAIN:       ***
      HEAVY REPAIR:             ***.* MINS.
      LIGHT REPAIR:             ***.* MINS. @ ***% REJECT RATE
      CELL DELAY:               ***.* MINS. @ ***% DELAY RATE
      # EFFECTIVE DOCKS:         **
      PUTPIC   FILE=OUT,LINES=8,(&SDAY)
```

```
------------------------------------------------------------

 RESULTS AFTER: ** SIMULATION DAYS
------------------------------------


 ENGINE PRODUCTION SUMMARY:
---------------------------
          PUTPIC     FILE=OUT,LINES=9,&EPROD(1),FLT(&EPROD(1)/&SDAY),_
                     (&EPROD(&J),FLT(&EPROD(&J)/&SDAY),&J=3,5),_
                     &EPROD(7),FLT(&EPROD(7)/&SDAY),_
                     MH(DSHIPS,100,21),FLT(MH(DSHIPS,_
                     100,21))/FLT(&SDAY),MH(TSHIPS,100,21),FLT(MH(TSHIPS,100,21))/_
                     FLT(&SDAY)
                             TOTAL     AVG./DAY
                             -----     --------
 J-HOOK PRODUCTION:          *****       ***.*
 TEST PRODUCTION:            *****       ***.*
 CUSTOM TRIM PRODUCTION:     *****       ***.*
 FINAL TRIM PRODUCTION:      *****       ***.*
 PAINT PRODUCTION:           *****       ***.*
 ENGINE SHIPPED:             *****       ***.*
 TRUCKS SHIPPED:               ***        **.*

          PUTPIC     FILE=OUT,LINES=11,(SA(EWIPQ),SM(EWIPQ),ML(ESYSPRF,1,5),_
                     S(EWIPQ),CA(FINV),CM(FINV),ML(ESYSPRF,2,5),CH(FINV),_
                     QA(TGRIDS),QM(TGRIDS),Q(TGRIDS),SA(TOTALQ),SM(TOTALQ),_
                     ML(ESYSPRF,3,5),S(TOTALQ),SA(DOCKS),SM(DOCKS),S(DOCKS))

 ENGINE PROCESS SUMMARY:
---------------------------
                                           AVG.      MAX.      MIN.     CURRENT
                                          -----     -----     -----     -------
 # ENGINES IN PROCESS/ J-HOOK TO 572:    ****.*     *****     *****      *****
 # ENGINES IN 572 (TRUCK GRIDS):         ****.*     *****     *****      *****
 # TRUCK GRIDS:                          ****.*     *****     -----      *****
 TOTAL ENGINES AFTER J-HOOK:             ****.*     *****     *****      *****
 TRUCK DOCK USAGE SUMMARY:               ****.*     *****     -----      *****

          PUTPIC     FILE=OUT,LINES=4,((ML(PROTIME,100,&J)/1440.0,&J=3,5),_
                     (ML(WHSETIM,100,&J)/1440.0,&J=3,5),(ML(TRKLDTIM,100,&J)/1440.0,_
                     &J=3,5))
 PROCESS TIME IN DAYS/ J-HOOK TO 572:    ****.*     ***.*     ***.*
 WAREHOUSE TIME IN DAYS:                 ****.*     ***.*     ***.*
 TRUCK LOAD TIME IN DAYS:                ****.*     ***.*     ***.*

          PUTPIC     FILE=OUT,LINES=8,(ML(SEQVAR,100,&J),&J=3,5)
 ENGINE FINISH SEQUENCE VARIATION:       ****.*     *****     *****


 FLOW RATE BY DEPARTMENT:
---------------------------
              TOTAL      # ENGINES   #SHIFT   DAYS/   EFFECTIVE    CALCULATED
 DEPARTMENT   PRODUCED     /DAY       /DAY    WEEK    MINS./DAY    FLOW RATE
(MINS/ENGINE)
 ----------   --------   ---------   ------   ----    ----------   ---------
          DO         &I=1,8
          IF         &OPSFT(&I)>0
          IF         &EPROD(&I)>0
          PUTPIC     FILE=OUT,(&MODID(&I),&EPROD(&I),FLT(&EPROD(&I)/&SDAY),&OPSFT(&I),_
                     &WDAYS(&I),&EFMIN(&I),FLT(&EFMIN(&I)*&SDAY/&EPROD(&I)))
  *****        *****       ***.*       *       *       ****          ***.*
          ENDIF
          ENDIF
          ENDDO
          PUTPIC     FILE=OUT,LINES=11,NSJHKCHG,FLT(NSJHKCHG)/&SDAY,_
                     &JHKCOTIM/60.0,&JHKCOTIM/60.0/&SDAY,FLT(_
                     NSJHKCHG)/NSJHKTOT*100.0
```

