Minimizing the Impact of Model Changeovers at a Medium-Mix High-Volume Cellular-Telephone Production Line

by

Kwan "Brian" Y. Tan

B.S., Electrical Engineering, Brown University, 1984

Submitted to the Department of Electrical Engineering and Computer Science and
to the MIT Sloan School of Management
in Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Electrical Engineering and Computer Science
and
Master of Science in Management

at the

Massachusetts Institute of Technology
May 1994

Signature of Author ______________________________
Department of Electrical Engineering and Computer Science
MIT Sloan School of Management
May 6, 1994

Certified by ______________________________
Alvin W. Drake
Professor of Systems Science and Engineering
Thesis Supervisor

Certified by ______________________________
Stephen C. Graves
Professor of Management
Thesis Supervisor

Certified by ______________________________
Jeffrey A. Barks
Associate Dean, Sloan Master's and Bachelor's Program

Minimizing the Impact of Model Changeovers at a Medium-Mix High-Volume Cellular-Telephone Production Line

by
Kwan "Brian" Y. Tan

Submitted to the Department of Electrical Engineering and Computer Science
and to the MIT Sloan School of Management
on May 6, 1994 in partial fulfillment of the requirements for the degrees of
Master of Science in Electrical Engineering and Computer Science
and Master of Science in Management

Abstract

This thesis addresses problems faced by high-volume continuous-flow production of electronics products, such as cellular telephones, when faced with an increasing frequency of model changeovers. The increase in model changeovers increases defects (due to operator error) and decreases throughput (due to changeover downtime.) This thesis presents approaches to quantify the impact of model changeovers on quality and throughput, as well as methods to minimize the impact.

First, the impact of changeovers on quality is addressed through data analysis. This thesis then describes the development of an on-line operator-assist information system to improve quality during the changeover process. This operator-assist-station (OAS) system consists of color X-terminals which provide operators with the right information at the right time—graphical electronic assembly instructions, changeover notification, enhanced factory communication, and on-line training. A set of complete user requirements and graphical-user-interface design (storyboard) is developed, and a prototype system with a subset of functionality is implemented. Additionally, a simple spreadsheet model is provided to estimate the dollar benefits of an OAS system.

Second, this thesis addresses the impact of changeovers on throughput. The methodology to quantify this impact shows some interesting results. Production lines can be very different at changeover efficiency—based on data collected over a four-month period, the throughput loss per changeover per shift was five times smaller at a line efficient at changeovers when compared to another line. It was observed that experience, training, and teamwork played major roles in changeover efficiency. The methodology here provides a new and more revealing way to better understand throughput performance and changeover downtime so that appropriate corrective actions can be determined.

Finally, the lack of a standardized changeover process is identified as a major contributor to changeover downtime. A set of changeover process checklists, based on the best practices at each operator position, is developed to help standardize the process.

Thesis supervisors: Alvin W. Drake, *Professor of Systems Science and Engineering*
Stephen C. Graves, *Professor of Management*

Acknowledgements

This thesis, while bearing the name of one author, has only been possible through the support of many individuals and teams who tirelessly helped, guided, motivated, and inspired the effort. To all of you who contributed so generously, I am most grateful.

I would like to especially thank everyone at Motorola who not only provided their energy and resources to make this project possible but also a great educational experience for me. Through this project, I have come one step closer in my continuous pursuit to learn and apply world-class manufacturing techniques. Special thanks goes to Julie Johnson and Ben Vitale for their constant guidance and direction. Sam Fitting deserves a very big thank you for his tireless help in software and for always being so ready to teach.

I am fortunate to be guided by two of the most dedicated manufacturing faculty advisors. Thank you to Al Drake and Steve Graves—I have always been inspired by your thoughtful ideas and have learned a lot from you both.

I also wish to acknowledge the Leaders for Manufacturing Program for its support of this work.

Finally, I wish to thank my family, especially my parents, and Gurumayi, for everything.

Table of Contents

Title Page 1

Abstract 3

Acknowledgement 5

Table of Contents 7

1 Introduction 9

1.1 Definition of the Problem 9

1.2 Motivation 9

1.3 Contributions 10

1.4 Key Findings 11

1.5 Background 12

1.5.1 What causes a changeover to happen? 13

1.5.2 Overview of production line 13

1.5.3 Overview of products and subassemblies 14

1.5.4 Overview of changeovers 15

1.6 Thesis Layout 17

2 Impact of Model-Changeovers on Quality 18

2.1 Back-End Model-Assembly Problem: Manual Processes 18

2.2 Model changeover situation at model assembly 19

2.2.1 Current changeover problems 19

2.3 Quantifying the Impact 21

2.3.1 A first-order estimate 21

2.3.2 More rigorous methodology 22

2.3.3 Problems with data 23

2.3.4 Guidelines for useful data to collect 24

3 Operator-Assist Stations to Improve Quality 25

3.1 What is an Operator-Assist Station (OAS)? 25

3.1.1 Benchmarking 26

3.2 Developing the OAS User Requirements 26

3.2.1 Objectives 26

3.2.2 Summary of user requirements 27

3.2.3 Graphical-user-interface (GUI) design 28

3.2.4 Methodology to collect the voice of the customer 28

3.3 Implementing a Prototype System at a New Production Line 30

3.3.1 Basic functionality to assist model-change 31

3.3.2 Documenting the model-assembly process 33
3.3.3 Hardware 34
3.3.4 Software 34
3.3.5 Documenter and operator training 35
3.3.6 Maintenance issues 35
3.3.7 Preliminary results and user feedback 36
3.3.8 A simple first-order model to assess bottom-line benefits 38

4 Impact of Model-Changeovers on Throughput 40

4.1 Front-end Changeovers as Bottleneck 40
4.2 Model change situation at front-end 40
4.2.1 What can go wrong? 41
4.2.2 The top three categories of problems 42
4.3 Quantifying the Impact on Throughput 43
4.3.1 Methodology to analyze the problem 43
4.3.2 Results 46

5 Standardizing Front-End Changeover Processes to Improve Throughput 51

5.1 Determining "Best Practices" for Changeovers 51
5.1.1 Checklist/suggestion forms and interviews 51
5.2 Creating a Process Checklist for Each Station 52

6 Additional observations from throughput data 53

6.1 Low bottleneck utilization? 53
6.2 Production metrics: "goal" versus bottleneck utilization 54

7 Results and Conclusion 57

7.1 Results of this research 57
7.2 Conclusions 58
7.3 Suggestions for future work 60

Appendix A: FQA sampling plan 62

Appendix B: User requirements for operator-assist stations 63

Appendix C: Graphical user interface (GUI) design for operator-assist stations 68

Appendix D: Checklist for model changeover at front-end 77

Bibliography 79

1 Introduction

The research for this thesis was conducted at Motorola's Cellular Subscriber Group in Libertyville, IL. The author spent six-and-one-half months on-site to examine issues related to model changeovers at Motorola's cellular-telephone production lines.

1.1 Definition of the Problem

This research addresses the problems associated with high-volume continuous-flow production lines which were traditionally low-mix (focused) but now face medium-mix schedules (under five model changes per shift.) How does one measure the impact of increased changeover frequency in terms of quality and capacity? How can the lines respond more flexibly to an increasing number of model changeovers? That is, how can such lines change models without increased unit defects and reduced capacity?

1.2 Motivation

The negative effects of changeovers

At the time of when this research began, the general perception at this plant was that model changeovers have increased defects and cost, and decreased capacity. Defects are introduced due to operator error while capacity is lost due to changeover downtime. However, the extent which changeovers have impacted line throughput and quality was not understood since no systematic investigation had been done. This provided the motivation for this thesis—to understand and to quantify the impact of changeovers, and to find ways to minimize such impact. The motivation and interest were especially heightened in light of the trend towards a higher frequency of changeovers, higher product mix, and more complex types of changeovers.

1.3 Contributions

This thesis presents approaches to quantify the impact of model changeovers on quality and throughput, as well as methods to minimize the impact.

The necessary data needed to quantify the problem are identified and methodologies for analysis are developed. The data analysis contributes to a better understanding of the effects of model changeovers so that the appropriate schemes and priorities can be developed to respond.

To minimize impact of changeovers on quality, this research examined the use of operator-assist stations (color video terminals) which provide appropriate and timely information to reduce operator errors. The user requirements are collected using the Japanese KJ method[1], and documented under IEEE software requirement standards. A prototype operator-assist-station system, with a subset of the functionality, was implemented in a new production line.

To minimize the impact on throughput, this research identified the lack of a standardized changeover process as one of the largest contributor to changeover downtime problems. The best practices for a changeover at each station were identified and changeover-process checklists were created to help operators reduce changeover time and variability.

---

[1] A KJ is a problem solving tool for collecting and organizing qualitative data, invented by Kawakita Jiro. It is an especially effective way to enable a team of people to develop ideas, organize them, and prioritize them in a consensus fashion. (Shiba, 1992; Kawakita, 1988)

1.4 Key Findings

The results of this research can be summarized into five key findings as follows:

Finding No. 1: The operator-assist station (OAS) concept is feasible and readily accepted by production operators. The implementation of a prototype OAS system demonstrated concept feasibility to provide on-line build-aid instructions. The positive feedback from operators throughout the development showed that operators buy into the idea that OAS will help them reduce assembly errors. In particular, the operators at the prototype-OAS line have a strong sense of ownership for continued OAS development since they were directly involved in the user-requirements development.

Finding No. 2: An OAS system can have many additional features to improve productivity. Originally, the OAS was primarily intended to provide build-aid assembly instructions. However, from the user-requirements collection process, the potential uses of OAS have been expanded to include functionality such as changeover notification, enhanced factory communication, and on-line training. Thus, to take full advantage of an OAS system, it is recommended that the additional features be implemented through further software development.

Finding No. 3: First-order estimates on the impact of changeovers on quality show that 1.8 times more defects are introduced in a high-mix line compared to a low-mix one. However, better factory-quality data tracking is needed for a more rigorous understanding. Data must be separated out by line and by production shift where each unit was produced.

Finding No. 4: A production line efficient at changeovers is minimally affected by changeovers on throughput but others can be significantly affected. The methodology developed here showed that a line efficient at changeovers was much less affected by changeovers than the original general perception at the plant. However, in comparison, data collected over four months showed that a line inefficient at changeovers had five times greater throughput loss per changeover per shift. It was observed that experience and training matter a lot in changeover efficiency. Further observations showed that the changeover process was not standardized to best practices—some operators were inefficient simply because they did not know what the best changeover process was. This research benchmarked and collected data from several production lines to create a set of checklists for the best changeover practices. These checklists should help the different lines minimize the impact of changeovers on throughput.

Finding No. 5: Bottleneck utilization is lower than generally expected. The data analysis from quantifying the impact on throughput showed that the bottleneck machine is utilized less than intuitive expectations (the real numbers are not disclosed in this thesis). Previous production metrics did not measure bottleneck utilization and generally, on the factory floor, many operators and supervisors had little awareness of where the bottleneck machine is nor the need to pay closer attention to it. This finding triggers the need to understand what the industry best-in-class utilization is, what a pareto of the bottleneck's downtime ("non-utilized") hours look like, and where there might be opportunities for improvement. Also, it is suggested that bottleneck utilization be used as an improved metric to understand line performance and to cultivate a culture which actively identifies and manages bottlenecks.

1.5 Background

At Motorola's Cellular Subscriber Group, demand for their portable cellular telephones has been explosive. This rapid growth has been accompanied by model proliferation, changing customer demand for product mix, and material shortages as vendors struggle to keep up with the necessary capacity increases. There are numerous high-volume production lines, each capable of making considerably more than 50-80 models. These production lines have to cope with an increasing frequency of model changeovers. Unanticipated changeovers may be due to customers (cellular service providers) who call to change orders or due to shortened production runs when a certain part runs out.

1.5.1 What causes a changeover to happen?

A brief simplified overview of the process leading to a changeover is as follows. The sales department collects orders from the customers. Monthly and weekly forecasts are provided to the distribution and purchasing departments. Since demand exceeds capacity, distribution allocates the appropriate quantities, model types, and priorities for the different customers or distribution channels. Distribution then informs the purchasing department, which then schedules the production lines accordingly. Re-allocation and re-scheduling continually happens as customers change orders or material shortages for certain models occur.

1.5.2 Overview of production line

The production line is a continuous flow U-shape type essentially containing two parts:

- the front end, consisting of chip-placement machines and reflow ovens, produce the circuit assemblies (i.e. "populate" the circuit boards.) The machines' cycle times are generally balanced within 10% of each other.

- the back end assembles the populated boards with the plastic-housing parts (called the "model-assembly" process) and tests the assembled units for mechanical and electrical behavior.

The front end is highly automated while the back-end model assembly is mostly manual mechanical-assembly work.

Demand typically exceeds the capacity to manufacture. Therefore, the production lines are frequently "upgraded,": at the front end, ever-faster chip-placement machines replace older ones while at the back-end model assembly, more operators are added. Typically, the front-end (with expensive machines) is made the bottleneck. In fact, at many lines where the author performed timing studies, model-assembly throughput rates were 30-80% greater than the front-end.

1.5.3 Overview of products and subassemblies

This section provides a somewhat simplified overview of the products and subassemblies at the production lines where this research was conducted. These lines represent a subset of the many lines at this facility where other cellular-phone products are also made.

The different levels of product families and assemblies fall into several categories. This is illustrated in Figure 1.1 below.

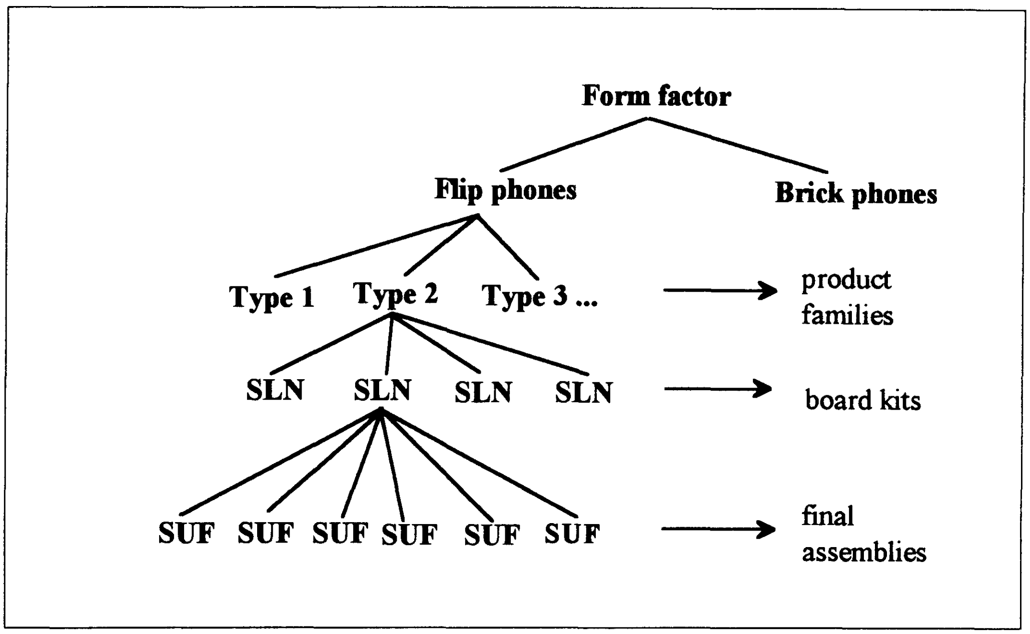


Figure 1.1 Product categories and assemblies

At the top level, there are two major *form-factors* for the portable cellular phones at this factory: "flip phones" and "brick phones." Flip phones are the smaller, lightweight, Star-Trek-like models with a flip which covers the keypad and microphone area. Brick phones are larger, heavier, almost the size of a brick, and do not have a flip. Within each form-factor, there can be several *product families.* For example, at the flip-phone lines where this research was conducted, there were three or more product families, which we will call Types 1, 2, 3, etc. Each product family has a certain set of differentiating features such as weight and software-driven capabilities.

