A MODEL FOR PREDICTING AND MANAGING A PRODUCTION RAMP-UP OF A NEW PRODUCT

by

JAMES WILLIAM CASAMENTO

S. B. Mechanical Engineering -- Massachusetts Institute of Technology (1988)

Submitted to the
Department of Management and
Department of Mechanical Engineering
in partial fulfillment of the requirements for the degrees of

MASTER OF SCIENCE IN MANAGEMENT and
MASTER OF SCIENCE IN MECHANICAL ENGINEERING

in conjunction with the LEADERS FOR MANUFACTURING PROGRAM at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1992

© Massachusetts Institute of Technology 1992 (All rights reserved)

Signature of Author ______________________________________

Department of Management
Department of Mechanical Engineering
May, 1992

Certified by ______________________________________

George Chryssolouris
Associate Professor of Mechanical Engineering
Thesis Supervisor

Certified by ______________________________________

Alfredo M. Kofman
Assistant Professor of Management
Thesis Supervisor

Accepted by ______________________________________

Jeffrey A. Barks
Associate Dean, Sloan Master's and Bachelor's Programs
A MODEL FOR PREDICTING AND MANAGING A PRODUCTION RAMP-UP OF A NEW PRODUCT

by

JAMES WILLIAM CASAMENTO

Submitted to the Departments of Management and Mechanical Engineering on May 8, 1992 in partial fulfillment of the degrees of MASTER OF SCIENCE IN MANAGEMENT and MASTER OF SCIENCE IN MECHANICAL ENGINEERING

ABSTRACT

The ability to both accurately forecast and better manage production ramp-ups is vital to the success of a new product. Accurate forecasts provide firms with the opportunity to accurately plan annual budgets, new product introductions and marketing plans for products that are being obsoleted. Better management of ramp-ups allows firms to get new products to market more quickly. Any delay in the availability of a new product results in a one-day delay in revenue generation in the best case and in revenue lost forever in the worst case. In both cases a significant delay can have a significant negative effect on the overall lifetime profitability of the product.

The final assembly of a new product at Eastman Kodak Company was studied. The model for manual assembly operations proposed here links ramp-up performance to the human cognitive learning of manual assembly operations by direct-labor operators. The longer it takes operators to reach a certain level of proficiency, the longer ramp-up will take. The capacity of the training process which operators must go through before starting to work on the product defines when operators can start their learning process. With learning linked to cumulative production, any factor which prevents operators from producing more units also prevents them from learning. Thus other events are modeled as factors which impede production. Problems due to a poorly designed product require operators to troubleshoot and make repairs limiting their rate of output. Parts shortages halt operator activity altogether. The occurrence of part shortages are linked to the generation of engineering change orders which is modeled as an engineering learning process.

A continuous simulation Fortran model was constructed to simulate the effects of each of these factors on output during ramp-up. Actual output fell within the modeled range of possible outputs for the actual measured parameter values. Theoretical analysis showed that decreasing the standard time of the longest operation would result in substantial improvement in ramp-up performance. The model verified this fact. Other means of improving ramp-up performance included decoupling learning from production and installing parts in off-line sub-assemblies.

Thesis supervisors:

George Chryssolouris, Associate Professor of Mechanical Engineering
Alfredo M. Kofman, Assistant Professor of Management
Richard B. Hamer, Eastman Kodak Company
Dennis E. DeMeyer, Eastman Kodak Company
A. **ACKNOWLEDGEMENTS**

I would like to thank the following people for helping me through this project:

The Leaders for Manufacturing Program sponsors and administrators who funded this project and without whom this remarkable program would not be possible.

Dick Hamer, Dennis DeMeyer and all of the other people at Eastman Kodak who provided the needed support to get the research done.

My thesis advisors at MIT, Fred Kofman and George Chryssolouris.

My LFM classmates who are the greatest support network I have ever had. May we meet again in the boardrooms of American industry.

Eva, for loving me just the way I am.

My parents who instilled my values in me at a young age and are largely responsible for the person I am today.
# TABLE OF CONTENTS

ABSTRACT .................................................................................................................. 3
ACKNOWLEDGEMENTS ................................................................................................. 4
TABLE OF CONTENTS ............................................................................................... 5
TABLE OF FIGURES .................................................................................................... 6
I  INTRODUCTION ....................................................................................................... 7
   Project Scope and Motivation .................................................................................. 7
   Primary Issues Addressed ..................................................................................... 8
   Specifics of this Case ............................................................................................. 9
II  THEORETICAL FRAMEWORK FOR DISCUSSION OF RAMP-UP ...................... 12
   Background on Topic ............................................................................................ 12
   Basic Framework of Model .................................................................................. 13
   Factors to be Studied ............................................................................................ 13
III ANALYSIS OF FACTORS FOR THIS PRODUCT ..................................................... 25
   Means of Data Collection .................................................................................... 25
   Analysis of Important Factors .............................................................................. 26
   The Learning Process ........................................................................................... 27
   The Training Process ............................................................................................ 30
   Design Performance ............................................................................................. 32
   Parts Availability .................................................................................................. 34
   Design Stability ..................................................................................................... 37
IV  MODEL ..................................................................................................................... 40
   The Learning Process ........................................................................................... 40
   The Training Process ............................................................................................ 41
   Design Performance ............................................................................................. 41
   Parts Availability .................................................................................................. 43
   Design Stability ..................................................................................................... 44
   Model Performance ............................................................................................... 44
V  APPLICATION OF MODEL TO THIS PRODUCT ...................................................... 45
   Parameters ............................................................................................................. 45
   Validation of Model .............................................................................................. 46
   Analysis of Opportunities for Improvement ....................................................... 48
   Best-case Ramp-up ............................................................................................... 57
   When to Start the Ramp-Up ................................................................................ 58
VI  CONCLUSION, RECOMMENDATIONS AND FUTURE WORK ....................... 60
   Conclusion ............................................................................................................. 60
   Recommendations ............................................................................................... 62
   Opportunities for Future Work ........................................................................... 64
References ................................................................................................................... 67
Appendix: Fortran Program ....................................................................................... 72
**TABLE OF FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Downward Adjustments in Master Production Schedule</td>
<td>7</td>
</tr>
<tr>
<td>2.1</td>
<td>Series vs. Parallel Example</td>
<td>17</td>
</tr>
<tr>
<td>2.2</td>
<td>Effect of Part Shortage</td>
<td>22</td>
</tr>
<tr>
<td>3.1</td>
<td>Assembly and Repair Time vs. Operator Iteration for Assembly #5</td>
<td>28</td>
</tr>
<tr>
<td>3.2</td>
<td>Assembly and Repair Time vs. Operator Iteration for Assembly #6</td>
<td>29</td>
</tr>
<tr>
<td>3.3</td>
<td>Assembly Time vs. Operator Iteration for Checkout</td>
<td>30</td>
</tr>
<tr>
<td>3.4</td>
<td>Trend of Repair and Troubleshooting Time in Checkout</td>
<td>33</td>
</tr>
<tr>
<td>3.5</td>
<td>Length of Engineering Holds</td>
<td>34</td>
</tr>
<tr>
<td>3.6</td>
<td>Histogram of Length of Part Shortages</td>
<td>36</td>
</tr>
<tr>
<td>3.7</td>
<td>Engineering Change Orders/units shipped vs. Cumulative Production</td>
<td>38</td>
</tr>
<tr>
<td>3.8</td>
<td>Histogram of Weeks After ECO of Occurrence of Part Shortage</td>
<td>39</td>
</tr>
<tr>
<td>5.1</td>
<td>Actual Production vs. Average of 100 Model Simulations</td>
<td>47</td>
</tr>
<tr>
<td>5.2</td>
<td>Range of Simulations</td>
<td>48</td>
</tr>
<tr>
<td>5.3</td>
<td>Model Output Under Different Configuration Scenarios</td>
<td>50</td>
</tr>
<tr>
<td>5.4</td>
<td>Model Output Under Different Configuration Scenarios if Changed Now</td>
<td>52</td>
</tr>
<tr>
<td>5.5</td>
<td>Effect of Reducing Part Shortages in Current Configuration</td>
<td>54</td>
</tr>
<tr>
<td>5.6</td>
<td>Effect of Improving Design Performance in Current Configuration</td>
<td>55</td>
</tr>
<tr>
<td>5.7</td>
<td>Effect of Reducing Part Shortages in Scenario 2</td>
<td>56</td>
</tr>
<tr>
<td>5.8</td>
<td>Effect of Improving Design Performance in Scenario 2</td>
<td>57</td>
</tr>
<tr>
<td>5.9</td>
<td>Best-case Production under Different Scenarios</td>
<td>58</td>
</tr>
</tbody>
</table>
I. INTRODUCTION

Project Scope and Motivation

The goal of this project was to propose an analytical model of production ramp-up which can be used to more accurately forecast and better manage ramp-ups. The research was performed at a division of Eastman Kodak Company using the final assembly of a new product as a case study. The project was motivated by the division's failure to meet early production schedules for the product and a monthly downward adjustment in the master production schedule (Figure 1.1). These events led the division's personnel to question the methodologies used to forecast ramp-up production schedules.

| Figure 1.1: Downward Adjustments in Master Production Schedule$^1$ |
| Production rate |
| Month 0 |
| Month 4 |
| Month 6 |
| Month 7 |
| Actual |

In this case the division's management set an aggressive ramp-up schedule based on the performance of a prior ramp-up several years earlier. The production department

$^1$The first "month" actually represents a month and two weeks.
had reservations about the proposed ramp-up schedule, but did not have the data on the prior ramp-up or an analytical framework on which to base a different forecast. As a result of the overly optimistic forecast, the division made a premature product introduction announcement which led to a sharp reduction in sales for the division. Customers could not purchase the new product because it was not yet available and many were unwilling to purchase one of the firm's other products knowing that they would soon be obsoleted.

**Primary Issues Addressed**

The model presented here can help managers more accurately forecast and better manage ramp-ups. Although the need for the former was the primary driver in the decision to embark on this project, the most long-term benefit may ultimately result from the latter.

**Importance of Good Production Forecasts**

The division's budgeting process begins with a revenue forecast for the coming year. The expense and staffing levels needed to achieve that revenue are estimated and an annual budget is presented to the CEO for approval. Once the budget is in place, the profitability of the division is largely a function of its ability to meet the predicted revenue stream. Meeting the schedule for a week or a month is important, but meeting the schedule for the year is of paramount importance.

**Importance of Management of Ramp-ups**

Time to market is a critical factor in determining the overall lifetime profitability of a product. If the product is one-of-a-kind, potential customers will likely wait as long as it takes for the product to reach the market. If that is the case, each delay of one day in product availability may result in only a one-day delay in that day's revenue. If, as was the case with this product, it is entering a competitive marketplace, a potential customer may simply purchase a competing product and a sale will be lost forever. Assuming the product has a finite market lifetime, there ultimately will be fewer units over which to amortize development and tooling costs.
Reducing product development time is usually the activity on which attention is focused in order to speed a product to market. However, delivering that first, second and third unit to the market is in large part also a function of the production department's ability to quickly ramp-up.

Specifics of this Case

Overview of Development Process

Not unlike many other companies, this division's research and development function is divided into two groups. The research group is made up of research engineers responsible for discovering, developing and bringing to maturity new technologies. The project engineering group is comprised of project engineers who are responsible for developing new products using the new technologies. When the decision to develop a new product is made, a project manager is assigned overall project management responsibility for the project. The project manager reports to the head of research and development and is given access to resources within the the project engineering group.²

In the mid- to late-eighties the division embarked on three major product development projects, each involving several new technologies. Product development for this product, the time required from the start of development to the time the product is granted shipping approval³, was originally slated for 2 1/2 years. The product studied here involved twenty new technologies and more new parts than any new product introduced by the division in several years.⁴ As development proceeded it became apparent that many of

²As a result of the less than hoped for performance in the development of this and other products, the entire development process in the division has been the focus of intense scrutiny. Recent changes include movement towards a "heavyweight" project manager. The organization structure described here is the one that was in place for the greater part of the development process of this product.
³"Shipping approval" means that the product is of sufficient quality that it can be delivered to external customers.
⁴One of the other projects was on the same scale as this one and the other project was not far behind.
the new technologies were not mature and would require significant reengineering. With development resources already tight as a result of the simultaneous projects, this created a great strain on the division. None of the projects received all of the resources required to complete development on schedule and all three fell well behind.

With development behind, the division started manufacturing the product before several design issues were resolved in an effort to get the product to market as close to schedule as possible. After production of some units, a major design flaw became apparent and assembly stopped while engineers redesigned a subsystem. Several months after this aborted attempt at production ramp-up, the division embarked on another ramp-up effort. At this time the development group believed that they were only about three months away from shipping approval on the product. As the second ramp-up proceeded, many other design problems appeared resulting in many engineering change orders. Finally about eight months after the beginning of the second ramp-up and over two years behind schedule the product was granted shipping approval. This paper studies the second ramp-up attempt.