```
J-HOOK CHANGEOVERS                  TOTAL       AVG./DAY
-------------------                 -----       --------
         # CHANGEOVERS:             ****          ***.*
         CHANGEOVER TIME(HOURS):    ***.*         ***.*
         % CHANGEOVER:              ***.* %        ----

HOURLY FLOW METER SUMMARY (UNITS/HOUR)
---------------------------------------
AREA                             AVG.    MAX.    MIN.
---------------                 -----   -----   -----
         DO         &I=1,5
         IF         &I=1
         LET        &DUM='JHOOK'
         ELSEIF     &I=2
         LET        &DUM='TEST'
         ELSEIF     &I=3
         LET        &DUM='CUSTOM TRIM'
         ELSEIF     &I=4
         LET        &DUM='FINAL TRIM'
         ELSE
         LET        &DUM='PAINT'
         ENDIF
         PUTPIC     FILE=OUT,&DUM,(ML(FLOWRT,&I,&J),&J=3,5)
****************                ***.*   ***.*   ***.*
         ENDDO
         PUTPIC     FILE=OUT,LINES=5

CRITICAL QUEUE SUMMARY
-----------------------
AREA                             AVG.    MAX.    MIN.
---------------                 -----   -----   -----
         DO         &I=1,6
         IF         &I=1
         LET        &DUM='EMPTY'
         ELSEIF     &I=2
         LET        &DUM='ATTIC'
         ELSEIF     &I=3
         LET        &DUM='TEST LOOP'
         ELSEIF     &I=4
         LET        &DUM='CUSTOM TRIM'
         ELSEIF     &I=5
         LET        &DUM='FINAL TRIM'
         ELSE
         LET        &DUM='PAINT'
         ENDIF
         PUTPIC     FILE=OUT,&DUM,(ML(KEYQUE,&I,&J),&J=3,5)
****************                ***.*   ***.*   ***.*
         ENDDO
         PUTPIC     FILE=OUT,LINES=7,&DUM

ENGINE PRODUCTION DETAIL:
---------------------------
DAILY ENGINES SHIPPED:
                    PRODUCTION DAYS:
ENGINE                 1    2    3    4    5    6    7    8    9   10    TOTAL
---------------     ---- ---- ---- ---- ---- ---- ---- ---- ---- ----   ------
         DO         &I=1,99
         IF         MH(DSHIPS,&I,21)>0
         PUTPIC     FILE=OUT,&PARTNO(&I),(MH(DSHIPS,&I,&J),&J=1,10),_
                    MH(DSHIPS,&I,21)
****************    ***  ***  ***  ***  ***  ***  ***  ***  ***  ***    ******
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=7,(MH(DSHIPS,100,&J),&J=1,10),_
                    MH(DSHIPS,100,21)
------------------------------------------------------------------------------
TOTAL               ***  ***  ***  ***  ***  ***  ***  ***  ***  ***    ******

DAILY TRUCK SHIPMENT BY CUSTOMER:
```

```
                        PRODUCTION DAYS:
CUSTOMER                 1    2    3    4    5    6    7    8    9   10     TOTAL
---------------         ---- ---- ---- ---- ---- ---- ---- ---- ---- ----   ------
          DO           &I=1,99
          IF           MH(TSHIPS,&I,21)>0
          PUTPIC       FILE=OUT,&CUSTID(&I),(MH(TSHIPS,&I,&J),&J=1,10),_
                       MH(TSHIPS,&I,21)
***************         ***  ***  ***  ***  ***  ***  ***  ***  ***  ***    ******
          ENDIF
          ENDDO
          PUTPIC        FILE=OUT,LINES=8,(MH(TSHIPS,100,&J),&J=1,10),_
                        MH(TSHIPS,100,21)
----------------------------------------------------------------------------------
TOTAL                   ***  ***  ***  ***  ***  ***  ***  ***  ***  ***    ******