Within each product family, there are several *circuit-assembly board types* or *kits*. Circuit-board kits are the major subassemblies produced at the front end of the line. Each type is given a subassembly part name with the prefix *SLN.* At the production line, an SLN subassembly is first made (i.e. a bare board is populated with chips) and then it is assembled with the various plastic parts into the *final phone assembly* at the back-end/model-assembly portion of the line. The final assembly is given a final model number for the phone with the prefix *SUF.*

There are under 20 different SLN board kits and well over 50-80 SUF final assemblies per factory (due to the further customization with a wide variety of plastic parts at the back end.)

1.5.4 Overview of changeovers

As mentioned earlier, the front-end of the line is typically the bottleneck. Therefore, an SLN (board-kit) changeover which involves front-end downtime (such as changing a stencil or part feeder) results in throughput decrease. On the other hand, a SUF (final-assembly) changeover which involves model-assembly changeover results in quality problems due to operator error during the manual assembly processes. (Note that some SLN or SUF changeovers may only involve automated software change with minimal impact on the line.)

We will now describe the four major types of changeovers according to their levels of complexity or disruption to the production line. First, a form-factor changeover causes the most disruption since essentially everything has to change: the board kits, pallet sizes, plastic parts, tools, fixtures, software test programs, etc. Second, the next level of complexity comes from a product-family changeover. Typically, both the board-kit stencil screeners (one for each side) have to change and most of the plastic parts have to change. (Both SLN and SUF changes.)

Third, within a product family, an SLN changeover can have different levels of complexity and throughput impact on the front end. Some of them are listed as follows, starting with the most complex:

1. Chip-feeder change

2. Stencil screener change

3. Large-IC part change

4. EPROM software change

5. Chip-placement machine program change

Fourth, a SUF changeover within a product family and using the same SLN board will only involve a model-assembly change—different plastic parts such as keypads or displays have to be used.

Because the front-end is the bottleneck, production schedulers use the rule of minimizing circuit-board changeovers when scheduling the lines. Each scheduler typically has five or six lines to schedule, each line consisting of two 12-hour shifts.

1.6 Thesis Layout

Following this introductory chapter, this thesis is organized into four major sections. First, Chapters 2 and 3 covers the impact of model changeovers on *quality*. Chapter 2 presents the methodology to quantify this impact, with data results which compare a low-mix line with a high-mix one. Chapter 3 then describes the development of an on-line operator-assist information system to improve quality during the changeover process.

Second, Chapters 4 and 5 covers the impact of changeovers on *throughput*. Chapter 4 presents data analysis which quantifies this impact at three production lines and compares the results and implications. Chapter 5 describes the creation of process checklists for changeovers and their use to standardize front-end changeovers to best-in-class practices.

The third section of this thesis is covered in Chapter 6, where additional observations from Chapter 4's throughput data are discussed. In particular, the use of "bottleneck utilization" as an improved production metric is explored.

The final section, Chapter 7, summarizes key results and conclusions, as well as provides suggestions for future work.

2 Impact of Model-Changeovers on Quality

2.1 Back-End Model-Assembly Problem: Manual Processes

To provide a reference point on what is meant by "quality,", we shall define quality as the number of defects per unit found during the sampled inspection at the end of the line. This inspection process is called factory quality assurance (FQA) at Motorola. These defects are critical because the units have passed through on-line production tests and inspections and ready for customer shipment.

Interviews with FQA auditors and observations of their quality reports have shown that the impact of model changeovers on quality is largely due to manual processes, especially at the back-end model assembly. That is, when models change, model-assembly operators make more errors. In contrast, the front-end machine operators do not contribute much to model-changeover quality problems because most of the processes are automated, including on-line tests and inspection. For example, if a front-end operator punches in a wrong machine program, either the machine would identify a mismatch with respect to the printed-circuit board, or an electrical functional test would identify a problem before the phone reaches FQA.

While the FQA quality reports do not categorize which defects occurred due to model changeovers, a good indicator of the problem is the category of defects called "Wrong Part" or WP. WP occurs largely due to operator errors when the model changeover calls for a new part. For the month of August 1993, for a particular product which spans across several production lines, several WP defects were found out of several thousand units sampled. All of these defects originated from model assembly, indicating that quality problems due to model changeovers are largely associated with the manual model-assembly process.

2.2 Model changeover situation at model assembly

Currently, when a scheduled model change happens, a front-end operator informs the rest of the crew by word of mouth and places a yellow tag which says "MODEL CHANGE" with the circuit board in a pallet when it travels to model assembly. There, the operators inform each other of the new model and look up a booklet or poster which contains CAD pictures of all the models and the associated unique parts and part numbers. They determine what part numbers, if any, are needed for their respective position, obtain the parts from a nearby rack, and then continue assembling.

In addition, at some of the lines, unscheduled changeovers occur due to repaired units. The repair technicians may group together a number of repaired (disassembled) units and introduce them at the model-assembly for re-assembly. This causes a very high mix situation. At other lines, the production supervisors decided that such a high mix from repaired units causes too much operator error and slows down the line. The repair technicians at those lines complete the re-assembly themselves. However, that is not considered optimal either since the repair technicians are not as familiar with the assembly process compared to the model-assembly operators.

2.2.1 Current changeover problems

Quality audits at the end of the production line have shown that a significant portion of the defects are caused by model-assembly errors related to model changeovers. The reasons leading to such quality-defect problems can be classified as follows:

- *New or inexperienced operators.* Due to the rapid expansion of this plant, new operators are hired at a faster rate than the time needed to train them extensively. As a result, wrong parts such as displays, lenses, and housings can be assembled when a model changes.

- *Ever increasing line speeds.* As faster machines replace older ones and more operators are added to the back end (to keep up with the front end), each operator's model-assembly cycle time drops. Current rates can be significantly below 60 seconds per unit. This gives an operator little time to notice model changes (indicated by the yellow tags on pallets) or to figure out what parts are needed from the booklet or poster.

- *Subtle differences in parts.* For some types of parts, there is little difference in physical appearance from one part number to another. For example, operators find it hard to differentiate between a green LED display versus an orange one.

- *"Low runners."* A line may be running a popular model (a "high runner") during most of the shift but a low runner may be introduced at a small lot size. Operators tend to continue assembling what they are used to, putting a wrong part into the low-runner model. This is especially true when the change is subtle. For example, for some of the large-form-factor phones, the microphone cushion color is gray for only about 10% of the models.

- *Error looking up model book/poster.* The number of models have proliferated to the extent that a model book (in a three-ring binder) which displays pictures of each model and its unique part numbers can be an inch or two thick. Looking up a part number incorrectly can especially be a possibility due to 1) fatigue (many lines are operating at twelve-hour shifts) or 2) increased pressure to build faster in order to make up for production time lost earlier during the shift.

- *Errors from re-working.* Virtually all defective units are re-worked off-line by technicians responsible for troubleshooting and repair. While experienced at repairing defects, these technicians are not as familiar with re-assembly work compared to assembly-line operators. Hence the repaired units tend to have a higher defect rate, including having the wrong parts.

- *Learning curve for different form-factor.* At certain production lines, a major changeover occasionally happens when a different form-factor phone has to be produced—for example, changing from a "flip" phone to a large form-factor "brick" type phone. It was observed that operators could take a few hours to become familiar with the substantially different assembly process. During this learning curve period, the assembly cycle times are much slower, the phones may be assembled incorrectly, or the wrong parts may be used.

2.3 Quantifying the Impact

2.3.1 A first-order estimate

As a first-order estimate of how much changeovers impact quality, factory-quality-assurance (FQA) quality data (% defective units) was collected over a two-month period (June-July 1993) to compare a low-mix line with a high-mix line. Figure 2.1 below summarizes the results.

Data collected June-July '93	Number of models	Number of prod-families	Approx. batch size ratio	FQA quality % defect ratio
Low-mix line	2	1	1	1
High-mix line	> 60	3	0.12	1.8

Figure 2.1 Comparison of defect ratio between a low-mix vs. high-mix line

The low-mix line only ran two models and one product family while the high-mix line ran well over 60 different models and three different product families. The rest of the numbers above are given in ratios to conceal the real values for proprietary reasons. The batch size ratio indicate that the high-mix line changes over about eight times more often.

The FQA sampling plan covered about 8-9% of the assembled units. This was a reasonably large portion so that one can expect that the number of FQA defects found from a line was a close representation of the actual number of defects produced by that line. Appendix A describes the FQA sampling plan in more detail.

The result here shows that the high-mix line produced 1.8 times more FQA defects than the low-mix line. While FQA does not classify whether defects are caused by changeovers, a large percentage of FQA defects were found to be changeover related—for example, wrong housing, wrong lenses, etc.

Note that caution should be exercised in interpreting the above result. The method is not entirely rigorous because the production lines employed different operators and supervisors, some of whom may be more adept at quality assembly and changeovers than others. It was estimated that the lines had people with similar levels of experience. However, there may have been differences and change in experience levels over the two month period. Also, the two lines were at different sub-factories within the plant. Thus, even the management and process-engineering support for these lines were different. Nevertheless, this result provides a useful first-order estimate of the impact of model changeovers on quality.

2.3.2 More rigorous methodology

The most accurate way to quantify the impact of model changeovers on quality is to collect defect data which are broken down by production *line* and *shift* (including date of shift.) The defect data is then correlated with the number of model-assembly changeovers[2] for a given line and shift. Thus, for a given line and shift, data is grouped into shifts with 0, 1, 2, etc. changeovers and the average defect rate for each group is compared. This method is similar to the one described later in quantifying the impact of changeovers on throughput.

It is desirable to collect and analyze data at the granularity of line and shift because each combination has:

- its own team of operators with a certain level of experience and competence at model changeovers,

---

[2] As discussed in Section 1.3.4, model-assembly changeovers are known as "SUF" changeovers. Each model number of a phone at the back end level of assembly is prefixed by a "SUF.

- large daily variation in the number of changeovers.

2.3.3 Problems with data

While the existing FQA sampling plan is a reasonable estimate of the production situation, this research discovered several problems with the way defective units were tracked back to where and when they were built. After considerable effort to use the above mentioned methodology to track defects by line and shift, this project identified several improvements to enhance the process of quantifying the impact of changeovers on quality.

First, the process which tracks when and where (i.e. which line) a defective unit was built can be improved. While a sticker with a "date/line #" code is stuck on the back of each phone, this code is actually assigned at the *pack station* at the very end of the line, not at model-assembly. Since the pack position is slower than model assembly, assembled units backlogged at the pack station of one line would be brought over to another line which has an idle pack station (due to say, a front-end machine downtime problem.) Observations from the daily production "build pack" records showed that there can sometimes be a mismatch between the number of units packed at a given line and the number of units assembled at that line's model assembly. Over a two-week period, up to a third of the units were packed at a line different from where they were originally assembled. Thus, the current FQA reports which uses the "date/line #" code assigned at the pack station is an unreliable way to figure out when and where the unit was actually assembled.

Second, the procedure to enter data for repair units creates some uncertainty in the database which associates a circuit board's unique barcode with the source production line. The line number gets updated to wherever the unit was repaired, not where it was first built. Thus one cannot use the phone's circuit-board barcode to determine which line had assembled that unit. Motorola's Advanced Manufacturing Technology Factory

Control System (AMT FCS) software has the capability to track the original assembly line but the factory has not yet installed that functionality.

Third, manual entry of defects can sometimes lead to inconsistencies in the database. Several such inconsistencies were observed in the data entry at a particular line.

2.3.4 Guidelines for useful data to collect

Some guidelines are provided here on how defect data might be collected in order to be useful not only for this particular analysis, but for other purposes such as tracking the history of a defective unit or the quality performance of a particular line.

Based on the above discussion, the two key changes in the data collection process ought to be:

1. Assign and install the "date/line #" code (at the back of an assembled phone) at model assembly, not pack station. This may be most suitably done at the last model-assembly position.

2. Use the functionality of the AMT FCS software which tracks where a board was first built. This uses separate record fields for "date/line #" of the board barcode to track when a board was first built versus subsequent repairs.

When the above guidelines are followed, data can be analyzed to determine how a production line's defects get affected by SUF changeovers. Such data would be useful for determining the cost of SUF changeovers and whether it is worthwhile to invest in operator-assist stations as described in the next chapter.

3 Operator-Assist Stations to Improve Quality

This chapter describes the use of an information system called operator-assist stations (OAS) to help operators improve the quality at model assembly. The concept is first explained, the user requirements are then presented, followed by a discussion of a prototype system implemented in a new production line, and finally, a first-order model for assessing the net-present value of an OAS system is presented.

3.1 What is an Operator-Assist Station (OAS)?

Given the evident problems which model-changeovers pose to model-assembly quality, how can the production lines overcome these problems? It is proposed that "operator-assist stations" be used, similar to the use of automated assembly instructions at high-mix low-volume lines in other industries, but with expanded functionality (to be described in the user requirements below.)

Operator-assist stations (OAS) here are automated X-terminals which form an *on-line information system* to provide operators with the right information at the right time. OAS's goal is to reduce operator assembly error and improve productivity (i.e. improve assembly quality and cycle time.) For example, when a new model comes down the conveyor, the OAS terminal would automatically update its screen to provide the necessary assembly instructions and part numbers for the new model, as well as prompting the operator to acknowledge that a model change has happened. An OAS terminal can be placed in front of each model-assembly operator as necessary. OAS stations are configured as clients to the host server which runs the factory control system (FCS).

3.1.1 Benchmarking

Systems employing similar ideas have recently been used at other companies such as Texas Instruments in the manufacture of programmable logic controllers (Morse, 1992), at Japanese automobile manufacturing companies (He, et. al., 1992), and at Motorola (Lach, 1993; Barker, 1993). These applications, however, are for high-mix (batch sizes under, say 10 units) *low-volume* lines and mostly limited to providing automated assembly instructions.

Automated assembly instructions have never been employed at Motorola for high volume lines with a medium mix. At this particular plant, the distribution department which packs the telephones with the batteries and other options, uses an electronic system which includes instructions and scanned photographs. The system is developed in a hyper-script environment and has a reasonable graphical user interface. However, it does not use a database and the information is *not* automated, i.e. the operator has to manually type in the part number and perform a series of mouse clicks before the instructions get displayed. Such a system may not be suited for high-volume assembly. Indeed, operators at the distribution assembly/pack lines tended not to use the system unless the assembly is totally unfamiliar.

3.2 Developing the OAS User Requirements

3.2.1 Objectives

Motorola wants to increase the quality of the model-assembly process by making it more flexible to changeovers. In addition, the operator's job should be simplified and cycle time should be reduced. To develop an OAS system to meet such objectives, interviews and team meetings were conducted with the various "customers"—operators, supervisors, process engineers, and CIM engineers. From the suggestions and requirements collected, it became evident that the OAS system has potential to be more powerful than the benchmarks mentioned above.

This section will provide an overview of the user requirements, the method used in collecting these requirements, and a discussion of the requirements of the graphical user interface (GUI). Appendix B describes these requirements in detail. The requirements, including the GUI design, have been documented as functional requirements in a formal Requirements Document following the IEEE format for software requirements. This document has been handed off to Motorola's CIM organization which will either choose to develop the software in-house or through a contractor.

3.2.2 Summary of user requirements

The user requirements can be summarized into the following categories:

1. *Changeover notification.* When a new model arrives at a model-assembly position, several events happen to reduce operator error: a) the terminal screen flashes, b) an audible warning is sounded (beep or voice), and c) an acknowledgment window pops up (as shown at the bottom of Figure C.1, Appendix C.) The audible warning continues until manual acknowledgment is performed (click "OK" on the window.)

2. *Electronic build-aid / assembly instructions.* When a new model arrives and the user acknowledges the model change (as described in item 1 above), the default screen updates with the new appropriate build-aid picture of the current WIP unit as shown in Figure C.1. This screen displays a front view of the model, the current unit's board barcode, and a summary of the assembly process steps.

3. *Learning tool for process instructions and process changes.* As the plant expands rapidly, giving new or inexperienced operators adequate training is becoming problematic. OAS can serve as a learning tool. An operator can click on different windows to review process instructions and process changes on-line for any model and any assembly position. This will be especially helpful when model assembly is idle due to say, front-end machine downtime.

4. *Communication tool.* This will incorporate several capabilities to enhance communication amongst managers/supervisors, operators, and material handlers:

   - an operator sends a message to material handlers, requesting additional material
   - a material handler sends a message to all operators regarding, for example, a defective or obsolete part that should not be used (this is particularly important for messages which must be communicated in between shifts.)

- a supervisor sends a message or announcement to all operators
- a user contributes to an intershift report or reads it.

Please refer to Appendix B for a detailed discussion on these user requirements.

3.2.3 Graphical-user-interface (GUI) design

Throughout the development of the user requirements, the graphical user interface (GUI) for the OAS system was also designed to ensure satisfaction of the user requirements. The designs (and redesigns) reflect inputs for the GUI from Motorola CIM engineers and production personnel. Appendix C shows each of the GUI designs while the descriptions for them are integrated into the user requirements given in Appendix B.

3.2.4 Methodology to collect the voice of the customer

The user requirements were developed through a structured effort to collect the requirements from all involved parties at three different factories within the cellular-telephone plant complex. This section describes the requirements collection process which was thorough yet reasonably simple. Since a structured software requirements collection process is not commonly practiced by the factory CIM engineers, it is suggested that future factory software development efforts should consider following this process. Ample evidence from the literature shows the benefits of defining clear precise requirements up front in the development process (National Research Council, 1991; Smith and Reinersten, 1991) The process used here followed several steps as described below.

First, the operators were given suggestion forms to collect their ideas on paper. The forms not only presented the idea of using computer terminals to provide them with automated build-aid/assembly instructions but also went further asking for improvement ideas to expand beyond build-aid. The question posed was not "What would you like in build aid functionality?" but "What information will help you do your job better?" in order to broaden the potential ideas.

Second, a series of interviews were conducted with operators, supervisors, and manufacturing-process engineers. The interviews included questions on information and communication problems on the production lines. This generated a variety of ideas from diverse perspectives.

Third, the author conducted three team meetings with six operators, a manufacturing-process engineer, and a CIM engineer. The objective of these meetings, which employed process steps similar to KJ meetings, was to share, refine, and select the user-requirement ideas. Each meeting was a continuation of the preceding one, lasting about an hour each. The process steps were:

1. Each operator wrote down each of his or her suggestions from the suggestion form on a piece of "yellow-sticky" (3M Post-it) paper.

2. One by one, each yellow-sticky paper was pasted on a white board. The team critiqued each one and discussed improvements to the original idea. The final refined idea was then written on a fresh piece of yellow-sticky paper and moved to the side of the white board.

3. Having collected and refined the ideas, the team then selected the most important ones through a democratic process. The team members formed a queue and each picked an idea (on a yellow-sticky paper), placing it on the middle of the white board. Each person got to go about three times, giving a final count of about 20 suggestions (user requirements) which is considered a manageable size (for software development.)

In this way, each team member felt that their suggestions were considered democratically while being subjected to improvements through a collective effort. The operators felt very enthusiastic about taking ownership of the project. Some reactions included "Boy, we've designed this thing, so we better make it work." and "Can we have *all* our team meetings be in this format?" The latter comment has been fed back to the organization's training department, which has scheduled KJ training sessions to be taught on site. Given the positive response and support from the operators, it is recommended that the process introduced here be used for other continuous-improvement efforts involving teams.

During the meetings, the author observed that four out of the six operators had many creative ideas while two of them were generally ambivalent about the OAS system and the team process. For the latter type, it helped when very specific ideas of the OAS system were explained to them and they were asked how such scenarios might affect them. Having the CIM engineer present at the meetings was very helpful to quickly eliminate ideas which would be unreasonably difficult to develop from a software perspective.

3.3 Implementing a Prototype System at a New Production Line

Following the priorities set by Motorola management, the implementation of a prototype OAS system has been the main emphasis of this project. To avoid disruption of existing lines, this prototype system was implemented and tested at a new production line which is designated to make new models of digital-cellular telephones. This, however, was a trade-off because the line was also undergoing several disruptions typical of new production lines which were debugging new processes and new products. In addition, only four models were designated for production at this line during the course of this project, with very infrequent changeovers (typically less than once per shift.)

Figure 3.1 shows the model-assembly line configuration. Each operator position has a rack in front of it, on which a color X-terminal sits. The rack is about a foot above eye level (seated position.)

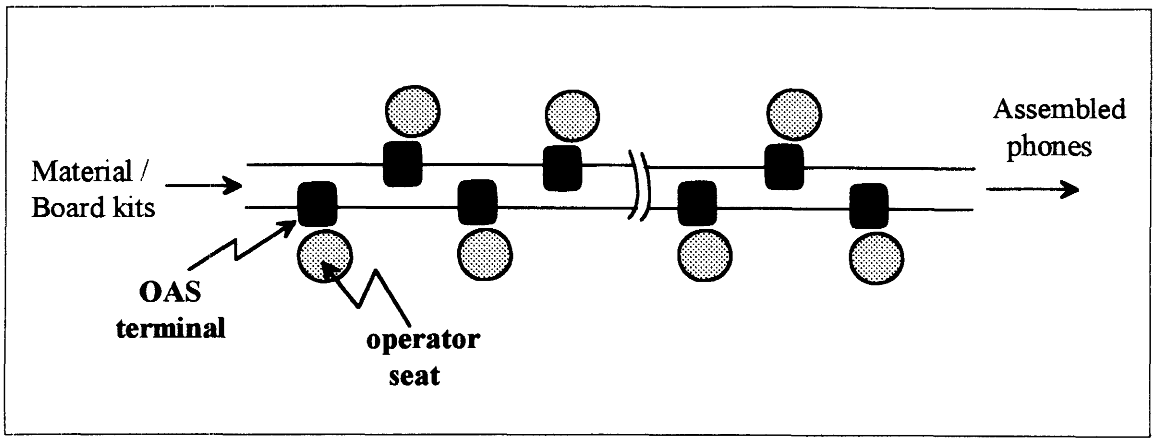


Figure 3.1 Model-assembly configuration with OAS terminals

3.3.1 Basic functionality to assist model-change

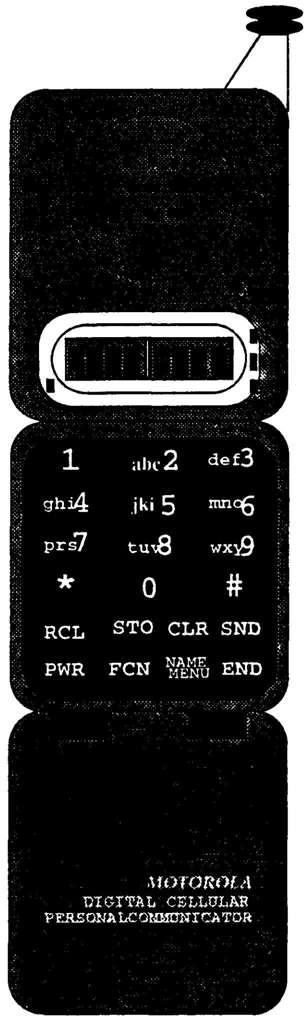
The goals of this prototype system are to 1) provide basic functionality to assist model change at model assembly, 2) prove the OAS concept, 3) collect model-assembly defect data, and 4) identify improvements for future development (i.e. add to the user requirements.)

To provide basic functionality, the following procedure and capability were developed to assist a model change:

1. an operator scans the circuit board's barcode

2. the screen updates with a color "build-aid" picture such as that shown in Figure 3.2. This picture contains a front view of the model with all the corresponding colors, model number and key part numbers, a summary of the process steps, and the necessary tools. Each build-aid picture contains specific information for a model and a position.

3. If a defect is found, the operator simply clicks once on the DOCTAC defect-tracking application's window and enters the appropriate defect.



SUF 1234G

BODY COLOR: STEEL GRAY
EUSEUTCHEON COLOR: BLACK
FRONT HOUSING: # 123456LXX
REAR HOUSING: # 123456JXX
KEYPAD:BLACK # 1234567

DISPLAY
#1234567112

PROCESS STEPS:

1. **Check the housing for cosmetic defects.**
2. **Place the Alert.**
3. **Place the gasket.**
4. **Attach the Mic gasket.**
5. **Place the speaker screen.**
6. **Place the keypad into front housing.**
7. **Insert the keypad into front housing.**
8. **Place the volume switch in the appropriate location.**
9. **Place the speaker.**
10. **Place the alert, then pad.**
11. **Place the Mic with the grommet.**
12. **Attach the left grip.**

TOOLS:

1. TWEEZER
2.PLASTIC TOOL
3.SCREWDRIVER

Figure 3.2 Sample build-aid picture at a prototype OAS terminal

Although there were several operator positions, only three were identified to have prototype OAS terminals. These positions were the ones which experienced unique parts when a model changed:

- Display position. At this position, the operator inserts one of two types of LED displays differentiated by color (green or orange.) This is a rather subtle change since both displays look the same— the LED color is not revealed at this position since the phone is not ready to be powered at this point.

- Keypad position. This is where the operator assembles the keypad and circuit board into the front housing. Here it is more obvious to note the difference in model-specific parts since the display can be either black with white alphanumeric letters or white with black letters. However, at other production lines, several types of keypads exist, so there are some more subtle differences.

- "Quick test" position. This is the last model-assembly position. The operator performs several tests such as 1) inspection of parts for correctness and cosmetic defects, 2) brief functionality test, and 3) drop test. Obviously, it is desirable for the operator to have OAS's build-aid pictures and information.

The basic functionality at these OAS terminals included an on-line defect tracking software called DOCTAC. DOCTAC was an in-house developed text-based package and had already been in use at some other existing lines for tracking electrical defects. It was configured for the first time to track model-assembly defects at this new line.

The initial production plan called for a high-mix of products to be assembled at this line towards the later part of the project. However, several factors changed and in the end, only two model types (families) were assembled at this line throughout the project. For each family, only two models existed, making a total of four models. Thus, we had a low-mix situation. However, in the future, each family can easily proliferate to over 20 models each.

3.3.2 Documenting the model-assembly process

The current practice uses build aids on paper, which is a front-view drawing of the phone and a list of critical part numbers. This project proceeded to extend the functionality of the paper build aids. To understand what information ought to be included in the *electronic* build-aid pictures and in DOCTAC, the author and two process engineers interviewed all the model-assembly operators to document their processes. This also enabled an accurate count of the number of defect opportunities at each step for calculating quality statistics such as *ppm* (number of defects per million opportunities) and *sigma level.*

3.3.3 Hardware

The hardware platform is built on the existing CIM system which uses Motorola UNIX computers called Delta 800 Series and Network Computing Devices (NCD) color X-terminals. The three terminals, placed for easy viewing at their respective model-assembly positions, are connected via an Ethernet network to the MPC and run in a client-server mode. The importance of color screens were noted because build-aid pictures simulated the actual models' colors and the annotation or assembly instructions had colorful highlights. A 15-inch screen size was selected to compromise between ease of viewing, shelf space, and cost.

The terminals were setup and configured by assigning the following network addresses for each: Ethernet, terminal IP, host IP, default gateway, and subnet mask. Other pieces of installed hardware included scanners, wedges, power supplies, and ethernet cables. Trackballs were the preferred input device by the operators over mice since there was limited table-top space. Touch screens were explored but they were very expensive for X-terminals (about $2000 for each screen) and the operators did not like the idea of getting up to reach for the screen.

A factory technician has been trained on all the hardware setup steps for all future installations.

3.3.4 Software

The build-aid application module was written in-house by Motorola CIM engineers. The final selected version uses a C program which, based on the barcode detected, accesses the CIM database to determine the assembly model number and OAS station id, then calls up a standard UNIX utility called "xv" to update the screen with the right build-aid picture.

The author was involved in the following software activities to enable build-aid implementation:

- Created build-aid pictures and annotation on Framemaker (off-the-shelf graphics package.) Framemaker was chosen because it was available on the Motorola UNIX computers and relatively easy to use.
- Converted Framemaker files to gif (graphical interchange format) files in appropriate database-server directory.
- Populated the CIM database tables with build-aid gif files for four models and three model-assembly positions (total of 12 pictures).
- Setup and configured the build-aid application module.
- Configured operator-name entry and certification, and auto login.
- tested the application to ensure that the build-aid pictures switched properly from one model to the next.

3.3.5 Documenter and operator training

Throughout the process, a "documenter" (Motorola lingo for a draftsman who does CAD drawings) was trained and the process documented. The operators were also trained on how to use the OAS system without any difficulties.

3.3.6 Maintenance issues

As more models and terminals are added to this line and others, this prototype system is not anticipated to require more maintenance than the current practice of providing paper hardcopies of build-aid pictures at model-assembly positions. Currently, the factory employs about two documenters full-time to support every five production lines for build-aid. When a new model is introduced or an old one is updated, a documenter makes a new CAD drawing, prints out a large number of color copies, and then installs them at all relevant positions at each line—a process which takes about 10 hours! In addition, the documenter also has to create a detailed process sheet for each position if the new model involves process changes.

With the new on-line system, a documenter does the same CAD work but instead of dealing with the hardcopies, he or she spends some extra time to provide additional information to the pictures:

1. understand the process steps for the new model (through interviewing process engineers or operators),
2. annotate the picture with process steps and tools for each OAS position,
3. convert the graphics drawing from Framemaker to gif format,
4. configure / populate the database, and
5. verify that the pictures are switching correctly at the OAS terminals on the line.

This new process is not expected to take more than 5 hours for an experienced documenter.