Exhibit 1 shows a flowchart of the process of getting from design to steady-state production. Ideas for assembly tooling and procedures are developed during the design process. Using prototyped parts, engineering models are produced out of which preliminary assembly procedures and tooling are developed. When production quality parts are delivered, a witness assembly is performed during which assembly procedures are verified and finalized. Once the witness assembly has been performed, the production ramp-up can begin. Once production starts, there is a process of operator learning and

---

5 The project’s chief engineer commented “With 20 new technologies, there is a 100% probability that at least one is not mature and if even one is not mature, it can slow the whole process down.”

6 Although much could probably be learned from the first ramp-up attempt, there was no documentation of it. Data collection in the division was generally quite poor and particularly so during ramp-up. In some cases, information which could provide valuable insight into ramp-up is not tracked until shipping approval is granted.
training which leads to faster production. Throughout the entire assembly process
development process, there is a feedback loop to design. Thus production improvements
are a result of operator learning and design improvements. Production ramp-up then
involves improvement over two loops until a steady-state production rate is attained.

Overview of Assembly Process

From the beginning the development team decided to design the process using
assembly modules as opposed to the more assembly line-like approach used by the existing
product. The often-cited advantages of such job enlargement are increased flexibility,
greater employee satisfaction and less susceptibility to employee absenteeism. As
development fell behind, the decision was made to convert assembly of the existing product
to a more modular build. There were a couple of glitches, but the process was largely
completed without major problems.

Exhibit 2 shows a process-flow diagram of the modular assembly of this product
giving the relative standard time for each operation. All of the major subassemblies shown
and the final assembly and checkout of the product were performed in one production area
under the direction of a single Operations Manager. The new product was built in the same
facility as the product which was to be superseded by it. For a while, the two products
were built alongside each other as the new product ramped up and the old product ramped
down. There are six major assembly processes, fed by eleven major sub-assembly
processes. When everything but the external panels have been assembled, the product goes
through a checkout operation which is intended to involve some minor adjustments and
verification that the product’s performance meets specifications. During ramp-up,
however, this process always involved some repair and troubleshooting. After checkout,
the product goes through two other processes and is delivered to the stockroom for final
packaging and shipping.
II THEORETICAL FRAMEWORK FOR DISCUSSION OF RAMP-UP

Ramp-up is the time from the beginning of production of the first unit of a product to the achievement of a long-run steady-state output rate. Simply stated, the output rate of a system is a function of the capacity of the system to process units of production and the length of time required to process each unit.

\[
\text{output rate} = \frac{\text{capacity}}{\text{processing time}}
\]

Ramp-up is thus characterized by two activities: (1) installation of the needed capacity and (2) reduction of processing times to an expected standard. In this case reduction of processing times was the critical path to attainment of the desired steady-state rate.

Background on Topic

Ramp-up is a learning process. Since the 1930s the framework on which research on learning in a manufacturing environment has focused is the learning curve.\textsuperscript{7} Recent work on learning in a manufacturing environment has focused on identifying organizational factors which contribute to an organization-wide learning rate. Adler and Clark (1991) tried to apply an organizational learning curve to an electronic equipment company. They investigated the notion that the rate of learning was a combination of autonomous learning-by-doing (first-order learning) and management-induced learning through engineering changes and training (second-order learning). Despite the fact that three-quarters of the engineering changes generated during early production of the product were motivated by cost and manufacturability concerns and not design errors, the engineering change process itself was found to have a disruptive effect on productivity.

Bohn (1987, 1988) studied semiconductor manufacturing. He discredits the learning curve as a model for organizational learning instead suggesting a mathematical

\textsuperscript{7}The learning curve is also referred to as the experience curve or manufacturing progress function. See Dutton, Thomas and Butler (1984) for a historical survey of the topic.
framework based on learning by experimentation. Bohn agrees, however, that as a model of simple human cognitive learning, the learning curve is valid.\(^8\)

**Basic Framework of Model**

Rather than defining a rate of organizational learning as these and other authors have tried to do, the model proposed here focuses on the rate of human cognitive learning by the assembly operators as the basis of ramp-up performance for a manual assembly operation. This was justified for two reasons. First, the assembly operation was configured in such a way that achievement of long-term desired rates was dependent on operators being able to perform in the expected standard times. Second, during early production of this product, engineers were too busy dealing with design problems to focus on process improvements. Organizational factors were then studied in the framework of how they inhibit the operator learning process.

**Factors to be Studied**

Exhibit 3 is a cause and effect diagram showing some of the many factors which can affect the ramp-up process. Direct-labor learning is the primary driver of ramp-up. Through interviews and brainstorming sessions with managers at Kodak, it was determined that the other factors which had a significant effect on the ramp-up were the operator training process, the design performance to specification, parts availability and design stability. Other factors shown in Exhibit 3 such as availability of production floor space or tooling did not limit the ramp-up in this case. Since it is not possible to quantify factors which did not affect this ramp-up, only the factors listed here were investigated further.

\(^8\)Phone conversation with Professor Bohn, October 1991.
The Learning Process

Mathematical Framework

Operators cannot initially perform their operation in the standard time. The gradual improvement toward the standard was modeled using the learning curve:

\[ \text{assembly time} = a \times \text{iteration}^{-b} \] (2.1)

where \( a \) represents the length of time required for the first unit
\( b \) determines the rate of learning

The desired long-term steady-state output rate cannot be achieved until every operator is able to perform his job in the standard time.

The slope of the learning curve is defined as \( 2^{-b} \) and represents the improvement achieved as cumulative output is doubled. Thus if a learning curve has a slope of 90% and the first unit requires 10 hours to assemble, then the second unit would take 9 hours, the fourth unit 8.1 hours, the eighth unit 7.29 hours etc. A 99% curve would be considered flat, while a 70% curve would be considered steep. Thus the greater \( b \) is, the steeper the slope is.

Equation 2.1 can be generalized as:

\[ t_{i,j} = a(i)' \times s_i \times j^{-b(i)} \] (2.2)

where: \( t_{i,j} = \text{time required to perform the } i^{\text{th}} \text{ operation on the } j^{\text{th}} \text{ unit} \)
\( s_i = \text{standard time of operation } i \)
\( a'(i) = a(i)/s_i \)

If operations are of similar complexity, we make the assumption that the constants are the same for each operation and only the standard time changes. Thus

\[ t_{i,j} = a^*s_i^*j^{-b} \] (2.3)
In practice operators show improvement for some time and eventually level off at the standard time. Thus equation 2.3 becomes:

$$t_{i,j} = \text{Maximum}(a' \cdot j^{-b}, 1) \cdot s_i$$  \hspace{1cm} (2.4)

In order to reach steady-state output rates, operators must reach the standard time. The number of iterations required to reach this proficiency is found by setting equation 2.3 to $s_i$:

$$n = (a')(1/b)$$  \hspace{1cm} (2.5)

where: $n =$ number of iterations required to reach standard time

Thus the number of iterations required to reach the standard time is independent of the standard time. Assuming there are no stoppages in production, the amount of time required to reach this level of proficiency is:

$$t_r = \sum_{j=1,n} a' \cdot s_i \cdot j^{-b}$$

$$t_r = s_i \cdot C$$  \hspace{1cm} (2.6)

where: $t_r =$ time required to ramp up

$$C = \sum_{j=1,n} a' \cdot j^{-b}$$

Thus the length of time required for any operator to ramp-up is directly proportional to the standard time of the operation.

**Examples**

Let's take an example of an assembly line made up of operations of similar complexity. Assume that (1) there are $m$ operations in series; (2) each operation has a single operator; (3) operation $b$ is the operation with the longest standard time; (4) there are no stoppages in production. In this model, operations upstream of the bottleneck create a lag in the initial starting time for the bottleneck after which they can keep pace with the

---

9 Although improvement may be possible beyond the standard time, operators have no incentive to improve further. In any case if management has forecast requirements correctly, they should not want operators to work faster than the standard time because doing so would result in too much output.
bottleneck and thus have no further effect on the output time of the system. Operations
downstream of the longest operation are starved for production and thus increase lead time
by only the time that it takes those operations to process the nth unit, which is the standard
time. Therefore,

\[ t_r' = (\sum_{i=1,b-1} a_i * s_i) + s_b * C + (\sum_{i=b+1,n} s_i) \]  \hspace{1cm} (2.7)

where: \( t_r' \) = ramp-up time of system

\( s_b \) = standard time of the bottleneck (longest) operation

At sufficiently high levels of \( n \), equation 2.7 becomes:

\[ t_r' = s_b * C \] \hspace{1cm} (2.8)

Thus the total ramp-up time of the system is proportional to the standard time of the longest
operation.

This result was based on the assumption that each operation regardless of standard
time has the same resources devoted to it. What about a system in which each operation
has as many operators as it needs to maintain balance in the steady-state, that is if the
standard time is twice as long, the operation is assigned twice as many operators?
Figure 2.1 shows an example of this scenario. In Figure 2.1a, the steady-state rate of output of two operators in series performing jobs with standard times of 1 would equal the steady-state rate of output of two operators in parallel performing jobs with standard times of 2 shown in Figure 2.1b. In both cases every unit of production requires two units of time and there are two operators available to do that work. However, the operators in series will ramp-up more quickly. Figure 2.1c shows an example of this for a scenario where the standard time of the job performed by the operators in series is 1 hour, the standard time of the job performed by the operators in parallel is 2 hours, a’ equals 3 and there is an 85% learning curve. Output of the operators in parallel almost matches that of the operators in series in week one because the second operator in the series configuration is dependent on the first operator to complete the first unit of production before he can start.
However, the operators in series quickly ramp-up and by week 5 have achieved full output. On the other hand, the operators in parallel do not achieve full output until week 8.

Specifically, for the operators in series, the ramp-up time will equal:

\[
tr_s = a' \ast s_s + \sum_{j=1,n} a' \ast s_s \ast j^{-b} \tag{2.9}
\]

where: \[tr_s = \text{ramp-up time of the operators in series}\]
\[s_s = \text{standard time of operation in series}\]

therefore,

\[
tr_s = s_s \ast C \tag{2.10}
\]

The two operators in parallel will ramp-up at the same rate as each other. Noting that the standard time of the operators in parallel is twice the standard time of the operators in series,

\[
tr_p = \sum_{j=1,n} a' \ast 2 \ast s_s \ast j^{-b} \tag{2.11}
\]

where: \[tr_p = \text{ramp-up time of the operators in parallel}\]

More importantly, as shown in Figure 2.1d, after a short time cumulative output of the operators in parallel will always be lower than that of the operators in series. For the operators in series:

\[
c_{qs} = a' \ast s_s + a' \ast s_s \ast \sum_{j=1,q} j^{-b} \tag{2.12}
\]

\[
c_{qs} = a' \ast s_s \ast \sum_{j=1,q} j^{-b} \tag{2.13}
\]

where: \[c_{qs} = \text{completion time for qth unit in series}\]

The two operators in parallel will produce output at the same rate as each other, (i.e. the first and second units will be completed at the same time, the third and fourth at the same time etc.). Thus, for even q:

\[
c_{qp} = a' \ast 2 \ast s_s \ast \sum_{j=1,q/2} j^{-b} \tag{2.14}
\]

Because \(j^{-b}\) is a constantly decreasing function:

\[
\sum_{j=1,q/2} j^{-b} > \sum_{j=q/2+1,q} j^{-b}
\]

\[
2 \ast \sum_{j=1,q/2} j^{-b} > \sum_{j=1,q} j^{-b}
\]

thus,
c_qp > c_qs \hspace{1cm} (2.15)

Cumulative output of the operators in series is greater than that of the operators in parallel.

This analysis still does not fully capture the series vs. parallel effect. The operators in parallel may not all be able to start at the same time due to capacity constraints on the training required by operators performing the job. In this case the cumulative output of the operators in parallel may be even lower.

Therefore ramp-up time and output can be improved without increasing manpower by breaking the longest jobs into shorter jobs. Just as in steady-state production, improvement to the system must be achieved at the bottleneck, in this case the operation with the longest standard time.

The analysis up to this point was based on the assumption that the operations are of similar complexity and therefore the constants of the learning curve are the same. This may not always be the case. If these constants are different for each operation then each operation would require a different number of iterations to ramp-up:

\[ n(i) = (a'(i))(1/b(i)) \]  \hspace{1cm} (2.16)

\[ t_r(i) = s_i \times C_i \]  \hspace{1cm} (2.17)

where: \( C_i = \sum_{j=1,n(i)} a'(i) \times j^{-b(i)} \)

The operation with the maximum value of \( t_r \) would now be the limit to how quickly the system could ramp-up, but it cannot necessarily be said that the length of the ramp-up is proportional to the standard time of this operation. Intuitively, if there is another operation before the longest one with a larger \( a'(i) \) but smaller \( n(i) \) (thus it must have a greater \( b(i) \)), it could conceivably starve the longest operation early in the ramp-up process with the result being that the system will take longer to ramp-up than the limit defined by equation 2.17.

**The Training Process**

All operators who worked on this product had experience assembling the product to be superseded by this new product. However, the new product was sufficiently different that each operator required training in the new product before working on it. The rate \( a: \)
which operators could be brought on-line was limited by the workcenter’s capacity to train. Thus the capacity of the training process determines when each operator can start the learning process. In order to ramp-up as quickly as possible, it is necessary to bring new operators on-line as quickly as training allows. It is important, however, to ensure that training is not short-changed. A poorly-trained operator will likely make more mistakes and may not learn as fast as a well-trained operator.

**Design Performance**

Design performance is defined as the unit’s ability to perform within the required specifications. If the product does not perform as the development team expects, operators may be required to spend time adjusting, troubleshooting and repairing the product. This time is above and beyond the time required to actually build the product. Since the overall processing time is greater than it would otherwise be, the time required to perform enough iterations to reach the standard time and time required to ramp-up also increases. When an operator finds a problem that he cannot solve, the unit may be put on engineering hold while engineers try to solve the problem. If all of the operators have other units to work on, an engineering hold will not affect the overall output of the system. However, if any operator sits idle waiting for the unit to come off engineering hold, a learning opportunity is lost and ramp-up will take longer.

The newer a product is, the more likely it is that there will be design problems. Thus it is likely that design performance is linked to the number of distinctly new technologies being incorporated into the product. As more units are produced, more potential problems should be identified and corrected. It is therefore expected that the amount of time expended doing repair and troubleshooting should decrease as the ramp-up proceeds.

**Parts Availability**

Part shortages create production stoppages. For several reasons described below, these events are more prevalent during ramp-up than during steady-state production.
Cause of Part Shortages

There are three broad categories into which the root cause of a part shortage can be classified: 1) engineering, 2) vendor non-performance, and 3) planning errors.

Engineering change orders to correct design problems or improve the performance, safety, assemblability or cost of the product are frequent during ramp-up. These changes often result in a new or changed part. If there is a serious safety or performance issue or if inventory of a part that is to be replaced is low, an engineering change may be given a near-term effectivity. However, depending on the part, vendors may need eight or more weeks to deliver a new or changed part for the first time. This increases the likelihood of a shortage. Once, a vendor does make delivery of a part for the first time, there is the possibility that it will not work as designed and engineers will have to redesign the part. This would also result in a shortage.

Vendor non-performance covers situations where vendors are unable to deliver a part on time or deliver parts of poor quality. During ramp-up, problems at vendors are exacerbated by the fact that they often are undergoing ramp-ups as well. These manufacturers may have significant tooling and manufacturing process issues to overcome if they are producing a brand new part. These problems could result in failure to deliver on time or poor quality.

Planning errors are those occasions when a part is accidentally ordered late or when inventory information is inaccurate. These are more likely to occur during ramp-up as the materials management function is verifying the accuracy of the information in the MRP system.

Effect of Part Shortages

The concept of the learning curve on ramp-up implies that production output is a function of all of the learning up to that point in time. If a day goes by without any learning, then the possible output in the following day can be no higher than what the possible output was on the day of the shortage. Stated another way, anytime a production
stoppage during ramp-up results in a halt in learning as well, the entire future output curve is pushed out by the length of the production stoppage and the length of time required to ramp-up is thus increased by the same length of time. Figure 2.2 shows this effect using the scenario shown in Figure 2.1b. A production stoppage in week 2 results in a loss of the 24 units expected to be delivered in week 2, but also causes week 3 to fall 6 units short of schedule, week 4 to fall 2 units short of schedule etc. The total production lost is 40 units, the long-term steady-state rate.

![Figure 2.2: Effect of Part Shortage](image)

Main line assembly operations are dependent on upstream operations. Thus a production stoppage and lost learning opportunities can occur at one workcenter as a result of a production stoppage at an upstream workcenter. If an assembly operation is predisposed to part shortages, as it may be during ramp-up, then it is beneficial to reduce the overall dependence of operations on other operations. This can be done by performing as much assembly as possible in subassembly areas which are not dependent (or at least less dependent) on other operations. If a few days worth of completed subassemblies are queued in front of each main line workcenter, then only a lengthy production stoppage in
one of the subassembly areas will cause a production stoppage along the main assembly line.

Shifting as much production to subassembly areas as possible will also reduce the total standard time of the main line assembly operations. The advantages of this are two-fold. First, it will likely allow a reduction in the standard time of the longest single main-line operation which, as earlier analysis showed, is the primary driver of ramp-up time. Second, the overall product lead time will be reduced allowing the system to be more responsive to the customer throughout the life of the product.

Design Stability

Design stability is meant to capture the frequency with which changes are made to the design. Design stability is surely linked to design performance. The more problems there are with the design, the more changes can be expected. Engineering change orders (ECOs) are frequent during ramp-up and even when written to improve the assembly process can be disruptive (Adler and Clark 1991). Just as operators follow a learning curve in terms of their ability to perform an operation, engineers can be expected to follow a learning curve which captures their ability to get the design perfected. With each unit that goes through production, engineers should learn something new. As with design performance the newness of the product likely has a significant effect on the number of engineering changes generated.

The most apparent problem created by ECOs, and the one on which this paper concentrates, is its effect on part shortages. However, there are several other areas where engineering changes can hinder production. Engineering changes may require retrofitting of units already produced or in production. These require operators to stop building new

\[10\text{This practice violates the principles of just-in-time. However, the system is stressed enough during ramp-up and likely cannot handle the additional stress of trying to perform JIT. Once vendor deliveries are stable and parts changes are infrequent, the system will be ready for JIT.}\]
units and update units which they have already built. Creation of a new part may result in a new assembly process for which operators must embark on a new learning curve. If the new part or process has not been fully tested and validated, even poorer design performance may result in the short term requiring even more repair and troubleshooting time.
III. ANALYSIS OF FACTORS FOR THIS PRODUCT

This section uses the factors discussed in Section II to analyze the ramp-up of the product studied here. In practice, a long queue developed at the checkout operation which was a major bottleneck. Not only was it the operation with the longest standard time, it proved to have the flattest learning curve and it experienced far more repair and troubleshooting and engineering hold time than any other operation. As a result of the bottleneck in checkout, the rate of output of the assembly operations was intentionally kept lower than possible to prevent an even bigger queue from developing in front of the checkout operation. Although queues did develop at times in front of other workcenters due to part shortages, they were quickly worked down when the needed parts became available.

Means of Data Collection

Data collection on processing times at two assembly workcenters started with approximately the 50th unit of production and was collected for well over 300 units of production. Data collection at the checkout workcenter started with the 31st unit of production and was halted after the 123rd unit of production.\(^{11}\) Although information on processing times was not collected for the initial units of production, and for some of the units in these ranges, in most cases it was possible to use another source of information to at least determine which operator had performed each operation.

Data gathering was done by operators using a sheet similar to the one shown in Exhibit 4. Operators were instructed to label each half-hour in which the unit was in their area with one of the following colors representing the activity listed: Green, building according to procedures including time spent while being trained; Yellow, troubleshooting,

\(^{11}\) An opportunity for collection of valuable information was lost both by the failure to collect data on early units and also as a result of the decision to stop collecting production data at this workcenter.
repairing, retrofitting or installing parts which were missing during the original build process; Red, machine on engineering hold; Black, waiting for missing parts to arrive or attending meetings. The operators were then asked to sum the number of hours for each color and to comment on anything unusual about the assembly. There appeared to be some confusion on how to fill out the sheets -- particularly as to when it was appropriate to use the color yellow. However, despite some inconsistencies in the data, it appears in large part to represent the actual build process.

Part shortage information, including the part and days affected and the reason for the shortage, was tracked by the master scheduler for the product. Information on each part including the most recent ECO affecting the part was tracked in Kodak's MRP II system. An engineering group tracked the number of ECOs and the parts affected by week.

Analysis of Important Factors

Table 3.1 tabulates production information for the three workcenters. It is important to note that the units themselves and not the operators are tracked. The numbers in the table represent the percentage of time units spent in each activity at the workcenter. Tracking started when an operator began working on a unit and ended when the operator finished working on a unit. Therefore, any time spent in queues is not included. Black time representing part shortages and meetings was only recorded for a unit on which production had started. For example if a part shortage was experienced for one week, and two operators had each started working on one unit, part shortage time would only be recorded for those units. Units in the queue, which are also affected by the part shortage, would not have part shortage time recorded.

<table>
<thead>
<tr>
<th>Assembly</th>
<th>Units tracked</th>
<th>Percent of Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assembly #5</td>
<td>399</td>
<td>Green 72</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Yellow 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Red 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Black 22</td>
</tr>
<tr>
<td>Assembly #6</td>
<td>339</td>
<td>Yellow 59</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Red 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Black 36</td>
</tr>
<tr>
<td>Checkout</td>
<td>68</td>
<td>Green 46</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Yellow 27</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Red 21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Black 6</td>
</tr>
</tbody>
</table>
Several observations were made. As expected assembly time represents a substantial percentage of time for all operations. Engineering hold time and repair and troubleshooting time are significant for the checkout operation but not the assembly operations. This result is understandable since the checkout operation is where unit functionality is verified and thus, where problems are investigated and repaired. Part shortages are significant for the assembly operations. A substantial percentage of time was recorded as black for these operations. Since the checkout operation did not involve installation of any new parts, all of the time recorded as black there must have been meeting time.

The Learning Process

Figures 3.1 and 3.2 plot the assembly plus repair time vs. operator iteration for each unit tracked through the assembly operations. A linear least-squares regression of the natural logarithm of assembly plus repair time vs. the natural logarithm of operator iteration was performed using DataDesk. The results of the regressions converted to a linear scale were as follows:12

\[ \text{Assembly \#5: \quad \text{assembly + repair time} = 3.7 \times \text{standard} \times \text{iteration}^{-0.25} } \]  
\[ \text{Assembly \#6: \quad \text{assembly + repair time} = 3.6 \times \text{standard} \times \text{iteration}^{-0.30} } \]  

This is equivalent to an 85% learning curve for assembly \#5 and an 81% curve for assembly \#6. In either case the first unit would be completed in about 3.6 to 3.7 times the standard time. Based on the analysis in the previous section, it would take 187 iterations for operators of Assembly \#5 to ramp-up and 72 iterations for operators of Assembly \#6 to

---

12 Regressing assembly plus repair time as opposed to just assembly time gave a little better average $R^2$. For Assembly \#5 the $R^2$ was 49.5 (vs. 47.6 just regressing assembly time). For Assembly \#6 the $R^2$ was 44.2 (vs. 44.7 just regressing assembly time). Actual assembly times are disguised to protect confidential information.
The two assembly operations tracked were longer and slightly more complex than the average assembly operation. Since operators in these two workcenters were not pressed for output due to the queue in front of the checkout operation immediately following these operations, learning rates might be better under other circumstances.

Figure 3.1: Assembly and Repair Time vs. Operator Iteration for Assembly #5

---

13 An interesting analysis for future work would be to try to fit a best curve of the shape $\text{time} = \max(a \cdot \text{iteration}^b, \text{standard})$, since this is the shape we have ultimately defined for it. If this were done in this case, the values of $b$ would most likely be larger meaning operators could fully ramp-up in fewer iterations than suggested here.
Figure 3.2: Assembly and Repair Time vs. Operator Iteration for Assembly #6

Figure 3.3 plots the assembly time vs. operator iteration for each unit tracked through the checkout operation. A linear least-squares regression of the natural logarithm of assembly time\(^{14}\) vs. the natural logarithm of operator iteration was performed using DataDesk. The results of the regressions converted to a linear scale were as follows:\(^{15}\)

\[
\text{Checkout: } \text{assembly time} = 3.9 \times \text{standard} \times \text{iteration}^{-0.96} \quad (3.3)
\]

Similar to the assembly operations, the first iteration would be completed in 3.9 times the standard time, but in sharp contrast to the assembly operations, the slope is only 94%.

Unlike all other operations, the checkout operation is entirely made up of performance

---

\(^{14}\)For the checkout operation, "assembly time" refers to the time spent performing the written checkout procedure.

\(^{15}\)The result of the regression gives an \(R^2\) of only 3.3%. The t-statistic of iteration was 1.48 with 64 degrees of freedom implying significance at the 90% level of confidence. The strongest correlation for the data was found when regressing assembly + repair time vs. operator iteration and overall workcenter iteration. The \(R^2\) of the relationship was 20.7%, but this relationship predicted that the first unit would take 35 times greater than the standard. Since there is no data available at small values of workcenter iteration (the first unit tracked was the 31st unit into the workcenter) I question the validity of extrapolating the results of this regression to low values of workcenter iteration.
verification and adjustments, making it a more complex operation. Learning of more complex operations likely proceeds more slowly. Nevertheless, this learning curve (or the standard time associated with it) is inconceivable as it would take over one million iterations for operators to reach the standard time. There must be unmodeled factors affecting the learning curve constants. Since much data was not recorded at the beginning of production, it is hard to identify what those factors are and what their relationship to the learning curve might be.

Figure 3.3: Assembly Time vs. Operator Iteration for Checkout

The Training Process

A rudimentary assembly process with several assembly modules was designed as part of the product development process. For each assembly module established, a highly-skilled operator was assigned to work with the design team. This operator helped the development team build engineering models and performed the witness assembly during which actual assembly procedures were finalized. This operator then performed all of the
initial assemblies and worked any bugs out of the process. He was then given the responsibility of training others when an assembly supervisor decided that he was capable of doing so.

There was no written training procedure for the assembly operations and no data on the training process itself, so actual practice was determined through interviews with assembly supervisors and lead operators. Training typically involved three phases. First, the trainee watched the trainer perform the operation while following along with the assembly procedure. Second, the trainer actively taught the process while the trainee performed the operation. Third, the trainer stayed nearby making sure that the trainee did not make any mistakes and being available to answer questions. During the third phase, the trainer usually spent some time doing something else. After the third phase was completed, the trainee was considered trained. At the discretion of the assembly supervisor, the trainer sometimes continued to inspect the newly-trained operator's finished product for some time. An operator who was training always had building priority if there was not enough work to keep the whole workcenter busy.

In the assembly operations, the initial operator typically performed twenty to forty iterations before training others. Training typically required the trainee to watch the trainer perform one iteration, perform two or three iterations with the trainer giving instruction, and then three to five more iterations while the trainee frequently required the trainer's help. The final phase required about 50% of the trainer's time. Because of the bottleneck at the checkout operation, operators were not brought on line as quickly as possible. Trainers usually returned to building rather than immediately training another operator. This allowed other operators who were eventually to be trained in assembly of the new product, to continue to build the old product. This was necessary as long as output of the new product was lower than desired. In each workcenter one person did almost all of the training.
There was a formal training process for the checkout operation. All training was done in a separate room away from the assembly floor, but the units processed there were still considered production output. Because more operators had to be brought on line in checkout than in the other operations, the first operator did only the first several iterations before starting to train other operators in groups of two to five. Training involved watching videotapes and explaining machine functionality for about three days before hands on training started. Trainees usually performed only two iterations before moving to the assembly floor.

**Design Performance**

Design performance itself is hard to quantify, but the effect of poor design performance on the assembly floor can be illustrated by tracking two things, the repair and troubleshooting time spent by operators and the amount of time units spend on engineering hold.

**Repair and Troubleshooting**

Figure 3.4 shows the trend in the amount of time spent by operators in the checkout operation doing repair and troubleshooting. Although poor design performance cannot necessarily be blamed for all of the problems necessitating repair and troubleshooting (other possible causes include bad parts and operator error), it is likely that a substantial portion of it can be categorized as being a result of poor design performance. Every unit tracked through checkout experienced some repair and troubleshooting time. The average amount of time spent doing repair and troubleshooting was 67\(^{16}\) and the standard deviation was 54, while the range was 2 to 342. As might be expected, the amount of repair and troubleshooting time shows a downward trend.

---

\(^{16}\)Like the assembly time information, actual times have been disguised. The information provided here is done for the basis of relative comparison of the mean and standard deviation to the assembly time.
A multiple linear regression performed on the data resulted in the equation\(^\text{17}\):

\[
time = 4.3 \times s - 0.017 \times s \times wc - 0.16 \times s \times op
\]

where: \(s\) = standard time of the operation
\(wc\) = workcenter iteration
\(op\) = operator iteration

For the assembly operations, the amount of time doing repair and troubleshooting was insignificant.

**Engineering Hold**

The other means by which design performance affects ramp-up production is through engineering holds. Figure 3.5 shows the trend of time that units spent on engineering hold. Of the 67 units tracked through the checkout operation, 28 or 42\% were

\(^{17}\)The \(R^2\) of the regression was 13.5 with 63 degrees of freedom. The \(t\)-ratio of the workcenter iteration was -2.19 while the \(t\)-ratio of operator iteration was -1.83.
put on engineering hold at least once. The average length of time on hold for those units that were on hold was 123 and the standard deviation was 201. The actual range was 2 to 927. The percentage of units that went on hold decreased with cumulative production.

![Figure 3.5: Length of Engineering Holds](image)

A multiple linear regression performed on the data resulted in the equation:

\[
\text{engineering hold time} = 4.8 \times s - 0.039 \times s \times \text{wci}
\]  
(3.5)

For the assembly operations, the amount of time spent on engineering hold was insignificant.

**Parts Availability**

This product included 1100 new parts which were unique to this product and many others which were used on other products within the division. The assembly department

---

18Unit at 450 on this graph was actually at 927.
19The R² of the regression was 5.6%. The t-ratio of the workcenter iteration was -1.96 with 65 degrees of freedom implying significance at close to 97.5% confidence.
experienced many part shortages. Here "part shortage event" is defined as the shortage of one part. A "part shortage day" is defined as the shortage of one part for one day. Thus each part shortage event had one or more part shortage days associated with it.

For the purposes of this study, some approximations were made to the way that part shortages are normally tracked at the division: (1) If two parts were short for the same reason and affected the same subassembly (e.g. a bracket and connector were both short due to an engineering change), it was considered a single part shortage event with the justification that the net effect of this is the same as just missing one of the parts; (2) If a part was short for a second time within 10 working days of a previous shortage of the same part it was considered a single part shortage event under the assumption that the vendor was only able to make a partial delivery for the same reason that he may have missed the original delivery.

Part shortage data for the 14th to 33rd weeks of production (96 days of production) was collected both in the department where final assembly and major subassemblies were performed and in another department where many minor subassemblies for the product were performed. Figure 3.6 shows the distribution of the lengths of the 125 part shortage events recorded. Both the average and the standard deviation of this distribution was 4.0 days. In total there 5.3 part shortage days/production day or 0.0048 part shortage days per production day per new deliverable. This number is 30 times greater than the 0.00015 part shortage days per production day per unique deliverable experienced by the division's mature products during the period from January to July, 1991. The mature products averaged 4.3 days per part shortage.20

---

20Some personnel have questioned the accuracy of the information tracking process used to determine this number, suggesting that the occurrence of all line-outs are not duly recorded.
Engineering was considered the primary cause of a part shortage for the following cases: (1) Any part shortage (regardless of any other reason) occurring within 60 days of the date that an engineering change was entered into the MRP system; (2) Any other part that was not available due to an engineering change; (3) Any part delivered to specification which did not work properly; or (4) A missed delivery because engineering did not release a hold on a tool at a supplier. Vendor non-performance was considered the root cause of a part shortage for the following cases: (1) Failure to deliver a part on time or (2) Delivery of a part that did not specifications. Planning was considered the root cause for the following cases: (1) Part was ordered late or (2) Inventory count was inaccurate.

Engineering was the primary root cause of part shortages, being responsible for 66% of the part shortage events and 73% of the part shortage days. Of the part shortage events attributed to engineering, 76% were on parts that had an engineering change within 60 days of the part shortage. Vendor non-performance was responsible for 26% of the
events and planning errors for 8%. For mature products, these values were engineering
11%, vendor non-performance 49% and planning errors 33%, with 7% other or unknown.

Each part was classified by Kodak as an A, B or C part. The intention is to have A
parts delivered every 5 days, B parts every 20 days and C parts every 60 days. There were
265 new parts classified as A parts, 358 classified as B parts and the rest were classified as
C parts. There were 66 A-part shortages (.25/Part) averaging 4.8 days, 29 B-part
shortages (.081/Part) averaging 3.1 days and 30 C-part shortages (approximately .06/Part)
averaging 3.3 days. This result is not surprising since A parts are the parts most likely to
be custom designed. Custom designed parts may require a vendor to invest in new tooling
and ramp-up a process at their factory. They are the parts most likely to be changed and
most likely to cause problems for vendors when changed.

Design Stability

There were many engineering changes in the early life of this product. The number
of engineering changes were tracked and measured against cumulative production. Figure
3.7 plots the number of ECOs written/unit vs. cumulative production. As can be seen in
the figure, approximately 180 ECOs were written between the time production started and
the first unit was shipped. This number dropped sharply. The data was transformed to
logarithms and a regression was performed on the transformed data. The result of the
regression converted back to a linear scale was as follows\(^21\):

\[
\text{ECOs/unit} = 140 \times (\text{cumulative production})^{-1.0}
\]  
(3.6)

\(^{21}\text{The R}^2\text{ of the regression was 97.6 with 6 degrees of freedom. The F-ratio was 242 and the t-ratio of the correlation was -15.6 which is significant at the 99.5\% confidence level.}\)
As discussed earlier, engineering changes increase the likelihood of part shortages. From the twelfth to forty-third week, 75% of the engineering change orders had effectivities within 60 days and those changes affected an average of 3.9 parts. Over one eleven week stretch, when full information on both engineering changes and part shortages for the next sixty days are available, there were 139 engineering changes with effectivities less than 60 days, and 30 (22%) of them resulted in 34 distinct part shortage events. In all, 62 line-outs that occurred within 60 days of the most recent ECO were identified. Figure 3.8 shows a histogram of the length of time between the engineering change and the part shortage.

22This relationship was determined by breaking production into periods, averaging the total engineering changes over the total production for the period and plotting that figure against the mid-point of cumulative production for the period.
From the beginning of production through the 35th week, there were 180 different retrofits required. On average 9 units were affected. The time spent doing retrofits was not tracked.

23 "Zero" means that the line-out occurred during the same week as the ECO while "1" means it occurred in the next week. Therefore if an ECO was written on a Friday and the part shortage was on Monday, then the event was recorded in week 1. When using this information in the model described later, an approximation was made to closer approximate how many events occurred in the first 40 hours and the next 40 hours etc.
MODEL

A Fortran program (Appendix) was written to simulate the production ramp-up.\textsuperscript{24} The program is a continuous simulation program which calculates the progress of each operator and workcenter each hour. The program constantly tracks the rate at which each operator is working on the unit currently in process or, if no unit is currently in process, the rate at which the next unit will be completed. Also, the unfinished portion of the unit each operator has in process is constantly tracked as well as the number of units completed by each operator. Each cycle, the program calculates whether the operator finishes a unit during the period and if the operator starts working on a new unit that period. If the operator finishes a unit during the period, it is sent to the queue in front of the next workcenter. In order for an operator to start working on a unit, there must be a unit waiting in the queue in front of the workcenter. Following is how each of the factors measured in the previous section were taken into account in the program. Throughout this section, "I" represents an index for workcenters and "J" represents an index for operators.

The Learning Process

Based on observation, operator assembly times follow a learning curve until the standard time of the operation is reached at which point there is no further improvement in assembly time. Thus assembly time was modeled as:

\[ LCTIM = ST(I) \times \text{maximum} \left( 1, \frac{LCE(I)}{IT(I,J)} \right) \]  \hspace{1cm} (4.1)

where LCM represents the multiplier of the learning curve, IT(I,J) represents the operator iteration and LCE(I) represents the exponent of the learning curve for the workcenter. The

\textsuperscript{24}An initial attempt was made to simulate this problem with I-think, a Macintosh-based continuous simulation package. Although a powerful tool, it could not easily simulate training as desired, was generally inflexible and required relatively much longer to simulate the ramp-up than the Fortran program. Because there is a single product and a single flow to simulate, much of the strength of a program like I-think is not required. Witness, a PC-based package, had similar limitations.
actual results showed a marked difference in the learning curve slope for assembly vs. checking operations. Therefore, let LCEB represent the learning curve exponent for a building operation and LCEC represent the learning curve for a checking operation. The job at each workcenter then represents a certain percentage of building represented by PERBLD(I). Therefore, let the learning curve exponent for each workcenter be:

$$LCE(I) = PERBLD(I) \times LCEB + (1 - PERBLD(I)) \times LCEC$$  \hspace{1cm} (4.2)

The Training Process

At each workcenter, initially one operator starts building. The number of operators who are to become trainers is determined (NOT). When each operator who is to become a trainer has performed a certain number of iterations (TITER), he starts training the other operators in groups of a certain size (T AoT). Training for this group continues until all operators in the group have completed a certain number of iterations (IT T). When each member of the group has finished training, the trainer starts training a new group. The total number of operators (NOO) to be trained at a workcenter is determined by the desired long-term daily production rate (PROD) and the standard time of the operation at that workcenter (ST(I)) in hours:

$$NOO(I) = \text{next highest integer} \left( \frac{\text{PROD}}{8.0/\text{ST(I)}} \right)$$  \hspace{1cm} (4.3)

Design Performance

The overall design performance is largely a function of the number of new technologies NEWTEC included in the product. Each new technology initially creates an expected amount of repair time per unit per hour of standard time, INRTTC. Since repair time is significant only for checking and troubleshooting operations, let WRTF(I) represent an initial expected repair time factor for each workcenter. Therefore:

$$WRTF(I) = (1 - PERBLD(I)) \times INRTTC \times NEWTEC \times ST(I)$$  \hspace{1cm} (4.4)

Based on empirical data, the expected length of repair time decreases with both cumulative production of the workcenter and of the operator:
ERTIM(I) = \text{maximum}(0, \text{WRTF(I)} \ast (1 - \text{WCRTFC} \ast \text{WCIT(I)} - \text{OPRTFC} \ast \text{IT(I,J)})) \quad \text{(4.5)}

where \text{WCIT(I)} represents the workcenter iteration and \text{WCRTFC} and \text{OPRTFC} represent constants.

The actual repair time \text{RTIM} is a random variable. Model it as an exponential random variable with expected value \text{ERTIM}. The rate at which an operator works on a unit is the reciprocal of the sum of the assembly time and the repair time:

\text{RATE(I,J)} = \frac{1}{(\text{LCTIM} + \text{RTIM})} \quad \text{(4.6)}

The design performance also affects the occurrence of engineering holds. While a unit is on hold, an operator cannot work on it. On the other hand it is not taking up operator time either. Therefore, model the engineering hold as a hold in the queue before production starts. Like repair time, the number of new technologies likely affects the length of an engineering hold. Each new technology initially creates an expected amount of engineering hold time per unit per hour of standard time, \text{INEHTC}. Engineering holds are only significant for workcenters which involve checking and troubleshooting. Therefore a workcenter engineering hold factor is determined which represents the initial expected length of a hold at each workcenter:

\text{WEHF(I)} = (1 - \text{PERBLD(I)}) \ast \text{INEHTC} \ast \text{NEWTEC} \ast \text{ST(I)} \quad \text{(4.6)}

Based on empirical data, the expected length of an engineering hold decreases with cumulative production. Therefore model the expected engineering hold time as:

\text{ELEH} = \text{maximum}(0, \text{WEHF(I)} \ast (1 - \text{WCEHFC} \ast \text{WCIT(I)})) \quad \text{(4.7)}

where \text{WCEHFC} represents a constant.

The actual length of an engineering hold is a random variable. Model it as an exponential random variable with expected value \text{ELEH}.
Parts Availability

Based on the close relationship between engineering changes and part shortages, part shortages were modeled as following engineering changes. Each engineering change has a certain probability of creating a part shortage, PSECO. Engineering changes are more likely to cause a shortage soon after the change with a diminishing probability up to nine weeks. Let's model this diminishing probability linearly. Therefore the expected number of part shortages in week \( W \) was modeled as:

\[
E(PS(W)) = PSECO \times \sum_{(n=w-8,w)} ECO(n) \times (.20 + .0222 \times (n - W)) \quad (4.8)
\]

where \( ECO(n) \) represents the number of ECOs in week \( n \).

Therefore the probability that any part starts a shortage any hour in a given 40-hour week is:

\[
PPS = \frac{E(PS(W))}{TOTNOP/40} \quad (4.9)
\]

where TOTNOP represents the total number of new parts in the product.

Let the number of parts installed at each workcenter be NOP(I). Therefore the probability of a part shortage starting at any workcenter in a given hour is:

\[
PPSWC = 1 - (1 - PPS)^{NOP(I)} \quad (4.10)
\]

where NOP(I) represents the number of parts installed at the workcenter.

The average length of a part shortage is LPS. Model the length of the shortage as a random variable with expected value of LPS.

Part shortages were modeled such that an operator could not begin production of a unit in the presence of a part shortage. Since the length is being modeled as a random variable, it is possible to have quite long part shortages in the simulation. In reality, the production operation would find an alternate process for obtaining parts if faced with a very long shortage. Therefore, the model overpredicts the effect of shortages.

\[25\text{Note: } \sum_{(n=-8,0)} (.20+.0222\times n) = 1\]
Design Stability

The primary effect of design stability appears to be its effect on part shortages, which was modeled above. Here, we model the number of engineering changes. The number of new technologies in the product once again affects the frequency of engineering changes. Let INECTC represent the expected number of engineering changes per new technology to be released during production of the first unit. Empirical evidence suggests that the expected number of engineering changes released per cumulative unit of production decreases logarithmically. Therefore, model engineering changes as a function of cumulative production. Let the number of ECOs caused by each unit be as follows:

\[ \#\text{ECO/unit} = \text{NEWTEC} \times \text{INECTC} \times (\text{cumulative production})^{-1.0} \quad (4.11) \]

This represents the number of ECOs released during production of that product as opposed to after production. Therefore, at the beginning of production, release all ECOs associated with the first unit of production. When the first unit is finished, release all ECOs associated with the second unit of production etc.

Model Performance

The simulation runs quickly. If little information is printed to the computer screen, two years worth of simulation can be performed in as little as one minute. Thus many iterations of the simulation can be performed.
V APPLICATION OF MODEL TO THIS PRODUCT

Parameters

The Learning Process

Based on the average multiplier calculated for the data collected from the three workcenters, LCM was set at 3.7. The average learning curve exponent of the assembly operations was 0.275. The exponent for a checking operation was set at 0.096 based on data collected for the checkout operation. Since almost all of the functional testing is done in the checkout operation, PERBLD(I) was set at 0 for the checkout operation and 1 for all other operations.

The Training Process

Training practices varied at each workcenter. Because the checkout operation was the bottleneck, operators were brought on-line there more quickly than other workcenters on a cumulative production basis. For the purpose of this simulation therefore, training was simulated following the practices of the checkout workcenter. The effect of this is that the simulation will predict faster than actual production at the workcenters which brought operators on more slowly. However, since the bottleneck is properly modeled, this preserves the validity of the throughput measurement. Thus for the simulation, the following values were set: NOT 1, TITER 2, TAOT 2 and ITT 2. This likely overpredicts the rate at which operators can reasonably be brought on line resulting in a likely overprediction of output.

Design Performance

There were 20 new technologies in this product, therefore NEWTEC was set to 20. Based on the regression of the actual repair times, INRTTC is 0.22. Also based on that regression, WCRTFC is 0.0040 and OPRTFC is 0.037. Based on a regression of engineering hold times, INEHTC is 0.24 and WCEHFC is 0.0081.
Parts Availability

Based on empirical data, 75% of ECOs had effectivities within 60 days. Over an eleven week stretch, there were 34 part shortage events associated with 139 ECOs with effectivities within 60 days (24.5%) and 50.2% of the part shortages were associated with the root cause of following an engineering change. Therefore, the number of part shortages per ECO was calculated as:

\[ P_{E\text{CO}} = 0.75 \times 0.245 / 0.502 = 0.37 \]  \hspace{1cm} (5.1)

Design Stability

Based on a regression of the engineering hold data, INECTC is 7.0.

Validation of Model

One-hundred 24-month simulations were performed using the parameters measured above. Figure 5.1 plots the average result of the simulations vs. the actual production output for the first fourteen months.\textsuperscript{26} Over this period, the average simulated cumulative output was 17% higher than the actual output. Overprediction might be expected, since the model does not account for holidays, meetings and other events which cause an operator to work less than forty hours in a week. Two compensating events did occur. As Kodak realized that the checkout operation was a bottleneck, some checkout operators were put on a fifty-four-hour week and more operators than were required for long-term production rates were brought on-line during the ramp-up to speed the flow of product through the bottleneck checkout operation.

The simulation showed realistic levels of part shortages, engineering holds, engineering changes and repair and troubleshooting time as well. Repair and troubleshooting time and engineering hold time were modeled to eventually go to zero.

\textsuperscript{26} As with the graph in the first section, the first "month" was actually a month and two weeks.
Therefore, these factors have no effect on the modeled output levels in later months of the simulation.

The top of the "monthly output" axis in Figure 5.1 represents the long-term desired output rate. The model shows that throughout the second year of production monthly output will be less than 50% of the desired long-term output rate. This can be attributed to the 94% learning curve imposed on the checkout operation. Completion of ramp-up is dependent on improving the slope of this curve. As product performance is validated, some steps of the checkout operation can likely be eliminated, improving the checkout throughput time. In any case, it is imperative to determine the present slope of the learning curve to determine when the completion of ramp-up might be possible.

The curve in Figure 5.1 represents the average output of the one-hundred simulations. However, even with known levels of uncertainty, actual performance can fall anywhere within a range due to the randomness of the uncertainty. Figure 5.2 shows two simulations along with the average simulation and the actual output. With 95% probability,
output performance will be better than the lowest curve in the figure. With 5% probability, performance will be better than the highest curve. Thus with 90% probability, based strictly on the randomness of uncertainty, the performance will fall between these two curves as the actual performance did. After 14 months the 5% curve was a factor of 1.82 above the 95% curve. Actual performance fell within the statistically expected range at the 82% probability of success level.

In practice, the actual levels of uncertainty would not be known beforehand. Given a range of potential values for each parameter, the range of potential outcomes would be even greater.

![Figure 5.2: Range of Simulations](image)

Analysis of Opportunities for Improvement

Once the ramp-up can be predicted, it is possible to determine the effect of different scenarios. There are two potential changes which should be considered: 1) changes in the configuration of the assembly operation and 2) changes in the values of the parameters. It is within the power of the assembly arena to make the first changes, but changing the
parameter levels is much more of a firm-wide initiative. Indeed for some parameters such as slopes of learning curves, it is unclear what effect any part of the organization may have.

Changes in the Assembly Configuration

The assembly department might consider changing the configuration of the checkout operation. The analysis in section two showed that when two or more operations have the same learning curve, the longest operation is the bottleneck to ramp-up. In this case the longest operation also had the flattest learning curve, guaranteeing that it would be the bottleneck. Therefore, breaking checkout into smaller operations was considered. This is not a novel idea. The production supervisor of Assembly #1 decided that the operation was too long to be learned at one time. Therefore, he broke it down into two smaller operations and had half of his operators learn each operation. It appears to have worked very well.

Effect of Reducing Length of Checkout from Beginning

Dividing the checkout operation into three smaller operations of equal length would make the standard time of each checkout operation shorter than the standard time of the longest assembly operation. The effect that this would have on the slope of the learning curve is unclear, but it is probable that there would be some improvement in its slope since there is less time from the beginning of one operation to the next for operators to "forget" what they learned during the previous iteration. As discussed in the final section, determining the drivers of the learning curve slope is an interesting project for future consideration.

Twenty simulations each were performed for the following scenarios: Scenario 1 - Checkout operation divided into three with the same 94% learning curve; Scenario 2 - Checkout operation divided into three with an improved 89% learning curve. Eighty-nine percent was chosen as halfway between the 94% of the actual longer checkout operation and the 84% of the assembly operations which would be of comparable length to the new shorter checkout operations.
Figure 5.3 shows the average output for each of these scenarios vs. the modeled output in the present configuration. For the first scenario, cumulative output over the first 24 months was 81% higher than the simulated output in the current configuration. For the second scenario, cumulative output was 132% higher than that of the current configuration.

An additional benefit of dividing the checkout operation into smaller operations would be that individual checkout operators would experience faster learning of the problems that can affect the parts of the unit that they are responsible for checking, thus reducing overall repair and troubleshooting. Since the smaller checkout operations would not necessarily have to be performed together at the end of the line, it might be possible to move one or more of the checkout operations between earlier assembly operations in an attempt to catch potential problems as soon as possible after they occur. This would also reduce repair time as parts added later would not have to be removed to get at the bad part. Ultimately the best way to perform a checkout operation is to have each individual assembly operator responsible for verifying that everything installed at his workcenter
works correctly. In this case operators take more pride in their work, and all problems are caught immediately, reducing repair time.

Effect of Reducing Length of Checkout 14 Months into production

In this case production has been going on for some time with the longer checkout operation. Is there value in dividing it into smaller operations now? To answer this question, simulations were run for the scenario where checkout is divided into three after 14 months of production. For the purposes of the simulation, the first, fourth, seventh etc. most experienced operators were assigned to the first of the new checkout operations; the second, fifth, eighth etc. most experienced operators were assigned to the second of the new checkout operation; and the third, sixth, ninth etc. most experienced operators were assigned to the third of the new checkout operations. Once again twenty simulations each were run for scenario 1 (94% learning curve) and scenario 2 (89% learning curve). The results are shown in Figure 5.4. For scenario 1 there is a dip in production in month 15, largely as a result of the fact that the checkout operators did not divide evenly into three and it was necessary to divert resources to training in order to ensure that each of the three operations had enough operators to meet the desired long-term rate of output. Cumulative output over the ten months following the division was 10% higher than that of the current configuration for scenario 1 and 60% higher for scenario 2. In addition, both scenarios will continue to outproduce the current configuration every month for years to come.
Changes in Parameters

Rather than changing the configuration of the assembly operation, it may be possible to improve output by improving some of the parameters associated with the ramp-up. The important factors to change would include the learning curve constants, the probability that any ECO causes a part shortage (as modeled this is the same as changing the number of ECOs), the expected initial repair time per new technology, and the expected engineering hold time per new technology. There are two concerns with attacking the problem from this angle. First, for a parameter such as the slope of the learning curve, it is unclear what to do to improve it. Second, for parameters which suggest obvious improvement opportunities, such as reducing repair time by improving the design, the effort is often out of the assembly department's hands.

One of the best ways to prioritize potential improvement opportunities is to establish factor levels for each parameter of interest and perform an analysis of variance (ANOVA) on the output of the simulations. In order for the results of the ANOVA to be meaningful,
the factor levels must be carefully determined and normalized against some other metric such as dollars of cost to achieve the improved performance. As an example, if it would cost a certain amount of money to reduce the probability of a part shortage in half, then the effect of doing this should be measured against the effect of spending the same amount of money to reduce the amount of repair time. The data does not exist to do this analysis properly. Therefore it was deemed inappropriate to perform an ANOVA under these circumstances. Nevertheless, the general effect of changing parameter levels can be determined. As an example, the effect of reducing part shortages and reducing design problems, which presumably reduces repair time and engineering hold time, was considered both in the current configuration and after changing the assembly configuration.

Effect of Reducing Part Shortages in Current Case

This product experienced many part shortages. Twenty simulations were performed for the case where there were half as many part shortages as actually experienced. The average result is compared to modeled performance with current levels of performance in Figure 5.5. Over the first 24 months, cumulative output would have been 19% higher. This number is likely high because as mentioned earlier the model overpredicts the effects of part shortages.

Effect of Improving Design Performance in Current Case

Design performance was blamed for many of the ramp-up problems. If the design had performed better, repair and troubleshooting time and engineering hold time would have been reduced. It is possible that the learning curve would have been steeper as well, but this is not clear. Twenty simulations were performed for the case where both repair and engineering hold time were reduced by 50%. The average result is compared to modeled performance with current levels of performance in Figure 5.6. Over the first 24 months, cumulative output would have been 29% higher. Thus design performance had a significant effect on output, but cannot be held responsible for missed schedules of the magnitude shown in Figure 1.1. If the cost of reducing part shortages by half and
improving design performance by half were roughly the same, then this analysis shows that there would be a higher payoff to improving design performance.

Figure 5.5: Effect of Reducing Part Shortages in Current Configuration

![Graph showing the effect of reducing part shortages in current configuration. The graph compares monthly output over months with and without part shortages. The line with squares represents the current scenario, while the line with circles represents the scenario with half part shortages. The graph shows a steady increase in output over time.]
**Effect of Reducing Part Shortages under Another Scenario**

What would be the effect of reducing part shortages under the scenarios for reconfiguring the checkout operation above? Twenty simulations were performed with half as many part shortages under scenario 2 (three checkout operations, 89% learning curve). The average result is compared to the performance of scenario 2 with the current number of part shortages in Figure 5.7. In this case, reducing part shortages increases cumulative production over 24 months by 40%. Thus if improvements can be achieved through reconfiguring the assembly operation, then reductions in part shortages will be even more important. Under scenario 2 a long queue does not build up in front of checkout. As a result, production stoppages in upstream workcenters are more likely to starve downstream workcenters.

**Effect of Improving Design Performance under Another Scenario**

The effect of improved design performance under scenario two was also measured. Twenty simulations were performed with half as much repair and engineering hold time
under scenario 2. The average result is compared to the performance of scenario 2 with the current design performance in Figure 5.8. In this case, cumulative production was slightly less with improved design performance. This is a result of the randomness of the simulations. In reality performance would not be poorer. However, the analysis suggests that if improvements are made in other places, the effect of poor design performance will be minimized. Thus if improvements can be achieved through reconfiguring the assembly operation, then the effect of improving design performance is greatly reduced. In this case effort should be expended in reducing part shortages.

Figure 5.7: Effect of Reducing Part Shortages in Scenario 2
Best-case Ramp-up

What is the achievable output in the absence of any design or vendor problems which cause uncertainty -- the "best-case" ramp-up? The best-case was simulated for the present configuration and for the two scenarios of dividing the checkout operation into three. The results are shown in Figure 5.9. Given the measured learning curves, even in the absence of any design or vendor problems, the assembly department as presently configured could not achieve full production in two years. For scenario 1, cumulative production over the first 24 months would be 1.37 times greater than that of the current configuration. For scenario 2, cumulative production over that time would be 2.25 times greater than that of the current configuration.
When to Start the Ramp-Up

Kodak managers asked whether they started the ramp-up at the right time. In order to answer this question definitively, it is necessary to know both what the expected parameter levels would have been at other times and what the effect of the potential delay in product availability to the market would be. As discussed in the first section, a delay in product availability results in the best case in a revenue delay, thus lowering its net present value, and in the worst case in revenue lost forever. When a product is entering a competitive market, as this one was, it is most likely better to start production earlier rather than later.

Production went on for some time, with constant design changes, before shipping approval was granted. However, Kodak had internal mechanisms in place to ensure that any product delivered to the customer was of high quality in order to avoid damage to Kodak's worldwide quality reputation. The product was not available in sufficient quantities on the shipping approval date as a result of the lower than expected ramp-up
performance. If production had not started as early as it did, the situation would have been worse. Although the cost of producing units of such poor quality that they can never be used or shipped to customers must be factored into any decision, the cost may be justified by the learning which will enable shipment in large quantities once shipping approval is obtained.
VI CONCLUSION, RECOMMENDATIONS AND FUTURE WORK

Conclusion

Completion of ramp-up in a manual assembly environment is dependent on each assembly operator achieving the proficiency of performing his operation in the standard time. Many factors can interfere with their ability to do this quickly. There was evidence of a strong learning curve relationship in the two assembly workstations studied. The evidence in the checkout workstation was less compelling. There were unmodeled factors which were having an effect on the slope of the learning curve there.

The firm's capacity to train new assembly operators defines the rate at which new operators can be brought on line to start training. The later the last operator comes on line, the longer the ramp-up will be. Although it is desirous to complete training quickly, it must be done well. Assembly operators received a substantial amount of training as a result of the fact that those workcenters had more time to do training. Because checkout was a bottleneck, the operators in that workcenter received training for only two iterations. It is quite possible that this was a significant factor in the slow learning observed at that workcenter.

Design problems lead to repair and troubleshooting time for operators and engineering hold time for the units of production. Units spent a substantial amount of time in the checkout workcenter. This again is a significant factor in the slow learning observed at that workcenter.

In summary, a key attribute that determines which types of assembly environments would be well served by ramp-up is the amount of time it takes the operators to become proficient in performing their work. This time is influenced by the amount of training time and the time operators spend troubleshooting.

...
Part shortages are more prevalent during ramp-up than during steady-state production. If learning is dependent on production, any time that an operator spends not working due to a part shortage increases the ramp-up by the length of the production delay. Not only will that day’s schedule fall behind, but every day’s schedule until ramp-up is completed will be adversely affected. Because the bottleneck checkout operation did not involve installation of parts, the effect of part shortages in this case was not great. Any workcenters shut down by part shortages quickly caught up to checkout when the shortage was over. However, if a means to improve the flow through checkout is found, part shortages could easily become the factor most limiting ramp-up.

There are a couple of potential ways to reduce the effect of part shortages. Providing training units which operators can tear down and rebuild while waiting out part shortages can reduce the dependence of training on production. If one operation is down due to a part shortage, it is possible that downstream operations will be starved resulting in lost learning opportunities there as well. Installing as many parts as possible in off-line subassembly areas can reduce the dependence of assembly operations on each other and improve the overall lead time for the product both during ramp-up and during steady-state production. Kodak has done a good job of taking advantage of subassembly areas.

Engineering changes have an adverse effect on most of the other issues described here, but most obviously on part shortages. Twenty-two percent of the engineering changes during one stretch of the ramp-up were closely followed by a part shortage of at least one of the parts changed. There was a very interesting relationship suggesting that engineering changes per unit of production are inversely proportional to cumulative production, so there is learning associated with the engineering task as well.

The Fortran program modeling these factors accurately modeled the ramp-up of this product. It is hoped that it accurately models any new product in a manual production environment, but further validation of this is necessary. The primary benefits of using this model are: 1) more accurate forecasts of ramp-up and 2) improved ramp-up performance.
ult of better management of the ramp-ups. The better forecasting of ramp-ups will
help firms plan more accurately with regard to budgets, product introductions and
marketing plans for other products. The better management of ramp-ups will help firms get
their products to market more quickly.

Recommendations

This research suggests that the following courses of action can improve the overall
performance of a production ramp-up:

(1) Reduce the length of the longest job

A manual assembly system cannot achieve full production until each operator can
perform his operation in the standard time. Operators following a learning curve will reach
this level of proficiency in a given number of iterations of that job. Thus the longer the job,
the more time it takes to perform that given number. The shorter the job, the less time it
takes. Operations should be divided with the goal of reducing the length of the longest
operation as much as possible. Dividing a known job into smaller jobs may also have the
desired effect of actually reducing the number of iterations required to reach proficiency,
further reducing the time to ramp-up.

The scenarios presented in this paper suggested dividing the checkout operation into
three, but even more substantial changes could be made. The total number of operators
required to achieve a steady-state production rate is the desired output per hour times the
number of standard hours per unit. In theory, the fastest ramp-up would occur if all
operators were put in series and given jobs of equal length. Such a system would be
inflexible in that a single absent operator would affect the whole line. However, in this
case the division is a long way away from worrying about this. In fact one of the
production supervisors actually divided the longest assembly job into two smaller jobs in
order to ramp the operation up more quickly.
(2) *Decouple learning from production by keeping early units of production as assembly training models.*

Part shortages (and other events) can create production stoppages. If learning is dependent on production, a production stoppage at any time during a ramp-up results in cumulative lost production at the desired steady-state rate of output and pushes the total ramp-up time out by the length of the stoppage. If learning is independent of production, a production stoppage will only result in lost output at the current rate of output. Therefore, it is necessary to decouple learning from production in order to prevent production stoppages from turning into lost learning opportunities. This could be achieved by keeping the early units of production in the production area and using them as training machines which operators can tear down and reassemble when there is a halt in production. In practice all of the initial units of production are sent to groups internal to Kodak such as marketing, service etc., so that they can be trained in the use and function of the product. Some units should be set aside for training of production operators as well.

(3) *Install as many parts as possible off of the main assembly line.*

There are three benefits to doing this. First, this would reduce the total standard time along the main line possibly making it easier to reduce the maximum standard time along the main line (see the first recommendation). Second, reducing the total length of the main line operations reduces the overall product lead time which is a benefit both during ramp-up and during steady-state production. Third, operations along the main assembly line are dependent on upstream processes for production. A stoppage in one workcenter can result in stoppages in downstream workcenters which results in lost learning opportunities there as well. Subassembly operations are not dependent on other processes. By moving parts to the subassembly operations and planning to have some extra inventory of subassemblies built up in front of the main line workcenters, part shortages of parts installed in subassemblies are less likely to create stoppages along the main assembly line.
(4) Collect data on all aspects of the ramp-up to identify ways of improving it

The drivers of the learning curve in particular are important to identify. In order to do this, it is necessary to collect as much pertinent data as possible. It is not a good idea to turn the production operation into a laboratory, but as production is happening, as much data as possible should be collected.

(5) Explicitly determine the effect of delays in product availability

Ramp-up performance largely affects the availability of the product. When making trade-offs concerning the ramp-up, it is vitally important to know with some accuracy what the effect of delays in product availability might be.

Opportunities for Future Work

This work has raised several issues which warrant further study.

Validity of Model for Other Ramp-ups

How well does this model predict ramp-up performance in other manual assembly operations? How well does it model ramp-ups in environments other than manual assembly operations? Exhibit 3 shows many factors which may potentially affect a ramp-up. This analysis modeled only the factors which appeared important and could be measured in this particular ramp-up. Future work should attempt to apply this framework to other products in an attempt to verify its validity in other cases and to identify other factors which should be explicitly modeled.

Drivers of Learning Curve

Based on limited data, this work has assumed that the nature of an operation (building vs. checking) is a key factor in determining the shape of a learning curve. It has also been suggested here that the length of an operation may affect this slope. Clearly there are other factors which affect the shape of the learning curve which were not discovered here. The most interesting relationships to research further would be the effect of the amount of time between iterations and the effect of time spent doing repair and
troubleshooting. In order to discover these relationships, careful collection of data from the beginning of production would be necessary. Other potential factors influencing the shape of the learning curve might be the kind and length of training, trainer competence, operator skill level and operator experience.

As this paper has stressed, completion of ramp-up is dependent on operators reaching a certain level of proficiency. The benefit of understanding what drives the speed with which operators learn is the ability to affect it.

*Other Learning Effects*

Learning certainly drives the length of time an operator requires to perform and operation, but are there other effects to learning. It would seem that operators learning a new assembly process would to make more mistakes. Is this true and if so how can the effect be measured or controlled.

*Relative Merits of On- vs. Off-Line Training*

For the assembly processes all training was done on-line by other operators. Since there were no written training procedures, training practices varied from one workcenter to the next. Training for the checkout process was done in a separate room off the assembly floor. Another new product was simultaneously being ramped-up next door. For that product all training was done by full-time trainers in a separate training area. In both cases all product produced during training was used to fill the production schedule and training, like other production, was subject to parts availability. There is no consensus on whether on-line training works better than off-line training and there is no data available to measure the relative effectiveness of each training method.

*Further Defining Design Stability and Relating it to Design Performance*

The stability of the design when production starts is the driving factor in terms of how many engineering changes will be necessary, which in part drives part shortages. The results of this case study suggests that each doubling of output will half the number of
engineering changes per unit. Can this result be generalized to other products, or is it specific to this case? How can we forecast the initial levels of engineering change activity?

Defining design stability, is an elusive thing. One manager\textsuperscript{27} suggested the list of leading indicators for design stability shown in Table 5.1. An area of future work could be to gather this information for several products and relate these factors to the rate of engineering changes, part shortages and the amount of AOC time spent by the operators.

Poor design performance is the factor which leads to engineering changes. Thus low design stability indicates poor design performance. If the design is not stable as production ramp-up begins, there will probably be many problems with poor design performance.

\begin{table}[h]
\begin{center}
\begin{tabular}{|c|}
\hline
1 Problem reports outstanding \\
2 Rate of generation of new problem reports (number/hour of testing) \\
3 Corrective actions outstanding \\
4 Rate of generation of new corrective actions \\
5 Average number of ECOs generated recently \\
6 Trend of ECO generation \\
\hline
\end{tabular}
\end{center}
\caption{Leading Indicators of Design Stability}
\end{table}

\textsuperscript{27}Dr. James C. Minor.
References


Dutton, John M., Anne Thomas, and John E. Butler, "The History of Progress Functions as a Managerial Technology," Business History Review, 58 (No. 2), 204-233.
EXHIBIT 1: SCHEMATIC DEPICTION OF THE PROCESS FROM DESIGN TO STEADY-STATE PRODUCTION
EXHIBIT 2: PROCESS FLOW DIAGRAM WITH RELATIVE STANDARD TIME FOR EACH ASSEMBLY
EXHIBIT 3: CAUSE AND EFFECT DIAGRAM FOR SUCCESSFUL PRODUCTION RAMP-UP

- Training
  - Comprehensive program
    - Hands-on
    - Funded
    - Trainers available
    - Working product available
  - Processes in control
    - BOM correct
    - Performs to spec
  - Supplier ramp-ups smooth
    - Stable
    - No retrofits
    - No unplanned demands
  - Design

- Personnel
  - Skilled
    - Experienced
    - Available
  - Support groups staffed
  - Learning
  - Parts

- Production Area/ Tooling
  - Available on-time
  - Flexible, if necessary
  - Operators like tooling
  - Configured properly
  - Suppliers as partners
  - Quality
  - No changes
  - Available when needed
  - Metrics in place
  - Customer focus
  - Realistic schedule/assumptions
  - Successful Production Ramp-up

- Management
EXHIBIT 4: SAMPLE DATA COLLECTION SHEET

<table>
<thead>
<tr>
<th>DATE</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>COMMENTS</th>
<th>G</th>
<th>Y</th>
<th>R</th>
<th>B</th>
</tr>
</thead>
</table>

RED- ENG. HOLD   YELLOW- WORK AROUND-C/O ONLY
REPAIRS
TROUBLESHOOTING
INSTALLING MISSING PARTS

GREEN- BUILDING
TRAINING

BLACK- NO PARTS
MEETING

TOTAL
Appendix: Fortran Program

C Compile this program with toolbox and no case sensitivity
C
C This program simulates the ramp-up of a manual assembly operation.
C It is a continuous simulation which measures the progress of each
C operator and each factor on an hourly basis. Flow of product to and
C from each workcenter/operator is discretized by calculating that each
C operator either did or did not complete a unit in the period and did
C or did not start working on another unit this period. The logic of
C this program assumes that no operation has a standard time of less
C than one hour.
C
C There are five factors which affect ramp-up that are explicitly
C modeled:
C
C (1) Learning.--Operator build times follow a standard learning curve
C until they reach a standard time after which point there is
C no further improvement. The learning curve has two constants:
C a multiplier, which is the same for each operation and an
C exponent, which varies from operation to operation. The
C multiplier defines how long the first iteration will take,
C while the exponent defines how quickly the operator
C will reach the standard time.
C
C (2) Training.--Initially only a single operator knows the operation
C and can build product. Once an operator has performed a
C certain number of iterations, he is considered proficient and
C stops building to train other operators. Training requires a
C certain number of iterations and proceeds as quickly as
C possible, until all operators have been trained.
C
C (3) Repair Time.--During ramp-up, some operators have to spend a
C significant amount of time doing repair and troubleshooting.
C The expected value of this repair time, should go down as
C more units are produced.
C
C (4) Part Shortages.--During ramp-up, there is a high probability
C that the assembly line will experience part shortages. This
C probability should decrease as more units are produced.
C
C (5) Engineering Holds.--During ramp-up, units will be put on
C engineering hold as a result of design performance. The
C expected length of these engineering holds should decrease as
C more units are produced.

C These statements are necessary to access the Mac's random number
C generator.

GLOBAL DEFINE
INCLUDE "Types.inc"
INCLUDE "QuickDraw.inc"
END

PROGRAM SIMULATION

C Explanation of variables:

72
HMIT = Number of iterations of simulation to perform.
ITS = Number of simulation presently being performed.
NEWTEC = Number of new technologies in product.
TOTNOP = Total number of new deliverables in product, including
those installed in subassemblies.
NOWC = The number of workcenters in the assembly process.
PROD = The desired long-term production rate in units/day.
HPP = Hours of production per week.
MQ = The maximum number of units that will be allowed to build up
in the queue in front of any workcenter.

ST() = Standard time of operation for workcenter.
NOP() = Number of parts installed at workcenter. The sum of all
NOPs will be less than TOTNOP because of subassemblies.
PERBLS() = Percent of operation that is building.

NOO() = Number of operators in each workcenter.
COM() = Number of units completed by the workcenter in last hour.
CUMFIN() = Cumulative number of units finished by workcenter.
WCIT() = Workcenter iteration of unit in process.
SM() = Supply of mainframes in queue in front of workcenter.

TITER = The number of iterations an operator must perform before he
can be a trainer.
ITT = The number of iterations required to train an operator
NOT = The number of operators who will be allowed to become trainers
TAOT = The number of operators a trainer can train at one time.
NTT() = Next operator to be trained at the workcenter.
TRNRO = Number of trainers who have done enough iterations to
start training at the workcenter.
TRAV() = Number of those trainers who are available to train
(i.e. not already training someone else.)

WK = Length of simulation in weeks.
MOS = Length of simulation in months.
LOS = Length of simulation in hours.
WOO() = Weekly output of system.
MOO() = Monthly output of system

LCM = Multiplier constant of learning curve.
LCEB = Learning curve exponent for building operation.
LCEC = Learning curve exponent for checking operation.
LCE() = Exponent constant of learning curve for workcenter.

INRTTC = Expected repair time/new technology for initial unit
WRTF() = Repair time factor for workcenter.
WCFAC = Repair time constant for workcenter iteration.
OPFAC = Repair time constant for operator iteration

RATE(IJ) = Rate of production for unit that operator is now
working, or if not working, the next unit.
TOFINI(IJ) = The uncompleted portion of the unit of production
which the operator currently has in process.
IT(IJ) = Operator iteration for unit that operator is now working,
or if not working, the next unit.
LCTIM = Operator build time as a function of the learning curve.
ERTIM = Expected value of the operator's repair time.
C RTIM = Operator repair time.

C LEH = Expected length of an engineering hold in hours.
C SEHE = Sum of all engineering hold events.
C SEHH = Sum of all engineering hold hours.
C INEHTC = Expected initial engineering hold time per new technology.
C WEHF = Engineering hold factor for the workcenter.
C EHFCUN = Engineering hold time factor for operator iteration.
C EH() = Number of units on hold at each workcenter.

C INEHTC = Expected number of engineering changes per new technology
C for first unit.
C ECOINT = Expected number of ECOs for first unit.
C ECO() = Number of ECOs in week.
C CUMECO = Cumulative number of ECOs.

C LPS = Expected length of a part shortage in hours.
C SPSE = Sum of all part shortage events.
C SPSH = Sum of all part shortage hours.
C PSECO = Expected number of part shortages per ECO.
C PPS = Probability that any part starts a shortage any hour.
C PS() = Number of part shortages at each workcenter.

C W = Index for weeks.
C I = Index for workcenters.
C J = Index for operators.
C T = Index for hours.
C M, N, Y, Z = General indeces.

C AV = Temporary variable which calculates the number of units
C available for the current operator to start.

C TEMP, TEMPI = Temporary variable.

IMPLICIT NONE

INTEGER HMIT, ITS
INTEGER NEWTEC, NOWC, HPP, MQ
INTEGER NOO(11), COMP(11), SM(11)
INTEGER TITER, IIT, NOT, NTT(11), TRNR(11), TRAV(11)
INTEGER WK, MOS, LOS, WO(418), MO(96)
INTEGER SEHE, SEHH, EH(11)
INTEGER SPSE, SPSH, PS(11)
INTEGER W, I, J, T, M, N, Z, Y
INTEGER AV, TEMPI

REAL TOTNOP, PROD
REAL ST(11), NOP(11), PERBLD(11)
REAL CUMFIN(11), WCTT(11)
REAL TAOT
REAL LCM, LCEB, LCEC, LCE(11)
REAL INRRTC, WRTF(11), WCFAC, OPFAC
REAL RATE(11,28), TOFINI(11,28), IT(11,28), LCTIM, ERTIM, RTIM
REAL LEH, INEHTC, WEHF(11), EHFCUN
REAL INEHTC, ECOINT, ECO(-8:0), CUMECO
REAL LPS, PSECO, PPS
REAL TEMP
C GLOBAL PARAMETERS
C Initialize global parameters
NEWTEC = 20
TOTNOP = 1100
PROD = XXX
HPP = 40
MQ = 40

C WORKCENTER PARAMETERS
C Initialize workcenter parameters
NOWC = 9

C XXX
ST(1) = XXX
NOP(1) = XXX
PERBLD(1) = 1.0

C XXX
ST(2) = XXX
NOP(2) = XXX
PERBLD(2) = 1.0

C XXX
ST(3) = XXX
NOP(3) = XXX
PERBLD(3) = 1.0

C XXX
ST(4) = XXX
NOP(4) = XXX
PERBLD(4) = 1.0

C XXX
ST(5) = XXX
NOP(5) = XXX
PERBLD(5) = 1.0

C XXX
ST(6) = XXX
NOP(6) = XXX
PERBLD(6) = 1.0

C XXX
ST(7) = XXX
NOP(7) = XXX
PERBLD(7) = 0.0

C XXX
ST(8) = XXX
NOP(8) = XXX
PERBLD(8) = 1.0

C XXX

ST(9) = XXX
NOP(9) = XXX
PERBLD(9) = 1.0

C Given the desired long-term output rate and the standard time of each
C operation, calculate the number of operators required at each
C workcenter.

DO 3 I = 1, NOWC
    NOO(I) = INT(PROD/(8.0/ST(I))+.99)
3    CONTINUE

C Read number of iterations of simulation to perform and seed random
C numbers.

PRINT*, 'HOW MANY ITERATIONS?'
READ*, HMIT
ITS = 0
PRINT*, 'RESEED RANDOM NUMBERS ENTERING AN INTEGER 0 TO 100'
READ*, TEMPI
DO 4 Y = 1, TEMPI
    TEMP = RANDOM()
4    CONTINUE

C SIMULATION LENGTH
C Read in the desired simulation length in weeks. Determine length
C in hours. The user may end the program by entering 0 weeks of
C simulation.

WK = 106
IF (WK .EQ. 0) GOTO 700
LOS = HPP*WK
MOS = MAX(0, INT((WK-2)/4.333))

C SIMULATION RETURNS TO HERE

C OUTPUT VARIABLES
C Initialize output variables. Simulation starts in week 1.

5    DO 7 M = 1, WK
        WQ(M) = 0
7    CONTINUE
    W = 1
DO 8 Z = 1, MOS
    MO(Z) = 0
8    CONTINUE

C WORKCENTER VARIABLES
C Initialize workcenter variables.
DO 10 I = 1, NOWC
    COMP(I) = 0
    CUMFIN(I) = 0
    WCIT(I) = 1
    SM(I) = 0
10 CONTINUE

C TRAINING PARAMETERS
C Initialize training parameters. Initially the next operator to
C train in each workcenter is operator number 2, and there are no
C trainers.

    TITER = 2
    ITT = 2
    NOT = 1
    TAOT = 2
    DO 20 I = 1, NOWC
        NTT(I) = 2
        TRNR(I) = 0
        TRAV(I) = 0
20 CONTINUE

C LEARNING PARAMETERS
C Initialize or read in learning variables.

    LCM = 3.7
    LCEB = -.275
    LCEC = -.096
    LCEC = -.188
    DO 35 I = 1, NOWC
        LCE(I) = PERBLD(I)*LCEB+(1-PERBLD(I))*LCEC
35 CONTINUE

C REPAIR TIME PARAMETERS
C Read in relationship between repair time and new technologies and
C initialize repair time variables.

    INRTTC = 0.22
    INRTTC = 0.11
    DO 40 I = 1, NOWC
        WRTF(I) = (1-PERBLD(I))*INRTTC*NEWTEC*ST(I)
40 CONTINUE

C OPERATOR VARIABLES
C Given learning curve parameters and repair time parameters, initialize
C operator variables. Initially the first operator will be building at
C a rate determined by the learning curve and repair time. Other
C operators have not been trained and do not initially build.

    DO 45 I = 1, NOWC
        DO 43 J = 1, NOO(I)
            RATE(LJ) = 0
            TOFINI(LJ) = 0
            IT(LJ) = 1
43 CONTINUE
IT(I,1) = 1
LCTIM = ST(I)^*LCM
ERTIM = WRTF(I)^*(1-WCFAC-OPFAC)
RTIM = (-ERTIM)^*LOG(1-ABS(RANDOM())/32768.0)
RATE(I,1) = 1/(LCTIM+RTIM)

CONTINUE

C ENGINEERING HOLD PARAMETERS
C Read in relationship between engineering hold time and new
C technologies and initialize engineering hold time variables.

SEHE = 0
INEHTC = 0.24
C
INEHTC = 0.12
EHFCUN = 0.0081
DO 50 I = 1, NOWC
    WEHF(I) = (1-PERBLD(I))^*INEHTC^*NEWTEC^*ST(I)
    EIH(I) = 0
50 CONTINUE

C ENGINEERING CHANGE ORDER PARAMETERS
C Read in relationship between ECOs and new technologies.
C Initialize engineering change order variables. The ECOs associated
C with the first unit of production are released at the start of
C of production and assigned to ECOINT

INECTC = 7.0
ECOINT = INECTC^*NEWTEC
ECO(0) = ECOINT
CUMECO = ECO(0)
DO 60 M = -8, -1
    ECO(M) = 0
60 CONTINUE

C PART SHORTAGE PARAMETERS
C Read in relationship between part shortages and ECOs and initialize
C part shortage variables.

LPS = 32)*HPP/40
SPSE = 0
SPSH = 0
PSECO = 0.37
C
PSECO = 0.185
PPS = ECO(0)^*PSECO^*(.20)/TOTNOP/HPP
DO 70 I = 1, NOWC
    PS(I) = 0
70 CONTINUE

C SIMULATION STARTS HERE
C Begin simulation, simulate for LOS hours.

DO 500 T = 1, LOS

C Simulate workcenters from last to first because ability to build this
C period is affected by performance of previous workcenter last period.

DO 400 I = NOWC, 1, -1
C Initialize variable

    COMP(I) = 0

C ENGINEERING HOLD
C New units delivered are subject to engineering hold if this is
C a checking operation. If not, new units increase the number
C of units in the queue. For the first workcenter, we will assume
C that the arrival rate is just fast enough to maintain a constant
C level in the queue.

150 IF (I .EQ. 1) THEN
   SM(I) = MQ
   GOTO 165
ENDIF
LEH = MAX(0.0, WEHF(I)*(1-EHFCUN*WCIT(I)))
IF (COMP(I-1) .EQ. 0) THEN
   GOTO 155
ELSEIF (LEH .EQ. 0) THEN
   SM(I) = SM(I) + COMP(I-1)
   GOTO 155
ELSE
   EH(I) = EH(I) + COMP(I-1)
   SEHE = SEHE + COMP(I-1)
ENDIF

C If any units are on hold, check to see that the hold is over.
C If no units are on hold, this section can be skipped.

155 IF (EH(I) .EQ. 0) GOTO 165

C For each unit on hold, check to see if the hold ends this hour. Since
C the length of the holds are an exponential random variable, use the
C property of this random variable that says that the probability of an
C event in any time period is the same regardless of what has happened
C before.

IF (LEH . EQ. 0) THEN
   SM(I) = SM(I) + EH(I)
   EH(I) = 0
   GOTO 165
ENDIF
TEMP = EH(I)
DO 160 M = 1, TEMP
   IF(ABS(RANDOM())/32768.0 .LT. 1/(LEH)) THEN
      EH(I) = EH(I) - 1
      SM(I) = SM(I) + 1
   ENDIF
160 CONTINUE
SEHH = SEHH + EH(I)

C PART SHORTAGES
C For each part short, check to see if the shortage ends this hour.
C Since the length of the shortage are an exponential random variable,
C use the property of this random variable that says that the
C probability of an event in any time period is the same regardless of
C what has happened before. If no parts are short at the workcenter,
C skip the section.
165 IF (PS(I).EQ.0) GOTO 180
    TEMP = PS(I)
    DO 170 M=1,TEMP
        IF(ABS(RANDOM(M))/32768.0.LT.1/(LPS-0.5)) THEN
            PS(I) = PS(I) + 1
        ENDIF
    CONTINUE
    SPSH = SPSH + PS(I)

C Calculate if any new part shortage starts this period.

180    TEMP = ABS(RANDOM(M))/32768.0
    IF(TEMP.LT.1-(1-PPS)**NOP(I)) THEN
        PS(I) = PS(I) + 1
        SPSE = SPSE + 1
        SPSH = SPSH + 1
    ENDIF

C CHECK THAT WORKCENTER HAS UNITS
C If there is a part shortage in effect, no new units can be started.
C Otherwise the limit to the number of units available to start is
C the number of units in the queue.

    IF (PS(I).GT.0) THEN
        AV = 0
    ELSE
        AV = SM(I)
    ENDIF

C CHECK FOR BLOCKS
C If queue for workcenter ahead is full, operators will not work.

    IF (I.NE.NOWC.AND.SM(I+1).GE.MQ) GOTO 400

C TRAINING
C If there are operators who need to start training and trainers
C available, training will start this period.

    IF (NTT(I).GT.NOO(I).OR.TRAV(I).EQ.0) GOTO 240
    TEMP = TRAV(I)
    DO 200 M=1,TEMP
        TRAV(I) = TRAV(I) - 1
        TEMPI = NTT(I)
        DO 195 N = TEMPI,TEMPI+TAOT-1
            LCTIM = ST(I)**LCM
            ERTIM = MAX(0.0,WRTE(I)*(1-WCFAC*WCIT(I)-OPFAC))
            RTIM = (-ERTIM)*LOG(1-ABS(RANDOM(I))/32768.0)
            RATE(I,N) = 1/(LCTIM+RTIM)
            NTT(I) = NTT(I) + 1
            IF (NTT(I).GT.NOO(I)) GOTO 240
    CONTINUE
195    CONTINUE
200    CONTINUE

C CALCULATE PROGRESS OF EACH OPERATOR
C In order to ramp-up as quickly as possible, the least-trained
C operators have building priority if there are not enough units
C available to satisfy all operators' needs. Therefore start
C simulation from least-trained operator who has started building.
C Operators who are training are not building.

240     DO 300 J = NTT(I)-1, 1+TRNR(I)-TRAV(I), -1

C If no units are available and the operator does not have a unit in
C process, he will not work at all this period. If he is training
C another operator, he will not build at all this period.

IF (AV .EQ. 0 .AND. TOFINI(I,J) .EQ. 0) GOTO 300

C If no new units are available, but a unit is in-process, then he will
C work at the calculated rate. If the rate permits him to finish the
C unit this period, then add one to CUMFIN and COMP variables and
C calculate operator rate for next unit.

IF (AV .EQ. 0) THEN
  IF (TOFINI(I,J) .GT. 0) THEN
    IF (RATE(I,J) .LT. TOFINI(I,J)) THEN
      TOFINI(I,J)=TOFINI(I,J)-RATE(I,J)
    ELSE
      COMP(I)=COMP(I)+1
      CUMFIN(I)=CUMFIN(I)+1
      TOFINI(I,J)=0
      IT(I,J)=IT(I,J)+1
      WCTT(I)=WCTT(I)+1
      LCTIM=ST(I)*MAX(1.0,LCM*IT(I,J)**LCE(I))
      ERTIM=MAX(0.0,WRTF(I)*(1-WCFAC*WCTT(I)-OPFAC*IT(I,J)))
      RTIM = (-ERTIM)*LOG(1-ABS(RANDOM(I))/32768.0)
      RATE(I,J) = 1/(LCTIM+RTIM)
  ELSE
    IF(IT(I,J),EQ,TTTER+1.AND.J.LE.NOT) THEN
      TRNR(I)=TRNR(I)+1
      TRAV(I)=TRAV(I)+1
    ENDIF
    IF(IT(I,J),EQ,ITT+1.AND.J.NE.1) THEN
      TEMPl=INT((I-1)/TAOT+.99)
      N=MIN(NOQ(I),INT(1+TEMPI*TAOT))
      DO 260 M=N,2+(TEMPI-1)*TAOT,-1
      IF(IT(I,M),LT,ITT+1) GOTO 270
    260     CONTINUE
    270     TRAV(I)=TRAV(I)+1
    ENDIF
  ENDIF
ENDIF
ENDIF

C If units are available, then the operator will start a new one if he
C finishes one this period. Otherwise everything else is the same
C as above.

ELSE
  IF (TOFINI(I,J) .EQ. 0) THEN

81
TOFINI(I,J) = 1 - RATE(I,J)
AV = AV - 1
SM(I) = SM(I) - 1
ELSE
IF (RATE(I,J) .LT. TOFINI(I,J)) THEN
TOFINI(I,J) = TOFINI(I,J) - RATE(I,J)
ELSE
COMP(I) = COMP(I) + 1
CUMFIN(I) = CUMFIN(I) + 1
TOFINI(I,J) = TOFINI(I,J) - RATE(I,J) + 1
AV = AV - 1
SM(I) = SM(I) - 1
IT(I,J) = IT(I,J) + 1
WCTI(I) = WCTI(I) + 1
LCTIM = ST(I)*MAX(1.0,LCM**IT(I,J)**LCE(I))
ERTIM = MAX(0.0,WRFT(I)*WCFAC*WCTI(I)-OPFAC*IT(I,J))
RTIM = (-ERTIM)*LOG(1-ABS(RANDOM))/32768.0
RATE(I,J) = 1/(LCTIM+RTIM)
ENDIF
IF(IT(I,J).EQ.IT+1.AND.J.LE.NOT) THEN
TRNR(I) = TRNR(I)+1
TRAV(I) = TRAV(I)+1
ENDIF
IF(IT(I,J).EQ.IT+1.AND.J.NE.1) THEN
TEMPS = INT((I-1)/TAOT+.99)
N = MIN(300,INT(1+TEMPS*TAOT))
DO 280 M = N,2*(TEMPS-1)*TAOT,-1
IF(IT(I,M).LT.IT+1) GOTO 290
280 CONTINUE
TRAV(I) = TRAV(I)+1
290 CONTINUE
ENDIF
ENDIF
ENDIF
ENDIF
CONTINUE
400 CONTINUE
IF (COMP(NOWC) .GT. 0) THEN
DO 410 M = 1, COMP(NOWC)
TEMP = ECOINT/(CUMFIN(NOWC)**2-M)
ECO(0) = ECO(0) + TEMP
CUMECO = CUMECO + TEMP
410 CONTINUE
ENDIF
PPS = 0
DO 430 M = -8.0, 0
PPS = PPS + ECO(M)*PSECO*(.2+.0222*(M))/TOTNOP/HPP
430 CONTINUE
WQ(W) = WQ(W) + COMP(NOWC)
IF(T/HPP .EQ. REAL(T)/HPP) THEN
ECO(0) = INT(ECO(0))
CUMECO = INT(CUMECO)
W = W + 1
WO(W) = 0
DO 450 M = -8, -1
   ECO(M) = ECO(M+1)
450 CONTINUE
   ECO(0) = 0
ENDIF

500 CONTINUE

MO(1) = WO(1) + WO(2) + WO(3) + WO(4) + WO(5) + WO(6) + 0.33*WO(7)
DO 540 Z = 2, MOS
   IF (REAL(Z)/3.0 .EQ. INT(Z/3)) THEN
      Y = 11 + ((Z-3)/3)*13
      MO(Z) = 0.33*WO(Y) + WO(Y+1) + WO(Y+2) + WO(Y+3) + WO(Y+4)
   ELSEIF (REAL(Z-1)/3.0 .EQ. INT(Z/3)) THEN
      Y = 3 + ((Z-1)/3)*13
      MO(Z) = WO(Y) + WO(Y+1) + WO(Y+2) + WO(Y+3) + 0.33*WO(Y+4)
   ELSE
      Y = 7 + ((Z-2)/3)*13
      MO(Z) = 0.67*WO(Y) + WO(Y+1) + WO(Y+2) + WO(Y+3) + 0.67*WO(Y+4)
   ENDIF
540 CONTINUE

PRINT *, 'ITERATION ', ITS+1
DO 590 Z = 1, MOS
   PRINT*, MO(Z)
590 CONTINUE

ITS = ITS + 1
IF (ITS .LT. HMIT) GOTO 5
700 PRINT*, 'SEE YA'
END