ENGINE PROCESS DETAIL:
----------------------
PROCESS TIME BETWEEN J-HOOK & 572 (IN HOURS):
ENGINE                 # COMPLETE    AVG.     MAX.     MIN.
---------------        ----------    -----    -----    -----
          DO           &I=1,99
          IF           ML(PROTIME,&I,1)>0
          PUTPIC        FILE=OUT,&PARTNO(&I),ML(PROTIME,&I,2),_
                        (ML(PROTIME,&I,&J)/60.0,&J=3,5)
***************          ****       ****.*   ****.*   ****.*
          ENDIF
          ENDDO
          PUTPIC        FILE=OUT,LINES=6,ML(PROTIME,100,2),_
                        (ML(PROTIME,100,&J)/60.0,&J=3,5)
---------------------------------------------------------
TOTAL:                   ****       ****.*   ****.*   ****.*

WAREHOUSE TIME (IN HOURS):
ENGINE                 # COMPLETE    AVG.     MAX.     MIN.
---------------        ----------    -----    -----    -----
          DO           &I=1,99
          IF           ML(WHSETIM,&I,1)>0
          PUTPIC        FILE=OUT,&PARTNO(&I),ML(WHSETIM,&I,2),_
                        (ML(WHSETIM,&I,&J)/60.0,&J=3,5)
***************          ****        ***.*    ***.*    ***.*
          ENDIF
          ENDDO
          PUTPIC        FILE=OUT,LINES=6,ML(WHSETIM,100,2),_
                        (ML(WHSETIM,100,&J)/60.0,&J=3,5)
---------------------------------------------------------
TOTAL:                   ****       ****.*   ****.*   ****.*

TRUCK GRID TIME (AWAITING SHIPMENT) IN HOURS:
CUSTOMER               # COMPLETE    AVG.     MAX.     MIN.
---------------        ----------    -----    -----    -----
          DO           &I=1,49
          IF           ML(GRIDTIM,&I,1)>0
          PUTPIC        FILE=OUT,&CUSTID(&I),ML(GRIDTIM,&I,2),_
                        (ML(GRIDTIM,&I,&J)/60.0,&J=3,5)
***************          ****       ****.*   ****.*   ****.*
          ENDIF
          ENDDO
          PUTPIC        FILE=OUT,LINES=6,ML(GRIDTIM,100,2),_
                        (ML(GRIDTIM,100,&J)/60.0,&J=3,5)
---------------------------------------------------------
TOTAL:                   ****       ****.*   ****.*    ***.*

TRUCK LOAD TIME IN HOURS:
CUSTOMER               # COMPLETE    AVG.     MAX.     MIN.
---------------        ----------    -----    -----    -----
          DO           &I=1,49
          IF           ML(TRKLDTIM,&I,1)>0
          PUTPIC        FILE=OUT,&CUSTID(&I),ML(TRKLDTIM,&I,2),_
                        (ML(TRKLDTIM,&I,&J)/60.0,&J=3,5)
```

```
***************        ****     ****.*  ****.*  ****.*
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=6,ML(TRKLDTIM,100,2),_
                    (ML(TRKLDTIM,100,&J)/60.0,&J=3,5)
-------------------------------------------------------
TOTAL:                 ****     ****.*  ****.*  ****.*

FINISH SEQUENCE VARIANCE:
ENGINE              # COMPLETE    AVG.    MAX.    MIN.
----------------    ----------   -----   -----   -----
         DO         &I=1,99
         IF         ML(SEQVAR,&I,2)>0
         PUTPIC     FILE=OUT,&PARTNO(&I),(ML(SEQVAR,&I,&J),&J=2,5)
***************          ****     ***.*   *****   *****
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=9,(ML(SEQVAR,100,&J),&J=2,5),&WDAYS(2)
-------------------------------------------------------
TOTAL:                   ****     ***.*   *****   *****

TECHNICIAN PERFORMANCE BY DEPARTMENT
DEPT: 568
OPERATING DAYS/WEEK: *
                             # ENGINES     AVG. TIME/
TECHNICIAN         SHIFT     PROCESSED     ENGINE       % ULT.
----------------------------------------------------------------
         DO         &I=301,400
         LET        &K=&I-300
         IF         MX(TCHASN,&K,1)=2
         PUTPIC     FILE=OUT,&TCHNM(&K),MX(TCHASN,&K,2),FC(&I),FT(&I),_
                    FRV(&I)/10.0
 **************     *         *****          ***.*        ***.* %
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=6,&WDAYS(3)