For the hardware, the X-terminals are virtually maintenance free once they have been installed.

3.3.7 Preliminary results and user feedback

The prototype OAS system successfully provided the basic functionality to assist model-change and to enter defects at three model-assembly positions. When an operator scans in the board barcode, the OAS terminal updates the correct build-aid picture on the screen within three seconds, displaying also the key part numbers, process steps, and tool list. The user can also switch between the build-aid function to the defect-entry/tracking function with one track-ball click.

Thus, this prototype system has been successful at proving the OAS concept's feasibility. Equally important, the effort showed that the documenter and operators can be trained relatively easily to maintain and use the system, and that these "users" showed generally enthusiastic support. While this system is by no means near the level of intended user-friendliness (as described in the user requirements document), most operators interviewed felt that it will help "make their job easier."

A positive response came from the lead operator of a new line called "Line 2" who requested that the OAS terminals also be installed on his line. This operator had originally worked at the line with the prototype OAS before transferring to start up Line 2. The reasons he gave for request included:

1. OAS provided a robust way of indicating which product has to be assembled without looking up build-aids on paper ("We really want to get rid of all paper lying around the conveyor system.")
2. "No more hassles in figuring out whether the paper drawing we have is the latest rev." (Only one drawing can be current on screen.)
3. "It's easier to train new operators when the process steps are right there with the picture."

However, it should be noted that the feedback and evaluation during this project were rather preliminary due to the infrequent changeovers at this new line, the small number of models designated, and the lack of automatic scanning. Hands-free scanning is being pursued by a team of other engineers using a cost-effective in-house designed optical method. Nevertheless, throughout the implementation of this prototype system, user feedback was collected and incorporated into the user-requirements document which is described in detail in Appendix B.

3.3.8 A simple first-order model to assess bottom-line benefits

It is useful to develop a first-order estimate of the net present value of an OAS system. This section provides a simple model for doing so, with the calculations organized in a spreadsheet as shown in Figure 3.3. We rely on the fact that a high-mix line with more frequent changeovers produces more changeover-induced defects and that the use of an OAS system helps to eliminate a certain fraction of these changeover-induced defects (say 50%). The estimated hardware costs in Figure 3.3 includes terminals, scanners, power supplies, and trackballs, while the software investment and annual maintenance estimates include initial software development and on-going software/database maintenance by documenters and CIM engineers. As is typical for high-technology capital investment, the time horizon used here for net present value calculation is five years. It should be noted though that this simple first-order model does *not* capture some of the intangible benefits of a fully-developed OAS system such as improving operator learning curve and communication among factory personnel.

Five scenarios are presented with varying estimates of the various parameters. While the numbers used are completely hypothetical, factories considering the use of OAS can use the model here to evaluate investment benefits. For example, if a factory has scenarios like one of the five presented, the bottom-line net present value of an OAS system can be $2-11 million.

Scenario	% defective diff: hi vs lo mix	Cost per defect ($)	Units per day per line (2 shf)	Days / yr	# of lines	% def elim by OAS	Annual Cost of def elim by OAS
A	1.0%	$100	1000	360	5	50%	$900,000
B	1.0%	$100	1000	360	10	50%	$1,800,000
C	0.5%	$100	2000	360	20	50%	$3,600,000
D	0.5%	$50	3000	360	50	25%	$3,375,000
E	0.2%	$50	3000	360	100	25%	$2,700,000

Scenario	Hardware cost per line	Total hardware	Software investment	Maintenance per year	Annual cost of capital	PV of 5-yr Outflows	PV of 5-yr Inflows
A	$20,000	$100,000	$200,000	$200,000	5%	$1,165,895	$3,896,529
B	$20,000	$200,000	$200,000	$300,000	5%	$1,698,843	$7,793,058
C	$20,000	$400,000	$200,000	$500,000	10%	$2,495,393	$13,646,832
D	$20,000	$1,000,000	$200,000	$750,000	10%	$4,043,090	$12,793,905
E	$20,000	$2,000,000	$200,000	$1,000,000	10%	$5,990,787	$10,235,124

Scenario	**Bottom-line NPV**
A	**$2,730,634**
B	**$6,094,215**
C	**$11,151,439**
D	**$8,750,815**
E	**$4,244,338**

Figure 3.3 Five hypothetical scenarios on estimating the net present value benefit of OAS. All numbers are disguised for confidentiality.

% defective diff: hi vs lo mix: this is the estimated difference in % of FQA defective units between a high-mix versus low-mix line. These defects are assumed to be change-over induced.

Cost per defect: rough estimate to account for labor (repair, re-test, re-inspect, etc.), material, and opportunity costs.

% def elim by OAS: estimated fraction of changeover-induced defects which the OAS system will eliminate. 50% appears to be a reasonably conservative number to most factory personnel.

Annual cost of def elim by OAS = (% defective diff between hi-vs-lo mix lines) X (cost per defect) X (units per day per line) X (days per year) X (# of lines) X (% defect elim by OAS)

4 Impact of Model-Changeovers on Throughput

This chapter describes a method used to quantify the impact of model-changeovers on throughput. The model-change situations which lead to throughput problems are discussed, followed by the methodology for analyzing the problem, and finally, the (somewhat surprising) results.

4.1 Front-end Changeovers as Bottleneck

As mentioned earlier, this factory's production lines keep the front-end chip-placement machines as the bottleneck—the front end can be 30-80% slower than model-assembly. Thus, in quantifying the impact of changeovers on throughput, this research concentrated solely on the front-end.

4.2 Model change situation at front-end

At the beginning of each shift the production supervisor checks the factory production-schedule board, writes down his or her line's schedule on a board located at the line, and informs the front-end operators (especially the one who operates the first machine) what models and quantities are to be made. The scheduler, supervisor, and the more experienced operators are all aware of scheduling and sequencing to minimize the number of circuit-board-kit changes and setups.

The operator at the first machine initiates the changeover. When there are about 50 more board panels of the old model, this operator obtains the new panels from a material handler and, if a stencil screen change is involved, he or she checks that the next screen is available in the vicinity (sometimes screens in shortage are borrowed by other lines.) The screener machine has the longest setup time but also at least three times the capacity of the chip-placement machines. Usually, an experienced operator takes about 5-10 minutes to set up the screener and under a minute to set up the next machine (chip placer.) Thus the operator screens the last 5-7 board panels of the old model ahead of time before changing the screen (the standard buffer between the screener and chip-placer is 3 panels.) This way, those panels can keep feeding the next machine while the screener is being set up.

The first two board panels of the new model are marked with its model number in black-marker ink in three places, both sides. The first operator informs the next operator of the model change so that the next operator will keep an eye out for the marked new panels and sets-up the next machine accordingly (usually changing the chip-placement program, which takes under a minute.) This flow of information, machine setups, and material propagate down the line. Teamwork and communication are critical to minimize downtime.

4.2.1 What can go wrong?

When a model changeover happens "smoothly," the author had observed virtually no downtime at the front end, even for the most complicated class of changeover within the same form factor, namely a changeover which involves two stencil-screen setups. However, most production supervisors and managers assert that changeovers are creating too much downtime and chaos on the shop floor. What can go wrong during a changeover and how much impact do changeovers have on throughput? To the author's knowledge, these two questions have never been systematically addressed at this plant—most people talked about changeover impact with the "language of emotion."

I decided to investigate front-end changeovers and what can go wrong. Front-end operators at several production lines were requested to page me each time they make a board-kit changeover so that direct first-hand observations can be made. Some specific problems observed include the following:

- Some operators did not screen the last few panels of the old model ahead of time, prior to screen change. Thus, during the screener setup time, which can sometimes take up to fifteen minutes for an inexperienced operator, the next machine can be idle and throughput may be lost (depending on the downstream buffer situation.)

- Some lines had poorer coordination and team work than others. This is especially evident for lines with new operators. One coordination problem observed was that a model change happened during a 15-minute break period when the line was running on a skeleton crew (typically half crew.) This led to a slower changeover process.

- Occasionally, last-minute unscheduled changeovers did not give enough time for operators to prepare with the right board panels and screens. These unscheduled changes can be due to customers changing the order, management decision to expedite a special order, or an unexpected material shortage for the current model.

- Stencil screener can sometimes be unusually long to set up due to problems such as: panel not loaded properly, track width off, incorrect X & Y offset, stencil not level, and software bugs.

- The next stencil screen may not be around because some other line had borrowed it.

- The right chip-placement program may not be installed in the machine. Two reasons observed for this are 1) all programs were erased earlier to debug a problem with the feeders and no one took the time to re-install the programs, and 2) engineering introduced some test programs (each machine only holds 10 programs at a time.) It can take 20 minutes for a maintenance person to re-install a program.

- EPROMs for the new model were not prepared (programmed) ahead of time. There could be a 20-minute wait for that.

4.2.2 The top three categories of problems

The large variability and lack of automated data-collection process on changeover downtime made it difficult to construct a quantitative pareto of the top problems. Nevertheless, the observations and interviews with production operators and supervisors arrived at a rough pareto on the issues which contributed the most to changeover downtime. The following section describes the top three classifications.

1. ***Large variability: dependency on experience and training.*** The experience and training of the operators play a large role in the duration of setup times and in the team's coordination ability to ensure a smooth changeover process (e.g. to not starve a machine).

2. ***Lack of Process Standard.*** While the process steps for running each machine has been clearly documented in "manufacturing process sheets" (MPS), there were no process steps defined and documented specifically for model changeovers. The operators at one line may follow a certain changeover procedure (e.g. screening the last few panels ahead of time before a changeover so that the line can keep running while the screener is being setup), but operators at another line may follow a different changeover process.

3. ***Lack of Bottleneck-Management Awareness.*** Generally, most front-end operators are not aware of which machine is the bottleneck nor are they trained to focus resources to maximize utilization of the bottleneck during a changeover. This is particularly important when a clear bottleneck machine exists. According to the process-engineering group's goals, all machines in this continuous flow line are supposed to be balanced within 5-10% of each other's cycle time. However, with constant upgrades on new machines, software, and new models, some lines have a clear bottleneck machine (which may be, say, 20% slower than other machines) for a period of time before the situation gets fixed (line re-balanced.) Also, on some lines, one machine might give more problems (such as nozzle misfeeds) than others, making it the bottleneck. Operators and supervisors need to be better trained on identifying and managing bottlenecks—that should help improve throughput during changeovers.

4.3 Quantifying the Impact on Throughput

4.3.1 Methodology to analyze the problem

This section describes the methodology to collect the data and to analyze the impact which board-kit changeovers at the front-end have on line throughput. As with quantifying the impact of changeovers on quality, described earlier in Section 2.3, it is necessary to collect and analyze data which break down the problem by production line and shift.

Despite the many tens of models which each production line is capable of making, a large portion of the customization happens at the back-end. Thus there are relatively fewer number of board-kit models and only three types of board-kit families. We shall call these families Types 1, 2, and 3 boards. Within each family, all boards contain approximately the same number of similar components, thereby taking the same cycle time at each machine.

Production metric: bottleneck utilization

What is the best metric to measure the lines in terms of throughput performance versus frequency of changeovers? The standard metric which the factory has been using is boards per hour. While data for such a metric has been easy to collect and well documented, the author found it to be inaccurate and inadequate after measuring cycle times for the three board families at each machine. Each board family takes a different amount of time at the bottleneck machine—Types 2 and 3, for example, take about 10% longer than Type 1. This led to the idea that no matter what board families a line may be producing, the ability to keep the bottleneck producing would be a more appropriate way to measure throughput performance (i.e. to have a metric independent of product family and linked to bottleneck performance.) Thus, bottleneck utilization, as defined below, was chosen as the best metric.

Bottleneck utilization = Time which bottleneck machine spent producing boards divided by total number of hours worked.

Time which bottleneck machine spent producing boards = [Sum of (total # of Type i boards built during shift)*(cycle time per Type i board at bottleneck)]

Data was collected and entered into spreadsheets like the one shown below in Figure 4.1 for three production shifts at two lines—from June 1 to October 15, 1993 for one line, and from July 29 to October 15 for the second line. Note that each spreadsheet is built for an individual shift and line.

Date	Hrs Worked	Type 1 bds built	Type 2 bds built	Type 3 bds built	Number of Bd Chngovers	Bottleneck Time Producing	Bottleneck Utilization
15-Jul							
16-Jul							
17-Jul							
18-Jul							
19-Jul							
20-Jul							
21-Jul							
22-Jul							
23-Jul							
24-Jul							
25-Jul							
26-Jul							
27-Jul							

Figure 4.1 Spreadsheet to build a database for changeover analysis

Based on the "Number of bd chngovers" column, which gives the number of board-kit model changeovers for a given shift (date), one can then build separate tables for shifts with 0, 1, or 2 changeovers respectively by performing queries on the above spreadsheet database. The average bottleneck utilization is then obtained for each respective query. (For the lines examined, the maximum number of board kit changeovers was two.) Thus, we look for *how bottleneck utilization changes* for a given line/shift when we compare all the days with no changeovers versus all the shifts with one changeover and all the shifts with two changeovers.

Note that in addition to changeover downtime, all other factors contributing to bottleneck utilization (or line downtime), such as machine failure, repair or maintenance, are included in the spreadsheet. However, since this data was collected over long periods and the other downtime factors do not bias any particular day with a certain number of changeovers, it was assumed that the non-changeover-related effects will average out. Statistical t-tests on the results discussed below show reasonable confidence levels for this assumption.

4.3.2 Results

This section presents the results of data collected over four months on three different production shifts, two of which work on the same production line. An analysis of how the shifts perform with respect to changeovers and with respect to each other is discussed. Finally, the results, some of which are somewhat unexpected, are summarized.

Two production lines, X and Y, are examined. Further, for Line Y, two shifts are examined—one experienced and one with a rather new crew of operators. The three shifts are labeled as follows:

Shift	Description
Line X day	Day shift running at line X; experienced
Line Y A-day	"A" day shift running at line Y; inexperienced
Line Y B-day	"B" day shift running at line Y; experienced

All the shifts are day shifts. Line Y has two day shifts due to a so-called "4-3-3-4" production schedule where, on a bi-weekly cycle, one shift works four 12-hour days, takes three days off, then works three 12-hour shifts, takes four days off, and the cycle starts all over. Thus the latter two shifts above operate at the same production line (Line Y), with the same machines and support resources.

We will start with examining the results for Line X day, which is known to be an experienced shift, shown at the top right corner of Figure 4.2. All utilization numbers are multiplied by a factor to disguise proprietary information. The average bottleneck utilization for all the days with zero, one, versus two changeovers are shown together with their standard deviations and the percentage of days with such occurrence. For Line X

Figure 4.2 Bottleneck Utilization vs Changeovers
(Bottleneck Utilization numbers are disguised and normalized as a fraction of Line X Day Shift's results.)

* Ave throughput lost per changeover per shift is the ave. throughput lost due to changeovers divided by the average number of changeovers per shift

LINE X DAY: Experienced shift / Efficient at changeovers

No. Bd-Kit Changeovers/Shift	Zero	One	Two
Ave Bottleneck Utilization	ρ%	.97ρ%	.92ρ%
Std Dev	5%	7%	8%
% Occurrences	36%	45%	19%
Drop in Bottleneck Utiliz.	**Baseline**	.03ρ%	.08ρ%
T-Test: Assoc. Probability (alpha)		25.7%	2.3%
Confidence Level (1-alpha)		74.3%	97.7%
Total % Util (Throughput) Lost Due to Changeovers:			δ%
Ave. throughput lost per changeover per shift*			χ%

(45% *.03ρ% + 19% *.08ρ%)

LINE Y A-DAY: Inexperienced shift / Inefficient at changeovers

No. Bd-Kit Changeovers/Shift	Zero	One	Two
Ave Bottleneck Utilization	.94ρ%	.85ρ%	.76ρ%
Std Dev	9%	11%	8%
% Occurrences	47%	45%	8%
Drop in Bottleneck Utiliz.	**Baseline**	.09ρ%	.18ρ%
T-Test: Assoc. Probability (alpha)		8.4%	14.4%
Confidence Level (1-alpha)		91.6%	85.6%
Total % Util (Throughput) Lost Due to Changeovers:			2δ%
Ave. throughput lost per changeover per shift*			3χ%

LINE Y B-DAY: Experienced shift / Inefficient at changeovers

No. Bd-Kit Changeovers/Shift	Zero	One	Two
Ave Bottleneck Utilization	1.13ρ%	.97ρ%	.97ρ%
Std Dev	5%	8%	5%
% Occurrences	25%	61%	14%
Drop in Bottleneck Utiliz.	**Baseline**	.16ρ%	.16ρ%
T-Test: Assoc. Probability (alpha)		0.008%	0.6%
Confidence Level (1-alpha)		99.99%	99.4%
Total % Util (Throughput) Lost Due to Changeovers:			4δ%
Ave. throughput lost per changeover per shift*:			5χ%

T-Test for Zero-Change Days	alpha =	0.01%
between Shift Y-A and Y-B	Confidence Level =	99.99%

day shift, the average bottleneck utilization is given a baseline of $\rho\%$ while the days with one changeover and two changeovers experienced $.03\rho\%$ and $.08\rho\%$ drop in bottleneck utilization respectively. While the standard deviations are quite large, the t-tests[3] show reasonable levels of confidence (74% and 98%) that the average bottleneck utilization numbers are different. The total % utilization or throughput loss due to changeovers is a relatively low $\delta\%$, obtained by summing the fact that on 45% of the days there is a loss of $.03\rho\%$ (due to one changeover) and 19% of the days there is a loss of $.08\rho\%$ (due to two changeovers.) Finally, the average throughput lost per changeover per shift for this line, obtained by dividing the total throughput lost by the average number of changeovers per shift, is found to be $\chi\%$ as a baseline normalizing number.

We will now examine the results for both the Line Y shifts (bottom half of figure) and compare them with Line X day above. (Note that the confidence levels for these statistics are much higher.) Several observations are noted. First, Line Y B-day at the bottom right of the figure shows much higher bottleneck utilization. This is expected since the factory's standard throughput metric has often ranked this experienced shift as one which produces one of the most boards per hour. On the other hand, Line Y A-day, an *inexperienced* shift which uses the same machines and support resources as Y B-day, shows much lower bottleneck utilization. This highlights the importance of the learning curve.

Second, the drop in bottleneck utilization due to changeovers is much higher for the Line Y shifts (compared to Line X), leading to a total % utilization (throughput) loss of $2\delta\%$ and $4\delta\%$ at the A and B shifts respectively. Additionally, the average throughput lost per changeover per shift is $3\chi\%$ and $5\chi\%$ respectively, compared to $\chi\%$ for Line X day. These results correspond with the author's observations that the Line X day shift

---

[3] The t-test conducted uses the @TTEST function in LOTUS 1-2-3. @TTEST (range1 ; range2 ; [type] ; [tails]) performs a Student's t-test on the data in range1 and range2 (such as all shifts with zero changeover versus one changeover) and returns the associated probability or significance level alpha. The confidence level is (1-alpha). [type] was specified for samples drawn from populations with unequal variances (heteroscedastic), and range1 and range2 do not have to contain the same number of spreadsheet cells. [tails] was specified for a two-tailed t-test.

changeovers very efficiently and with a well-coordinated team effort. This suggests that even an experienced shift such as Line Y B-day may not be efficient at changeovers and can improve at changeover performance, perhaps by benchmarking or learning from Line X day shift.

Third, one should note that Line X day has lower average bottleneck utilization than Line Y B-day. Thus, when comparing these two experienced lines over the four-month data collection period, the former was efficient at changeovers (only χ% average throughput loss per changeover per shift) but have not been effective at keeping the bottleneck busy producing boards. This may be due to a) operator inefficiency or, more likely, b) machine breakdowns anywhere in the line starving or blocking the bottleneck. (Over a three-week period, the author observed that one particular machine continually stopped in the midst of populating boards due to intermittent minor-adjustment problems but such problems were not formally recorded nor measured.)

Summary

The effort to quantify the impact of changeovers on throughput can be summarized as follows:

1. Production lines can be very different at changeover efficiency—based on four months of data to compare Line X day and Line Y B-day, a factor of five difference in average throughput loss per changeover per shift was observed. This confirms the earlier pareto of experience / training at changeover as critical.

2. There is some trend towards greater throughput losses with greater frequency of changeovers.

3. The methodology presents a new and more revealing way to measure and motivate line performance. It exposes line behavior and points to where more emphasis might be needed: for example, machine training & maintenance for Line X day (low bottleneck utilization); changeover-efficiency training for Line Y B-day (high bottleneck utilization but inefficient at changeovers); and both types of training for Line Y A-day (inexperienced line.)

4. Generally, the bottleneck utilization numbers (the real ones are not shown here for proprietary reasons) were lower than most factory personnel's expectations. This will be discussed further in Chapter 7

Given the results we have, it is important to develop an understanding of the cost of "changeover inefficiencies" or poor practices. The spreadsheet below gives some estimates to translate a hypothetical 7% loss of throughput (when changeover-efficient lines are compared with changeover-inefficient ones) into annual profits forgone. Since this is a business which is typically capacity constrained, each unit of throughput lost is directly translated into profits forgone. Some hypothetical scenarios are presented for different line throughputs per day (two shifts) and different profits per unit. (Note that the numbers below are not based on any real data for Motorola.) The significant profits forgone *per line* indicates the importance for every production line to acquire the best skills in the changeover process.

Units per day (2 shft)	Throughput Loss %	Daily Units Lost	Profit per unit	Total annual profits forgone per line due to changeover inefficiency
500	7%	35	$50	$612,500
1000	7%	70	$50	$1,225,000
2000	7%	140	$50	$2,450,000
4000	7%	280	$50	$4,900,000
500	7%	35	$100	$1,225,000
1000	7%	70	$100	$2,450,000
2000	7%	140	$100	$4,900,000
4000	7%	280	$100	$9,800,000
500	7%	35	$200	$2,450,000
1000	7%	70	$200	$4,900,000
2000	7%	140	$200	$9,800,000
4000	7%	280	$200	$19,600,000

Figure 4.3 Hypothetical scenarios to estimate annual profits forgone per line due to changeover inefficiency

5 Standardizing Front-End Changeover Processes to Improve Throughput

Given that major contributors to front-end changeover downtime are due to the lack of experience, training, and a standardized changeover process, what solution should be implemented to improve throughput? This chapter describes a first-step solution which the author has created, which is to determine the best practices for changeovers at each position and to create changeover process checklists for front-end operators to follow. In this way, operators can learn and be trained more effectively and the different production lines can standardize the process to the best way.

5.1 Determining "Best Practices" for Changeovers

As discussed in the last chapter, the author had observed that some operators at certain lines had a better changeover process than others —there was little consistency and operators did not have a process-instruction sheet on how to do a changeover. Thus, the first step was to determine what the best changeover process is for each of the front-end positions.

5.1.1 Checklist/suggestion forms and interviews

Front-end operators at four production lines were asked to fill out a form on each of their process steps when they make a changeover as well as suggestions for improvements. In addition, operators and supervisors at three production lines were interviewed on the process steps.

5.2 Creating a Process Checklist for Each Station

The data from the checklist/suggestion forms and interviews were compiled and compared. The author then determined the best process steps based on the criterion of minimizing the impact on line throughput. Checklists of the process steps for model changeover were then created for each position. The checklists were designed to be unlike the standard factory manufacturing process sheets which are typically several pages long, but rather like a concise summary under a page. This way, the operators have a simple list of steps to perform during a changeover. Examples of such checklists for two of the positions are shown in Appendix D. Note that the checklist consists of considerations for the operators at the *beginning* of the shift as well as during the shift.

In addition to providing a standardized reference for the best changeover practices, these checklists serve as a useful training aid for new operators. During training, it is recommended that operators perform changeover drills following these checklists. They are now under consideration for implementation.

6 Additional observations from throughput data

This chapter describes some additional observations from reviewing the throughput data discussed in Chapter 4. The issues raised here relate to the overall goal of improving throughput, but do not specifically relate to changeovers. It also illustrates how the data collection process from one improvement effort can raise some important questions and reveal the need for other investigation.

6.1 Low bottleneck utilization?

Most people reacted with surprise at the low bottleneck utilization numbers from the data analysis (Figure 4.2, numbers disguised.) What is the industry best-in-class machine utilization? Are the factory's numbers normal or is there a significant opportunity for improvement?

Equally important, what does a pareto of the bottleneck's downtime ("non-utilized") hours look like? This downtime is comprised of the following:

- chip feeders need replenishment of component parts
- machine waiting for operator to react to a red light at the top of machine (which indicates that attention is needed)
- machine downtime due to hardware failure or software bug
- machine starved or blocked
- machine waiting due to material shortage
- changeover downtime

There are currently efforts underway to develop such a pareto. That will focus efforts for certain improvements. For example, if machine starvation or blockage is dominant, investigations to increase the buffer size should be considered against the tradeoffs of higher inventory. Or, if operator response time to a machine which needs attention (such as feeder replenishment) is important, priority consideration should be given to the use of split feeder tables or a paging system which pages an operator when a machine detects the need for an upcoming feeder replenishment.

6.2 Production metrics: "goal" versus bottleneck utilization

This section discusses the current throughput metric for the production lines called "goal" and compares it to bottleneck utilization as defined in the previous chapter. The drawbacks of goal are illustrated and bottleneck utilization is suggested as an alternative metric.

"Goal," which is a throughput number (boards per hour), is determined by the process-engineering organization during the design of the line. It is derived from anticipated theoretical best throughput minus historical adjustment factors for machine reliability and operator-related inefficiency. This throughput goal is typically not increased until the front-end machines (or some other bottleneck part of the line) are upgraded. Using a single goal number to measure the lines can be a simple and efficient way (operators simply have to count how many boards got built every hour), but it can also have several undesirable effects on behavior.

First, a static goal number may produce a self-fulfilling prophecy, i.e. when a production line team is told that it is expected to perform up to a certain goal, it may very well get to the goal but not exceed it. And there is no strong local incentive for the team to push beyond goal consistently. In one incident, for example, when the author observed an inefficient changeover process and suggested a better way, the operator's comment was, "No problem, we're already above goal today." In other words, a static throughput goal does not pass as a continuous-improvement metric.

On the other hand, bottleneck utilization can be used to drive continuous improvement. It would be equivalent to the Japanese "zero defects" quality program. One will never get to 100% utilization nor zero defects but there is always room and motivation for continuous improvement.

Second, it may be too inaccurate to measure high-mix lines with a single goal or throughput number. While in the past, production lines were focused on one product family, the recent trend is for higher product mix, with each product family utilizing different amounts of bottleneck cycle time. For example, at the lines examined in the previous chapter, two of the product families spend 10% more bottleneck time than the third family. As shown there, bottleneck utilization is a relatively simple but accurate measurement for throughput performance, regardless of product families.

Finally, the use of bottleneck utilization promotes a culture of bottleneck-management awareness. On some lines, the machines may not be entirely balanced due to hardware or software upgrades—the author measured instances when a clear bottleneck existed, at 10-20% slower than others. Yet, most production supervisors and operators were not aware of 1) which machine is the bottleneck, or 2) whether the bottleneck is the same for the different product families. Bottleneck utilization as a metric helps line personnel to focus attention at keeping the bottleneck busy. In particular, line supervisors will take more interest in performing timing studies on each machine for the different product families. The increased involvement of supervisors in the operations management of the line (as opposed to their people-management role) will benefit the overall goal of increasing throughput.

Note on cost accounting: there may also be opportunity for improved cost accounting whereby overhead is allocated according to the time which product families spend at the bottleneck. In the cases examined, product-family types 2 and 3 utilize 10% more bottleneck time than type 1.

7 Results and Conclusion

7.1 Results of this research

The results of this project can be summarized into the following categories:

Data analysis to understand line performance and changeover impact

This research established methodologies to quantify the impact of changeovers on quality and throughput. The methodology for throughput proposed bottleneck utilization as a more effective and revealing way to measure and motivate line performance. The data analysis exposed line bottleneck utilization and changeover downtime problems so that corrective actions can be directed appropriately. For the impact of changeovers on quality, only a first-order estimate could be obtained at this point. To understand the situation better, more rigorous data analysis needs to be performed through improved data collection.

Operator-Assist Stations to improve quality

This research developed the concept of operator-assist stations (OAS) for on-line graphical build-aid and work instructions, changeover notification, enhanced factory communication, and on-line training. A set of complete user requirements and graphical user interface design (storyboard) was developed. Further, the project proved the preliminary feasibility of the OAS concept through implementation of a prototype system with a subset of functionality. Very positive operator buy-in and response has been established as a result of conscientiously involving the operators throughout the design, development, and prototype implementation. Finally, a simple spreadsheet model was offered to estimate the dollar benefits of an OAS system.

Changeover process checklists to improve throughput

The lack of a standardized changeover process was identified as a major contributor to changeover downtime at the front-end of the line. Thus, to improve throughput, changeover process checklists based on best practices for each front-end position were created to standardize the changeover process.

7.2 Conclusions

The impact of model changeovers on quality and throughput at high-volume medium-mix production lines can be better understood through the data-analysis methodologies presented here. The impact on quality can then be minimized using operator-assist stations (OAS). For throughput improvements, which depend on the line's ability to utilize the front-end machines, operator training and skill-development at changeovers are critical.

Due to two major trends—more frequent changeovers and ever-faster front-end machines—it will be increasingly important to simplify and automate the back-end model-assembly tasks so that operators can keep up. (The current practice has simply been to add more operators at the back-end, which not only adds cost but also uses up limited production floor space.) The OAS system proposed here will help to improve model-assembly quality, simplify operator tasks, and improve training and communication.

The general response and support for OAS from production operators, supervisors, and process engineers have been very positive throughout this project's development efforts. For that reason, as documented in the user requirements, many ideas for additional functionality beyond graphical build aid (the original OAS intention) have been generated collectively. In particular, production operators had a strong sense of ownership for OAS's success because they had direct input throughout the development and prototype implementation. The prototype system verified basic functionality to assist with model change and that documenters and operators can be trained relatively easily to maintain and use the system respectively. The positive feedback, while preliminary, justifies further development of the prototype to add the functionality described in the user requirements.

To better understand the significance of changeover-related quality problems and to make investment decisions on OAS, better factory-quality-assurance (FQA) data is needed. Specifically, there needs to be better ways to track defective units to the source production line. Such data will enable more rigorous understanding of the impact of changeovers on quality which, in turn, will enable better evaluation of the net present value of an OAS system.

On the throughput aspects, the data analysis on the front-end drew several interesting conclusions. First, production lines can have very different changeover-efficiency performance—for data collected over four months, one line had five times higher throughput loss per changeover per shift compared to another. This demonstrated the importance of training and skill at front-end changeovers. Thus, it is important for operators to follow a standardized changeover process such as the checklists developed here.

Second, for some lines such as the inexperienced Line Y A-Day, there is a clear trend of increased throughput loss with increased frequency of changeovers. While there is no data on what might happen if there were more than two changeovers per shift, this impact on throughput may get worse.

Third, there needs to be more awareness of bottleneck utilization. Most people reacted with surprise that the bottleneck utilization is quite low. Many operators and production supervisors were not aware of which machine is the bottleneck or whether the bottleneck changes with different product families.

Finally, the results also stimulated some questions for further investigation on the metric of bottleneck utilization. Is bottleneck utilization low by industry standard? Should bottleneck utilization be used as a more accurate measure of line throughput independent of product type and also as a metric for continuous improvement?

7.3 Suggestions for future work

The user requirements in Appendix B help to define the software development which needs to be done. In addition, automatic scanning of the circuit-board barcodes at the model-assembly conveyor system needs to be developed. Together, these two development efforts will give the OAS system its intended first-generation functionality.

An OAS system should be implemented in an existing line with high-mix so that comparisons of quality improvements can be made before and after.

As discussed, an improved way to track FQA defective units to the source production line will enable better understanding of the impact of changeovers on quality.

Looking into the future, the use of OAS will help model-assembly keep up with the front-end by reducing cycle time as well as errors (operators slow down when changeovers occur, especially when the form-factor changes.) Several improvements for OAS in the future can be identified. First, there is opportunity to directly download mechanical CAD drawings to the OAS database, eliminating the need to re-draw many build-aid pictures. Second, with the rapid advances of multimedia computing, it will soon be simple and inexpensive to provide on-line (and perhaps interactive) video training clips into OAS. This will help make improved training readily accessible. For example, whenever the front-end is down, operators can review and learn other positions while sitting at their stations.

Finally, there are several potential uses of OAS stations at the front end. First, OAS stations can provide build-aid information for the inspection station at the end of the front-end, where circuit assemblies are inspected for correct component placement. Defects can then be found and fed back quickly to the front-end machine operators. The improved feedback on quality can then improve throughput since less circuit assemblies would be tied up for repair. Second, during breaks when the line runs on half crew, OAS terminals can alert operators at the component hand-place positions when a machine needs attention. (The hand-place positions, being a manual process, are staffed all the time so during breaks or absenteeism, the hand-place operators help to run the machines nearby.) For such an application, the OAS screen will be positioned in front of the operator for easy viewing and it will display the status of the various machines: running, starved or blocked, or needing attention (equivalent to a flashing red light at the machine, which typically means that a component feeder needs to be replenished.) For the latter situation, the OAS terminal can beep so that the operator can respond quicker, minimizing lost time. Third, an OAS system can use the shop-floor control WIP count to calculate bottleneck utilization averages for the hour, week, and month. By tracking this, operators will be more motivated to manage the bottleneck and to increase machine utilization.

Given the initial level of acceptance of the prototype and the potential future applications and capabilities, it is recommended that the factory continues to develop the OAS system. The full benefits of OAS will only be realized through a structured, committed development effort which continually involves the operators' input to ensure their acceptance at each successive generation.

Appendix A: FQA sampling plan

At the end of the line, the phones are packed in trays of 30. Under normal conditions, FQA auditors sample 5 out of 30 phones from a tray. On average, about 50% of the trays are sampled. (For the month of August 1993, the average % of trays sampled was 53%, with typical values in the range 40-60% range.)

For the purposes of the analysis in this research (and from observations of the way FQA auditors worked), the samples can be considered random. Also, the defects did not seem to cluster in any predictable fashion—observations of FQA reports show that 90-95% of the defect entries are on single units.

Appendix B: User requirements for operator-assist stations

The development of these user requirements is described in Section 3.2. Below, each requirement is described in a format compatible with IEEE software requirements standard—a format currently adopted by Motorola's Advanced Manufacturing Technology organization. Each requirement has an *input* and an *output*. The input is an event(s) which initiates or triggers the functionality of that requirement. The output is the set of events or actions which the OAS system performs to satisfy that functional requirement. If the same (old) model were to arrive at a station, then there is no input trigger and there is no change on the screen's output. All figures referred to in this section are found in Appendix C, which shows all the graphical-user-interface (storyboard) design for the requirements here.

Requirement 1: Changeover Notification

Input

New model arrives at the operator position.

Output

a) Terminal screen flashes.
b) Terminal gives audible warning (could be beep or voice.)
c) Acknowledgment window pops up (as shown at the bottom of Figure C.1, Appendix C.); audible warning continues until manual acknowledgment is performed (click "OK" on the window.) Alternatively, the production line is stopped (e.g. the conveyor will not allow the operator's pallet to leave his/her station) until acknowledgment is performed. If this second option is chosen, there should be a manual mechanical override for the conveyor in case a terminal or host computer is down/malfunctioning.

Requirement 2: Electronic Build Aid (Graphical Assembly Instructions)

Input

Either:
a) New model arrives at position, or

b) User clicks the "Choose Model" button on Default Operator Screen (Figure C.1) to request a certain model to be displayed. A window pops up, displaying a list of all active product models. The user selects the desired model and clicks OK. This gives an operator the flexibility to review the build-aid instructions or part numbers for any model. For example, this is useful if there were a repair unit which needs to be checked, or if the operator wanted to quickly learn the process steps for a new model.

Output

When a new model arrives and the user acknowledges the changeover (as described in Requirement 1 above), the default screen updates with the new appropriate build-aid (graphical assembly) picture as shown in Figure C.1, Appendix C. This screen should display the following:
a) front view of the model
b) highlighted unique part numbers or new parts
c) list of tools
d) current unit's board barcode (unit that arrived at the station)
e) "condensed MPS": a concise step-by-step list of major process steps (as shown in Figure C.1)
f) for the *pack* station, a unique build-aid picture with the following should be provided: 1) picture with front and back of model, 2) label placement, and 3) part numbers for the labels and lens.

The "Choose Model" button/option also displays, after the user selects the desired model, a build-aid picture.

In addition, when the user clicks the "Position Detail" button, the build-aid picture should be replaced by a position-specific detailed picture (see Figure C.2) which can be comprised of one or more of the following:
a) assembly drawing relevant to that position (isometric three-dimensional drawing),
b) scanned picture
c) annotation which may highlight or point to a specific task (e.g. "Remember to solder here.")

Requirement 3: Learn/Review MPS (Manufacturing Process Sheets)

Input

User clicks the "MPS" button in Figure C.1 when he/she wishes to review or learn the manufacturing process steps at his/her position or at any other position.

Output

The operator-assist stations should serve as a learning tool for operators. An MPS window pops up as shown in Figure C.3, displaying the MPS (process steps) for the current model and position (default.) If the user clicks on the "NEW PRODUCT" button, then there should be a window which gives a list of all the active models. The user selects a model, clicks OK, and the screen returns to Figure C.3 with the MPS for the selected model. Similarly, the user can select a different position by clicking the "NEW POSITION" button. In this way, an operator can learn or review the process steps for his/her position or any other positions. (This is especially useful for training inexperienced operators when model-assembly is idle.)

When the user clicks "CLOSE" in Figure C.3, the screen will return to Figure C.1.

Requirement 4: Learn/Review MEMs (Manufacturing Engineering Memos, or process changes)

Input

Either:

1) User clicks "MEMs" button in Figure C.1 when he/she wishes to review the MEMs (manufacturing engineering memos which announces process changes) for any position, or
2) A new MEM is entered into the database. All new model-assembly MEMs should be read by all model-assembly operators.

Output

For Input 1), the MEMs window pops up as shown in Figure C.4, with a list of all the active MEMs for that product and position, with dates in reverse chronological order. The user selects the required MEM and clicks OK. Figure C.5 pops up to display the MEM content. After reviewing the MEM, the user clicks CLOSE and Figure C.1 returns to the screen.

Note: initially, it may be easiest to simply list *all* MEMs for model assembly of a particular product, regardless of position. Thus, any MEM can be viewed from any position without having to select the "POSITION" button.

In Figure C.4, the user has the option to select a new product or position. In addition, the user can click the "SEARCH" button and a window should pop up to allow a search of word(s) in the MEM titles with wildcard capability. This should be a global search of *all* model-assembly MEMs, i.e. not position or model specific. For example, if an operator has a problem with keypads and need to read all the MEMs concerning this part, the search will find all MEMs with the character string "*keypad*" in their titles. The window will then list all those relevant MEMs and the user can then select the desired one.

For Input 2), whenever a new MEM of a given model is entered into the database, Figure C.4a should pop up at all model-assembly positions if that model is being built, no matter what the current screen is and who the user is. This window should pop up for seven days from the time the given model is first built at a given line. This ensures that a new MEM is viewed by all positions and translated into action. The user clicks OK in Figure C.4a and Figure C.5 pops up to display the new MEM. This will continue to be displayed until a new model is detected, in which case the "MODEL CHANGE" acknowledgment window as shown in Figure C.1 pops up. When the user clicks OK, either Figure C.1 (build-aid picture) will be displayed or, if this new model has a new MEM associated with it, Figure C.4a should pop up, prompting the user to view the new MEM.

If possible, the most recent MEM should be displayed as the "screen saver" when the terminal is idle.

Requirement 5: Messaging Capability

Input

Either:

1) An operator clicks on the "MESSAGES" button in Figure C.1 to send a message to material handlers, requesting additional material when a part runs out at his/her bin.

2) A material handler clicks on the "MESSAGES" button in Figure C.1 to send a message to all operators regarding, for example, a defective or obsolete part that should not be used. This is particularly important for messages which must be communicated in between shifts.

3) A production supervisor clicks on the "MESSAGES" button in Figure C.1 to send a message to all operators.

4) A user clicks on the "MESSAGES" button in Figure C.1 and then on "INTERSHIFT REPORT" in Figure C.6 to either read or add to the Intershift Report notes.

Output

For Input 1), Figure C.6 pops up and the user clicks on "TO REQUEST MATERIAL" button. Figure C.7 then pops up and the user clicks on the checkmark column to select the parts which need to be delivered to his/her position. Clicking a second time on a part deselects it. (Quantity need not be indicated since standard quantities needed to

fill an operator's bin is implied.) The user clicks OK and a confirmation window Figure C.7a pops up. Clicking YES sends the material request to the material handler's station.
(NOTE: the GUI for the material handlers' window needs to be designed. This should indicate the part numbers and common part name, which position requested it, and which production line.)

For Inputs 2) and 3), Figure C.6 pops up and the user (material handler or supervisor) clicks on "TO SEND SUPERVISORY MESSAGE" button. Figure C.8 pops up. The user clicks on the production line(s) where the message should be sent (default: all lines.) Then the user clicks on the Message Content box and types a message up to 3 lines (80 characters/line.) When ready, the user clicks DONE and a confirmation window pops up as shown in Figure C.8a. Clicking YES sends the message, popping a message at all model assembly stations with Figure C.8c and sounding an audible beep. When the receiver at model assembly clicks OK, the message is displayed.

Each time Inputs 1), 2) or 3) is used, the messages are automatically logged in to an Intershift Report with Date and Time stamp. All operator input should be specific while supervisor and material-handler messages will be global to all lines and positions where the message was sent.

For Input 4), Intershift Report, Figure C.9 pops up. The user can then scroll through the upper portion of window to view two reports: one for the current shift and one for the last shift. The user can also click the bottom/smaller portion of the screen and add to the current-shift report. This should be an open-ended sliding window with five lines scrolling and no limit on text. After typing the necessary text, the user clicks ADD TEXT and will see the text added to the Current Shift report at the top box with date and time. Supervisors and material handlers can delete text but operators may not. When DONE is clicked, the default build-aid window Figure C.1 pops up.

Requirement 6: Cycle Time Monitoring & Bottleneck Identification

This requirement needs to be developed further. It is included here briefly for the sake of completion. OAS can be used to monitor and display the assembly cycle time of each model-assembly position. This gives an efficient way to identify the bottleneck(s). Since the pace of the line is determined by the bottleneck, operators can help each other to support and improve the bottleneck-operator's cycle time.

The work-in-process (WIP) status, consisting of the SUF model numbers and quantities, at upstream stations can also be displayed. With such information, operators can know what models will be coming, thereby preparing themselves for upcoming model changeovers.

Appendix C: Graphical user interface (GUI) design for operator-assist stations

Figure C.1 Default operator-assist screen

Appendix C: Graphical user interface (GUI) design for operator-assist stations

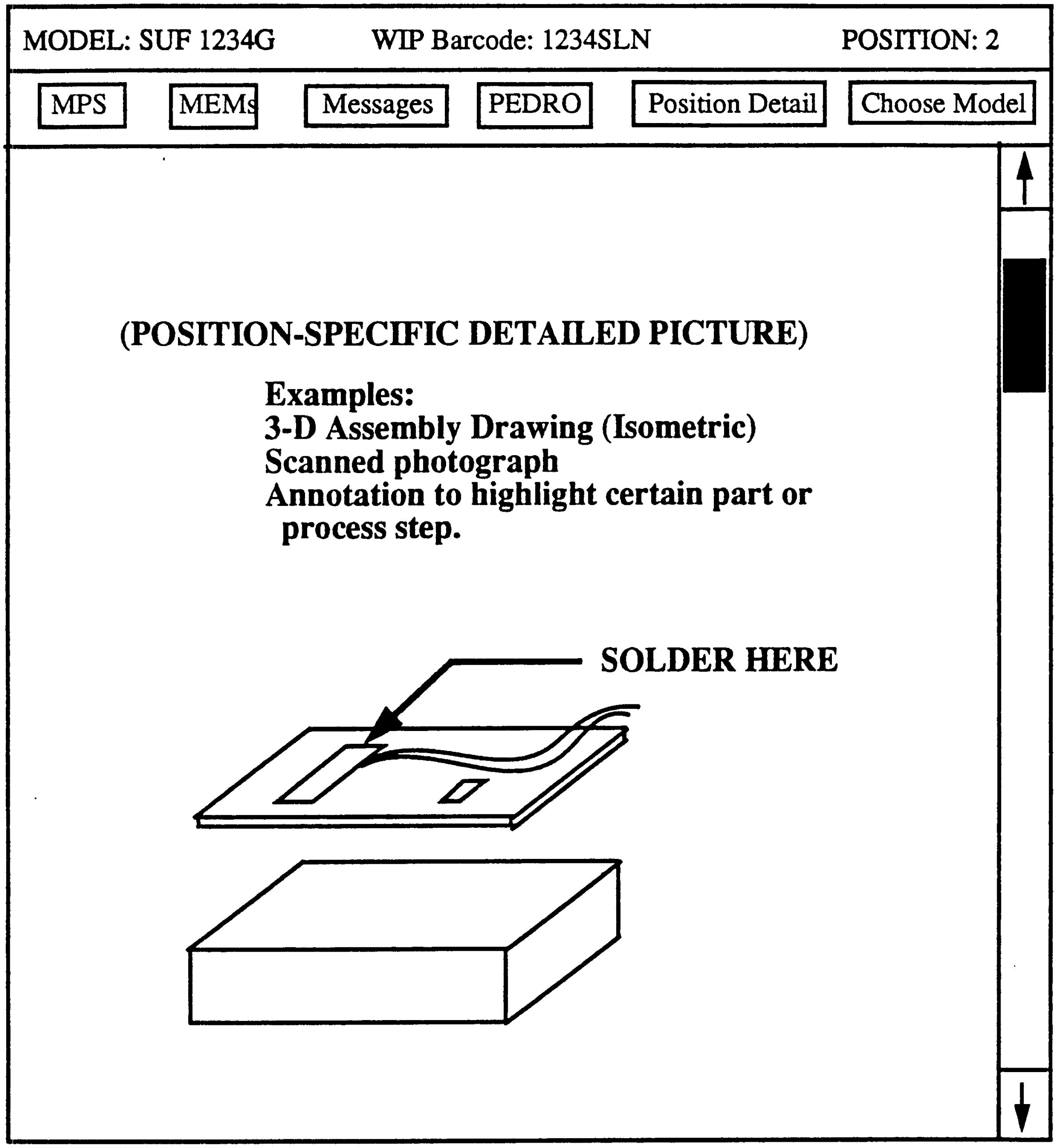


Figure C.2 Position-specific detailed picture

Appendix C: Graphical user interface (GUI) design for operator-assist stations

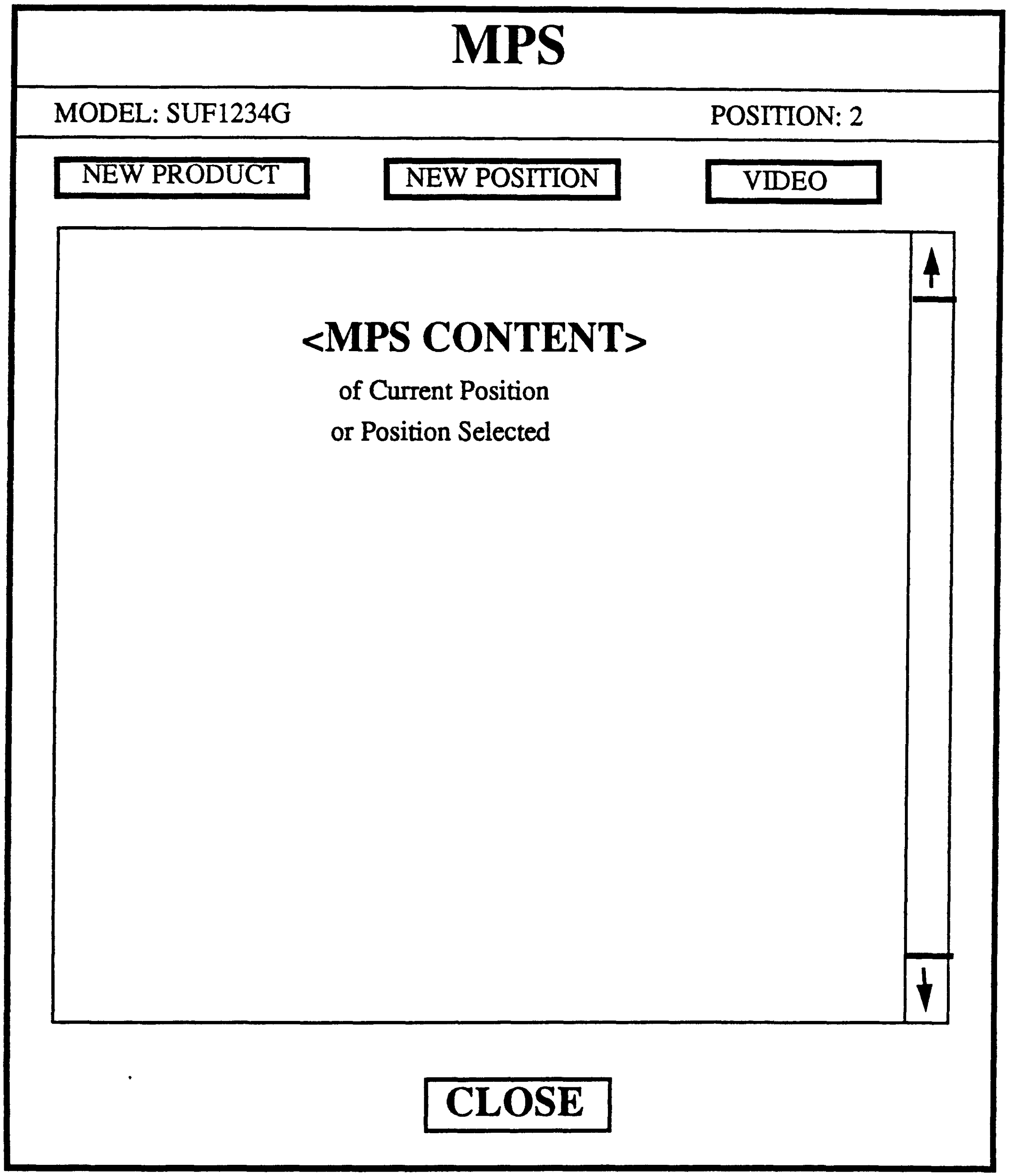


Figure C.3 Manufacturing process sheet (MPS) window to display the process steps for a selected model and position.

Appendix C: GUI design for operator-assist stations

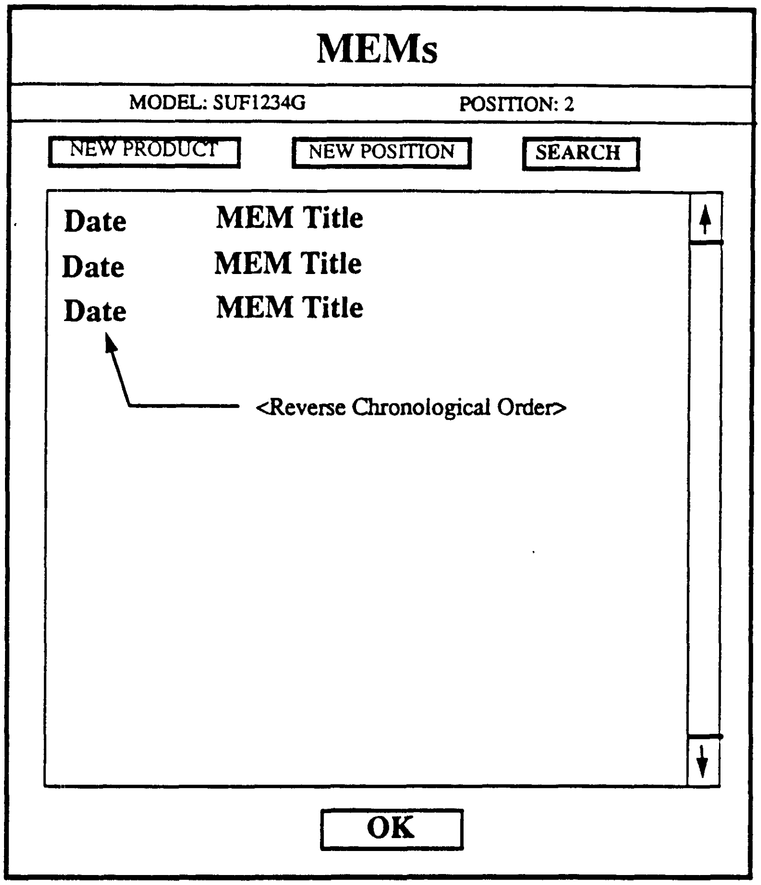


Figure C.4 Window to display manufacturing-engineering memos (MEMs), which are changes to the process steps.