DEPT: 569
OPERATING DAYS/WEEK: *
                             # ENGINES     AVG. TIME/
TECHNICIAN         SHIFT     PROCESSED     ENGINE       % ULT.
----------------------------------------------------------------
         DO         &I=301,400
         LET        &K=&I-300
         IF         (MX(TCHASN,&K,1)=3)OR(MX(TCHASN,&K,1)=8)
         PUTPIC     FILE=OUT,&TCHNM(&K),MX(TCHASN,&K,2),FC(&I),FT(&I),_
                    FRV(&I)/10.0
 **************     *         *****          ***.*        ***.* %
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=6,&WDAYS(4)

DEPT: 570
OPERATING DAYS/WEEK: *
                             # ENGINES     AVG. TIME/
TECHNICIAN         SHIFT     PROCESSED     ENGINE       % ULT.
----------------------------------------------------------------
         DO         &I=301,400
         LET        &K=&I-300
         IF         MX(TCHASN,&K,1)=4
         PUTPIC     FILE=OUT,&TCHNM(&K),MX(TCHASN,&K,2),FC(&I),FT(&I),_
                    FRV(&I)/10.0
 **************     *         *****          ***.*        ***.* %
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=6,&WDAYS(5)

DEPT: 571
OPERATING DAYS/WEEK: *
```

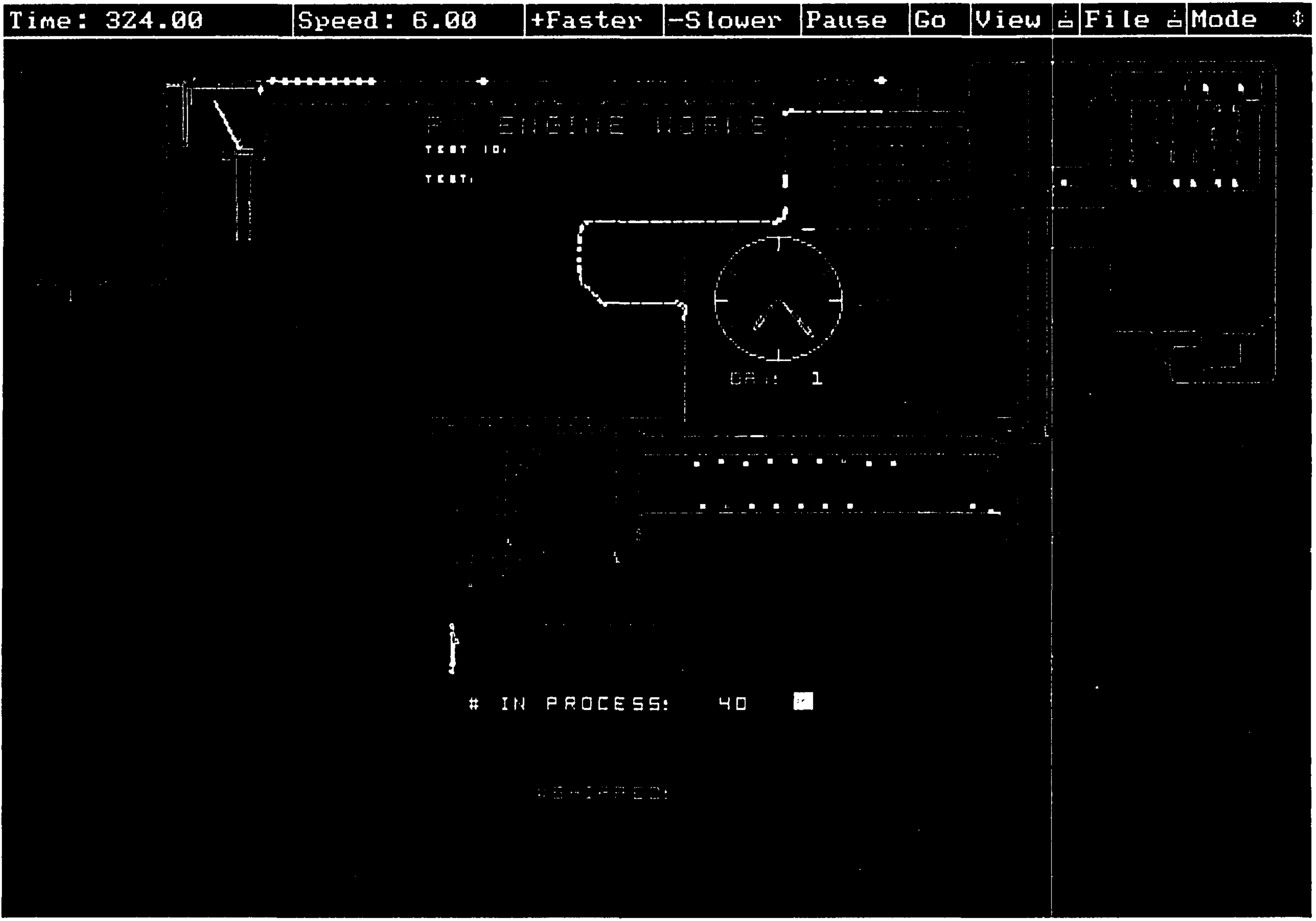
```
                            # ENGINES      AVG. TIME/
TECHNICIAN        SHIFT     PROCESSED      ENGINE        % ULT.
----------------------------------------------------------------
         DO         &I=301,400
         LET        &K=&I-300
         IF         MX(TCHASN,&K,1)=5
         PUTPIC     FILE=OUT,&TCHNM(&K),MX(TCHASN,&K,2),FC(&I),FT(&I),_
                    FRV(&I)/10.0
 **************     *           *****        ***.*        ***.* %
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=6,&WDAYS(6)