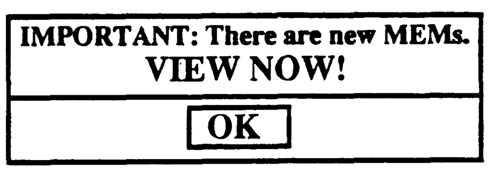


Figure C.4a Window to ensure viewing of new MEMs

Appendix C: Graphical user interface (GUI) design for operator-assist stations

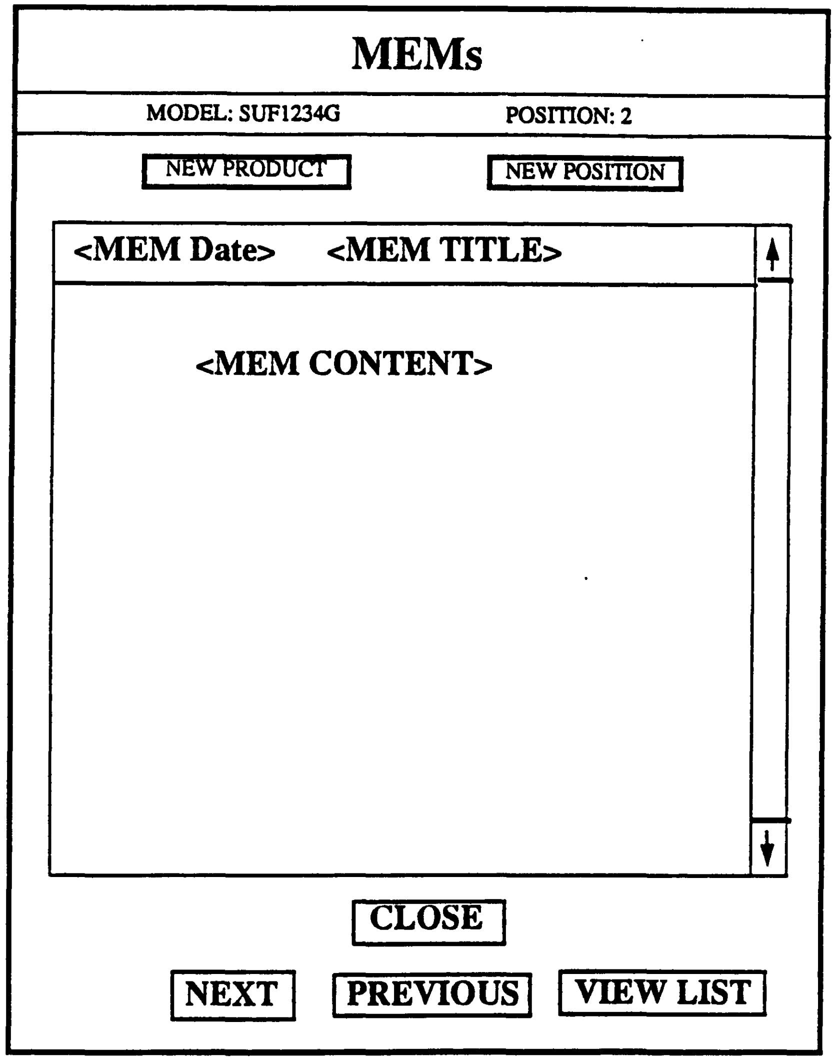


Figure C.5 MEM content window

Appendix C: Graphical user interface (GUI) design for operator-assist stations

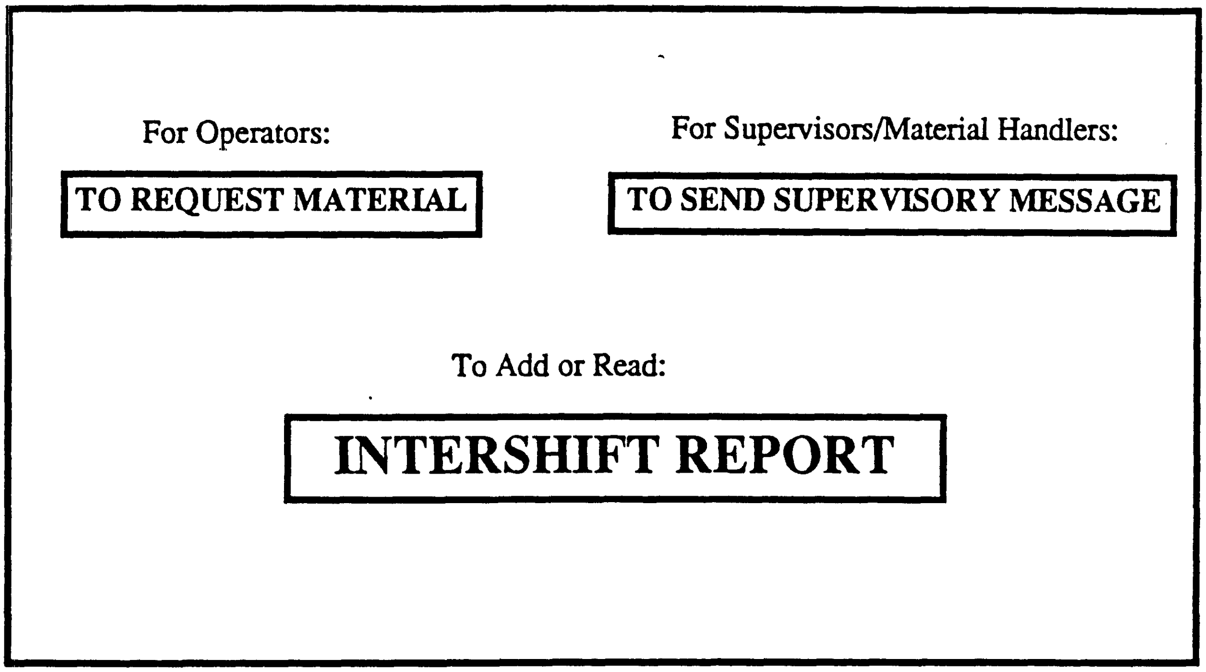


Figure C.6 Window to initiate messages and to add/read intershift report

Appendix C: GUI design for operator-assist stations

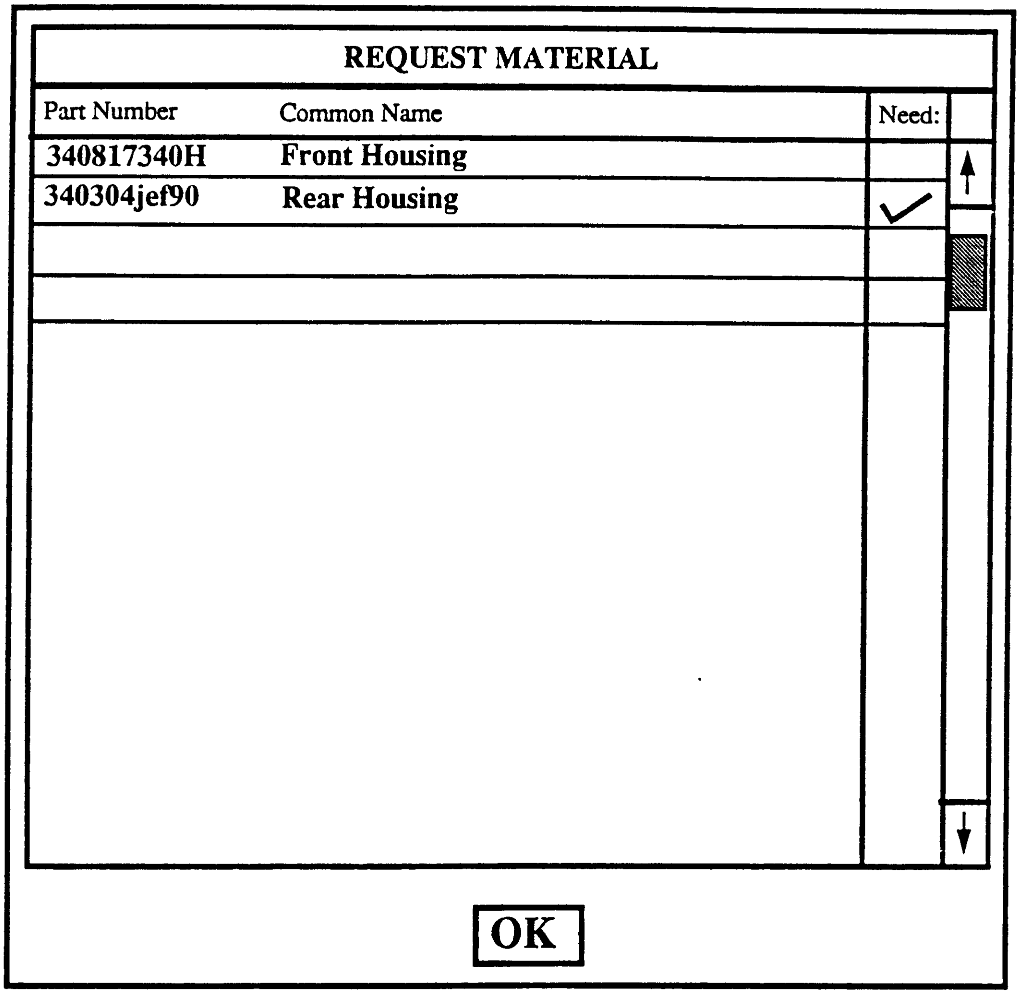


Figure C.7 Window for operators to request material

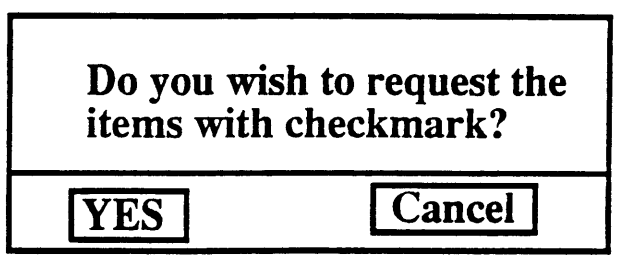


Figure C.7a: Confirmation window to request material

Appendix C: GUI design for operator-assist stations

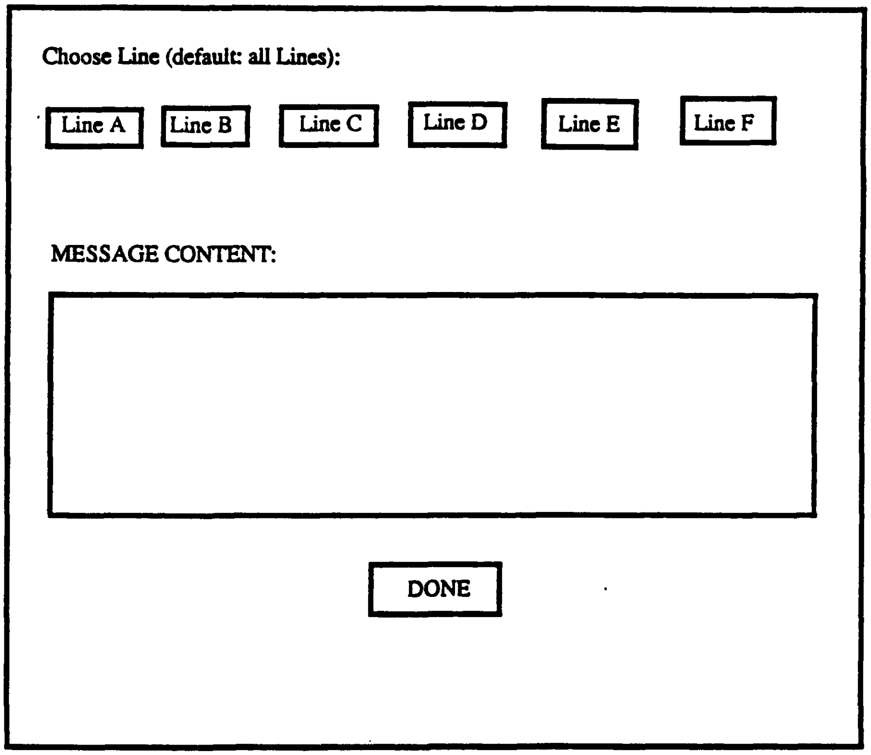


Figure C.8 Window for supervisory/material-handler message

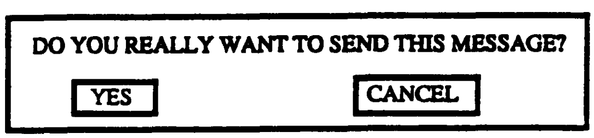


Figure C.8a: Confirmation window

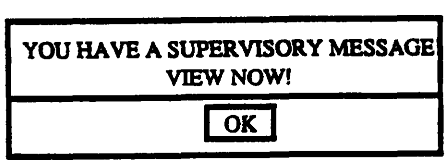


Figure C.8b: Window with beep to alert supervisory message

Appendix C: GUI design for operator-assist stations

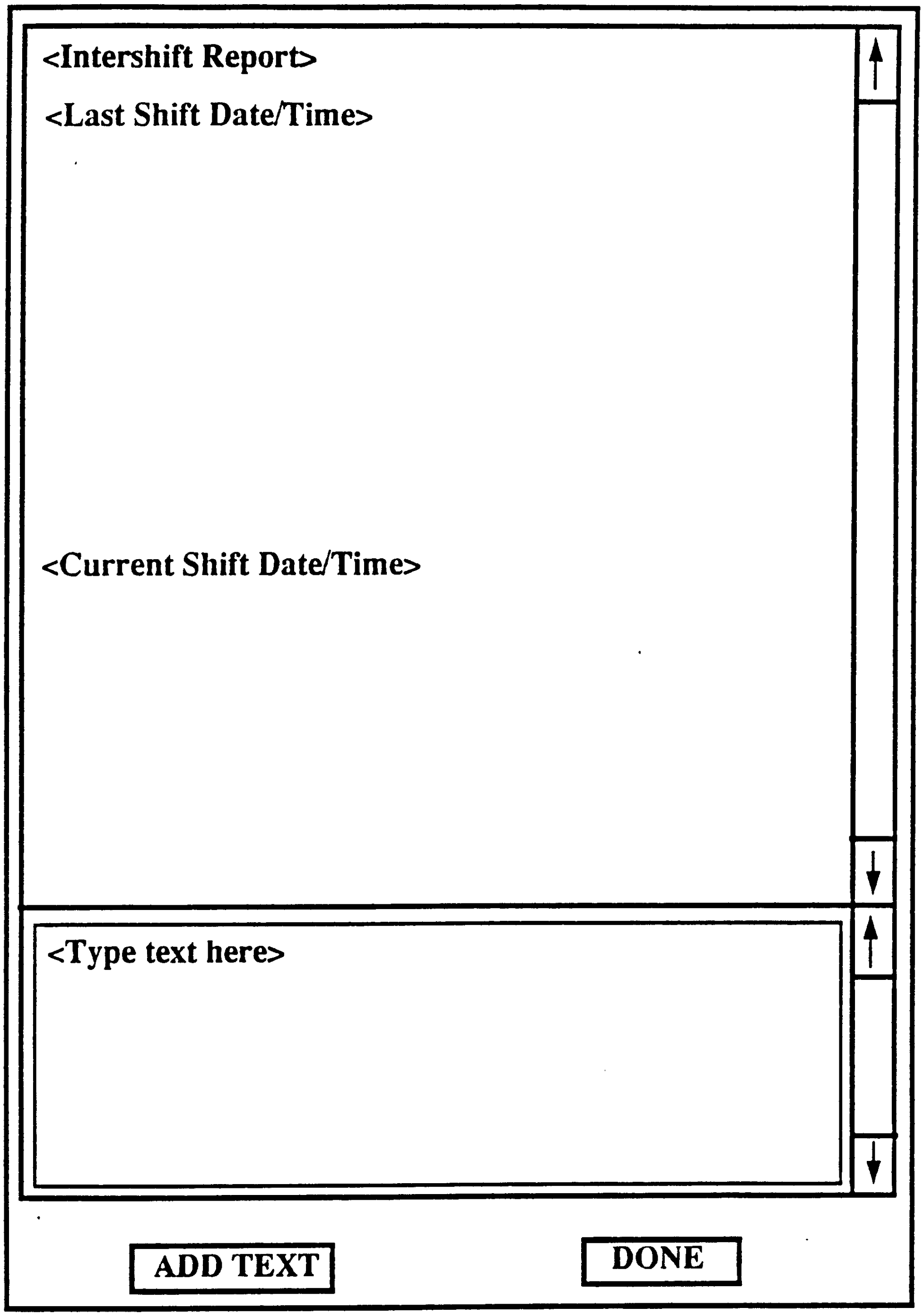
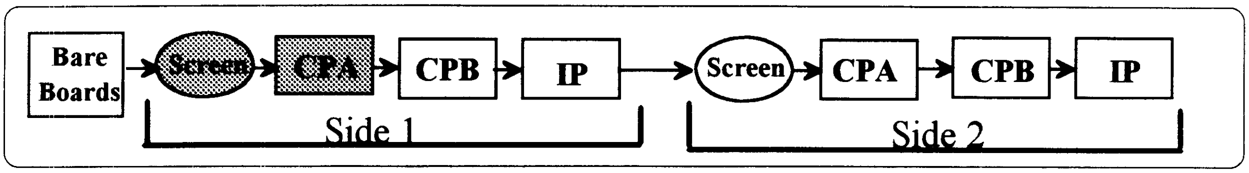


Figure C.9 Window to add or read intershift report

Appendix D: Checklist for model changeover at first position

CHECKLIST for MODEL CHANGE



Position: Side-1 Screener / CPA

At **BEGINNING** of shift:

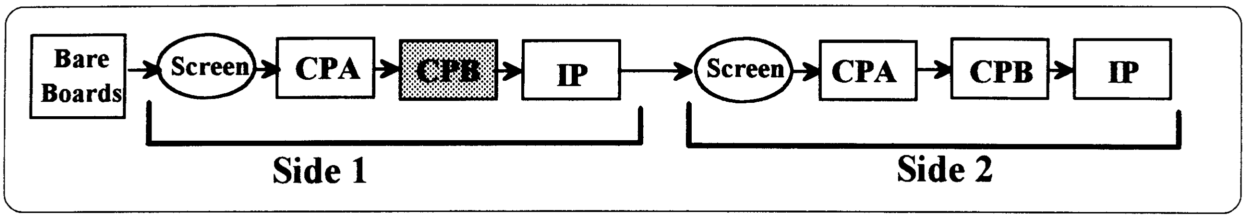
- look up production schedule board
- check that CP machine has all the programs for the scheduled models
- figure out current model's WIP (from last shift) at stacker, inspection, & tune. Subtract this from what is on production schedule
- Get boards; ensure there's enough for current model
- Tell all other operators what models will be made for the shift, **or if there are changes**.
- Remember that Logic IP operator needs to know model change at least 1 hour before you change so that he/she can prep the EPROMs.

DURING the shift:

1. Get boards from materials when there's about 50 panels left of old model.
2. Label first 3 boards with proper new SLN model number. Write on both sides at "breakaways" of board--4 places ea. side.
3. Find the next stencil (if stencil change needed).
4. Screen the last 5 board panels of old model ahead of time before stencil change (so that you can keep feeding the CP machine while changing stencil.)
5. Change stencil & wipe clean; level squeegees; add paste.
6. Start screening; check quality of process. (Remember to keep feeding the CP machine with old-model boards.)
7. If CP-program needs to be changed, hit *Sequence Stop* (F6 Button) at CP machine when last board of old model is done.
8. Change program on CP machine.
9. Start running & **communicate** to next operator of changeover.
10. Check the new model-change panel for proper placement of parts, right PC board, etc.

Appendix D: Checklist for model changeover at second position

CHECKLIST for MODEL CHANGE



Position: Side-1 CPB

At **BEGINNING** of shift:

- look up production schedule board or check with AL Screener operator to find out what models will be made for the rest of shift.
- check that your machine has all the programs for the scheduled models

DURING the shift:

1. When person who operates Screener & CPA tells you of model change, look up model matrix to see if you need to change program. If so, then:
2. Wait until board from current model leaves your machine.
3. Hit *Sequence Stop* (F6 Button), change program.
4. Start running new model.
5. Put the IP conveyor switch from *Pass Through* to *Inspect* position so that the new model-change panel does not move on to IP unless the IP operator is ready for change.
6. **Communicate** model-change information to next operator.
7. Check the new model-change panel for proper placement of parts, right PC board, etc.

NOTE: **Good communication** amongst all operators is critical for successful model changeovers.

Bibliography

Barker, C., "Dynamic presentation of customer order specific build documentation demand," presented at *Motorola Advanced Manufacturing Technology Symposium,* Schaumburg, IL, Summer 1993.

Center for Quality Management, *Concept Engineering: the key to operationally defining your customer's requirements,* document no. 71, Cambridge, 1992.

He, S., Kitahashi, T., Abe, N., "Understanding mechanical assembly illustrations — obtaining assembly information from auxiliary lines showing assembly relations." *Systems and Computers in Japan*, v. 23 no. 9, 1992, pp. 88-104.

Jordan, W., Graves, S., "Principles on the benefits of manufacturing process flexibility." working paper at Massachusetts Institute of Technology, WP33296-91-MSA, May 1991.

Kawakita, J., *The original KJ method,* Kawakita Research Institute, Tokyo, 1988.

Lach, A., "Electronics graphical build aids, an overview." *CIM Vision* (Motorola internal publication), v. 1 no. 1, July 1993, pp. 1-8.

Morse, R., "Taking a journey that leads to automated assembly instructions." *Industrial Engineering,* August 1992, pp. 38 - 43.

National Research Council, "Improving engineering design: designing for competitive advantage," National Academy Press, Washington, DC, 1991.

Pine, J., *Mass customization: the new frontier in business competition,* Harvard Business School Press, Boston, MA, 1992.

Pfleeger, S.L., *Software Engineering: the production of quality software, 2nd edition,* Macmilllan Publishing Co., New York, 1991.

Schmenner, R., *Production/Operations Management, 5th edition,* Macmillan Publishing Co., 1993

Schonberger, R., *Japanese manufacturing techniques*, The Free Press, New York, 1982.

Shiba, S., *Step by step KJ method: a problem solving tool for organizing qualitative data,* unpublished manual used to teach the KJ method at MIT and at the Center for Quality Management, 1992.

Smith, P., Reinersten, D., *Developing products in half the time,* Van Nostrand Reinhold, New York, 1991.

Thayer, R.H., Dorfman, M., eds., *Tutorial—Systems and software requirements engineering*, IEEE Computer Society Press, Los Alamitos, 1990.