DEPT: 572
OPERATING DAYS/WEEK: *
                            # TRUCKS      AVG. TIME/
TECHNICIAN        SHIFT     PROCESSED       TRUCK        % ULT.
----------------------------------------------------------------
         DO         &I=301,400
         LET        &K=&I-300
         IF         MX(TCHASN,&K,1)=6
         PUTPIC     FILE=OUT,&TCHNM(&K),MX(TCHASN,&K,2),FC(&I),FT(&I),_
                    FRV(&I)/10.0
 **************     *           *****        ***.*        ***.* %
         ENDIF
         ENDDO
         PUTPIC     FILE=OUT,LINES=6,&WDAYS(6)

DEPT: 570 PAINT
OPERATING DAYS/WEEK: *
                            # ENGINES      AVG. TIME/
TECHNICIAN        SHIFT     PROCESSED      ENGINE        % ULT.
----------------------------------------------------------------
         DO         &I=301,400
         LET        &K=&I-300
         IF         MX(TCHASN,&K,1)=7
         IF         (MX(TCHASN,&K,3)=PNTTC)OR(MX(TCHASN,&K,4)=PNTTC)
         PUTPIC     FILE=OUT,&TCHNM(&K),MX(TCHASN,&K,2),FC(&I)*2,FT(&I)/2.0,_
                    FRV(&I)/10.0
 **************     *           *****        ***.*        ***.* %
         ELSE
         PUTPIC     FILE=OUT,&TCHNM(&K),MX(TCHASN,&K,2),FC(&I),FT(&I),_
                    FRV(&I)/10.0
 **************     *           *****        ***.*        ***.* %
         ENDIF
         ENDIF
         ENDDO
*
* Test Summary
*
         PUTPIC     FILE=TSUM,&TESTID,&TESTDSCR,CURDATE,FLT(&EPROD(1)/&SDAY),_
                    (FLT(&EPROD(&J)/&SDAY),&J=3,5),FLT(&EPROD(7)/&SDAY),_
                    FLT(MH(DSHIPS,100,21))/FLT(&SDAY),SA(EWIPQ),SM(EWIPQ),_
                    CA(FINV),CM(FINV),SA(TOTALQ),SM(TOTALQ),_
                    (ML(PROTIME,100,&J)/1440.0,&J=3,4),_
                    (ML(GRIDTIM,100,&J)/1440.0,&J=3,4),(ML(SEQVAR,100,&J),&J=3,5)
** * ****** ***.* ***.* ***.* ***.* ***.* ***.* ****.* **** ****.* **** ****.* **** _
**.* ** **.* ** ***.* ***** *****
         CLOSE      TSUM
         END
```

APPENDEX C

Snapshot of Animated Simulation Run



Time: 324.00

Speed: 6.00

DAY: 1

IN PROCESS: 40

#SHIPPED: