Improving Equipment Performance Through Queueing Model Applications

by
Michael J. Capelle

B.S. Computer Engineering, University of Illinois - Urbana, 1990

Submitted to the Sloan School of Management and the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of

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

at the
Massachusetts Institute of Technology
May 1995

©1995 Massachusetts Institute of Technology, All rights reserved

Signature of Author _____
Sloan School of Management
Department of Electrical Engineering and Computer Science

Certified by _____
Associate Professor Charles H. Fine, Thesis Advisor
Sloan School of Management

Certified by _____
Professor Lionel C. Kimerling, Thesis Advisor
Department of Material Science Engineering

Accepted by _____
Jeffrey A. Barks, Associate Dean
Sloan Master's and Bachelor's Programs

Accepted by _____
Frederick R. Morgenthaler, Chairman
Department Committee on Graduate Students

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY
JUN 20 1995
LIBRARIES
Barker Eng

Improving Equipment Performance Through Queueing Model Applications

by
Michael J. Capelle

Submitted to the Sloan School of Management and the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of

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

Abstract

To manufacture its most advanced microprocessor products, Intel Corporation is investing significantly in state-of-the-art semiconductor processing equipment. With such large equipment costs, tremendous savings can be reaped by effectively managing its operations to reduce the required amount of capacity. In particular, equipment utilization is a powerful leverage point for avoiding additional capital procurement. The higher the equipment utilization level is increased, the less equipment that is required. The caveat is that increased utilization may accidentally cause a non-bottleneck operation to become a constraint in the process flow. An operations management analysis can provide insight into addressing this problem.

A queueing model framework was applied to one of Intel's Sort tester areas as a case study for understanding the utilization problem. Intel's microprocessors require incredible advances in tester equipment which does not come without its costs. Intel is anticipating significant capital expenditures on test equipment and is seeking the most efficient use of this expensive resource. Low tester utilization levels create an opportunity to understand what operational systems are needed to increase tester utilization and reduce future capital investment. Since tester improvement efforts have typically been approached from an engineering dominated standpoint, the operations perspective provides new insight into the root causes of the problem. The analysis points to the two fundamental factors limiting increased utilization and provides recommendations for addressing them. An implementation model for the specific Sort area under study is also presented.

Thesis Advisors
Associate Professor Charles H. Fine, Sloan School of Management
Professor Lionel C. Kimerling, Material Science Engineering Department

The author gratefully acknowledges the support and resources made available to him through the MIT Leaders for Manufacturing Program, a partnership between MIT and major U.S. manufacturing companies.

Acknowledgments

I am very grateful to Morgan Burke for his efforts in making this project a reality. Not only is he responsible for initiating the internship, but Morgan continually provided helpful advice to help focus my efforts.

There are many other people I would like to recognize as well:

- Roger Cook for his valuable support throughout, and his patience while I developed my thesis topic.
- Fred Melkey, Vicky Marsing, and Tom Moore for their help and insight.
- the Sort 9/11 organization, especially Jeff Berger and Grant DuCote.
- everyone who took the time to speak with me at the Rio Rancho site, A4/T11, Fab 6, and D2 Sort.
- my advisors, Professors Charlie Fine and Kim Kimerling, for their assistance and guidance.

Also, I would like to thank my parents for their support during these two years, and especially my wife, Amy, for her patience and endurance through this time apart. Finally, I dedicate the efforts of my work to the memory of my grandmother.

☮

Bibliographical Note on Author

Michael J. Capelle graduated from the University of Illinois - Urbana, in June 1990 with a Bachelor of Science degree in Computer Engineering. He proceeded to work for three years as a VLSI Design Engineer at Intel Corporation in Chandler, AZ, and then enrolled in MIT's Leaders for Manufacturing program with Intel sponsorship. After graduating from MIT in June 1995, Mike will be returning to work in Chandler, AZ, at Intel's Fab 12 semiconductor manufacturing facility.

Table of Contents

List of Figures

1. Introduction and Overview

1.1 Company History

Intel Corporation was founded in 1968, developing its first site in the burgeoning Silicon Valley area south of San Francisco, CA. Although its first market successes came from its memory and microcontroller businesses, IBM's decision to use Intel's 8088 microprocessor in 1978 for its first personal computer marked the beginning of Intel's current success. The demand for more powerful computers has driven Intel's microprocessor design and fabrication processing technology through several generations since the 8088 was introduced. The most recent products resulting from these efforts are the 80386 (1985), 80486 (1989), and the Pentium™ (1993) microprocessors. Intel has also moved into producing PC system "mother" boards as well as complete PC systems. Both of these extensions down the value-chain provide vehicles to further promote the Intel architecture. All of these advanced technology products helped Intel earn record 1994 revenues of $11.5 billion, making it the largest semiconductor producer in the world.

1.2 Project Overview

This project was developed based on several interviews within Intel's Albuquerque, New Mexico, and Chandler, Arizona, manufacturing organizations. Although a variety of issues were uncovered, capital avoidance for Intel's newest facility, Fab 11 in Albuquerque, NM, was the most pressing, since its planned expenditure exceeds $1 billion. Of the numerous equipment groups being installed, the cost of the Sort tester area rivals the cost of the most expensive equipment sets. Since the same test equipment is also used for final Test at other Intel sites, testers are the largest capital expense item at Intel. Many resources have been applied to improving tester efficiency, especially from a technical perspective. This results in efficient use of the tester when it is processing material, but significant improvement can be made in the area of equipment utilization. Instead of the traditional engineering focus, an operational perspective was more appropriate for addressing the historically low tester utilization levels. To provide a framework for analyzing the Sort operations, a queueing model from the literature was applied. Sort is an excellent

candidate for a queueing model since it is not a hard bottleneck in the manufacturing process, it is subject to the highly variable Fab output, and it is highly capital intensive. A queueing model provides an excellent framework since it relates many operational metrics including utilization, throughput time, and product arrival variability. Although applying a queueing model is beneficial, it should be clear that a queueing model is much more useful for understanding the system rather than controlling it, and that "the literature should concentrate more on bounds and sensitivity analysis rather than absolute numbers" (Burman 4.1, 4.2). Therefore, the model applications are oriented towards finding operational improvements for the Sort tester area. It improved the understanding of performance metrics and their tradeoffs, as well as identifying the leverage points for improving the overall performance of the system. The model supported the overall goal of increasing utilization of expensive equipment without necessarily creating a hard bottleneck within the process flow. Since the majority of the processing steps in the manufacturing process are designed as non-bottleneck operations, this type of analysis and its conclusions are very applicable throughout Intel's factories.

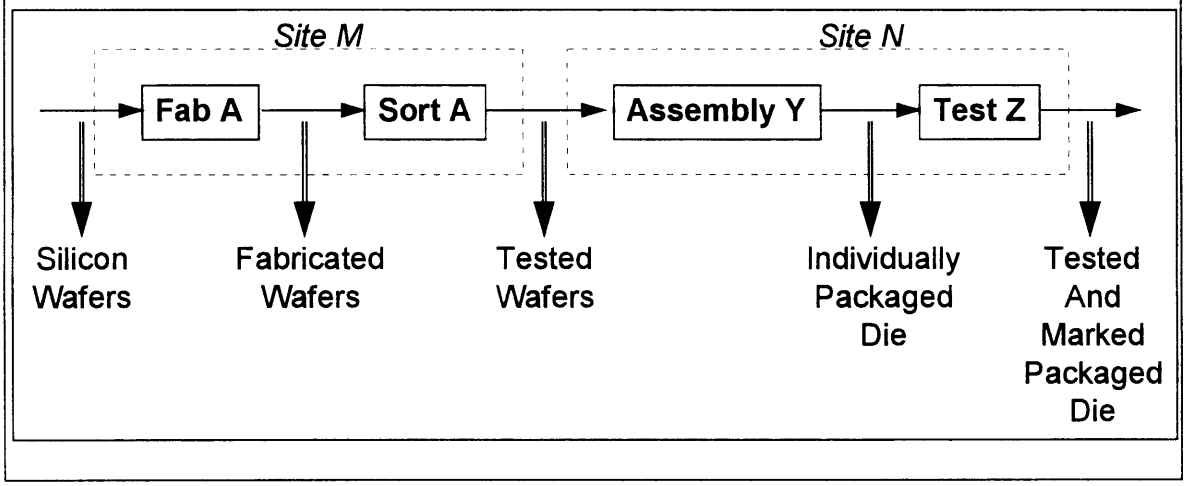
2. Intel Manufacturing Environment

Intel's high volume semiconductor manufacturing facilities are organizationally located within the Components Manufacturing Division. The facilities are geographically located at four of the five Western U.S. sites, as well as Ireland, Israel, Malaysia, and the Philippines. This chapter will first describe Intel's overall semiconductor manufacturing flow and then introduce the environment at the Rio Rancho, NM, site.

2.1 Manufacturing Process Flow

There are many manufacturing operations performed to produce a packaged semiconductor chip. Intel's typical high volume manufacturing flow covers two geographically separated sites: a fabrication site and an assembly site (Figure 1). The fabrication site can further be divided into two areas: Fab and Sort. The Fab processes a thin circular silicon wafer to produce many repeated individual die across the face of the wafer. Special test circuitry is simultaneously generated in the area between each die. This circuitry is used by downstream operations to track process performance. The wafers flow through the processing steps bundled together in a "lot" of up to 25 wafers. The manufacturing process first builds transistors at the silicon surface and then connects the transistors with multiple layers of metal interconnect. The wafers exiting the Fab have only been partially tested through in-situ monitors. At Sort, the Fab processing is more rigorously checked by two test operations. First, the test devices which were created between the individual die are tested and the results checked against normal process parameters. Out of specification material is removed from the production flow for further investigation. The second test involves checking each individual die on the wafer for functionality. When a non-functional, or "bad", die is found it is physically marked with ink on the wafer surface. These inked wafers are then shipped from the fabrication site to the assembly site.

Figure 1 - Intel Semiconductor Manufacturing Flow



The assembly site primarily packages, tests, and marks the individual die. It is composed of two organizations: Assembly and Test. Assembly receives the tested wafers from the fabrication site and places them into the Assembly Die Inventory (ADI) buffer at the head of the operations. Based on the weekly production schedule, Assembly will pull the appropriate wafers from ADI and begin processing. First, the die are individually cut from the wafer with the inked "bad" die being discarded. Then the "good" die are packaged into one of several different package types. These packaged parts then proceed to Test for final logical and performance testing. Finally, the packaged parts are marked and then shipped directly to a customer or deposited in an Intel warehouse.

2.2 Rio Rancho Site

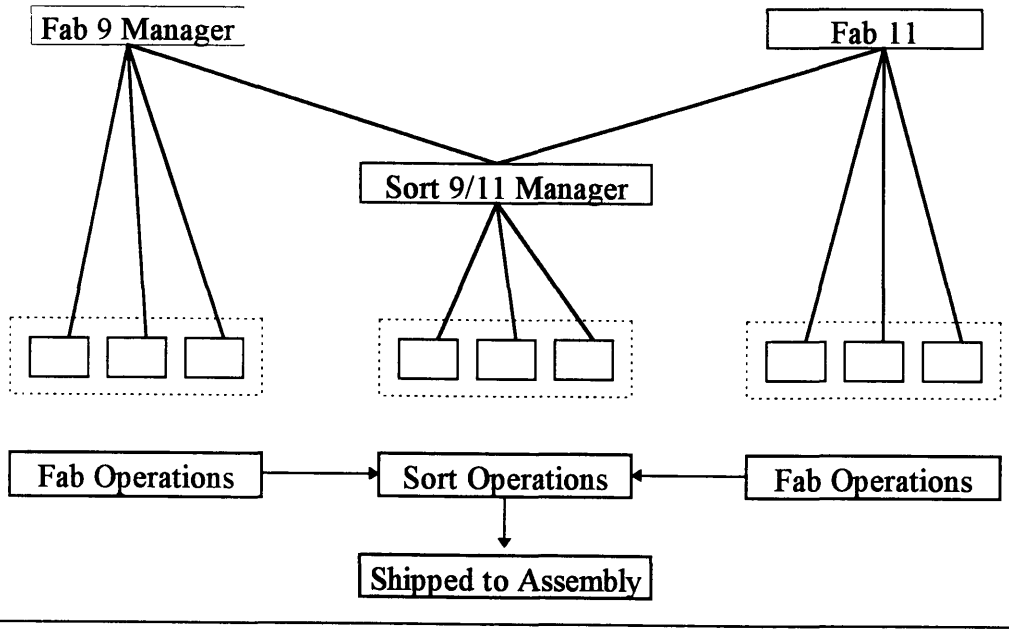
This project was carried out at Intel's Rio Rancho site located just north of Albuquerque, NM. Although most of Intel's domestic sites support a variety of activities, Rio Rancho is devoted to manufacturing. There are three independent fabrication organizations on site: Fab 7, Fab 9 and Fab 11. Fab 7 was built in the early 1980's and currently produces low margin commodity and memory products as well as Intel's new FLASH memory products. Fab 9 was completed in the late 1980's and currently manufactures many different products including high performance microcontrollers and Intel's mainstream microprocessors. Fab 11's primary facility is still under construction, yet it already has

begun small volume production of Intel's most advanced microprocessor products by using space within the Fab 9 shell. When Fab 11's main facility begins production, it will manufacture Intel's next generation microprocessor products.

2.3 Sort 9/11

The research for this project was primarily conducted within Sort 9/11. Like most of Intel's semiconductor manufacturing facilities, operations at Sort 9/11 run continuously, organized as 2 twelve hour shifts per day, seven days per week. As described previously, a production wafer completing fabrication proceeds to the Sort area for testing. While many Sort organizations are dedicated to one Fab, Sort 9/11 is responsible for processing material leaving both Fab 9 and Fab 11. Organizationally, the Sort 9/11 manager reports to both the Fab 9 and Fab 11 managers (Figure 2). The Sort manager's organization is very similar to the Fab organization having several functional groups reporting to him.

Figure 2 - Sort 9/11 Organizational Environment



The collocation of Fab 9 and Fab 11 enabled the creation of a joint Sort 9/11 floor. This arrangement creates some economies of scale for the Sort floor, as well as facilitating knowledge sharing across the different technologies. The disadvantage of this arrangement

is the loss of continuity between the Fab and Sort organizations. Having a completely separate organization interacting with two different upstream facilities removes some of the focus and continuos interaction that a one-to-one relationship provides. In examining the feasibility of a Sort 9/11 organization, the benefits of a common Sort floor overshadowed the disadvantages.

Production Planning schedules weekly Fab wafer starts and forecasts weekly Sort die-out schedules. The Sort organization holds a large part of the responsibility for meeting each plant's weekly production schedule. While Sort typically is given enough lead time to meet weekly schedules, there are weeks when late delivery from the upstream operations makes schedule fulfillment very difficult. In addition to delivery delay, the volume of material entering Sort and the tester area is highly variable. Fabs often produce large amounts of material during the course of a week. Due to the relatively inexpensive equipment Sort operated in the past, it was understandable that Sort should have enough equipment to process the occasional heavy load and not limit Fab output. Although tester costs have increased dramatically this mentality still seems to hold today.

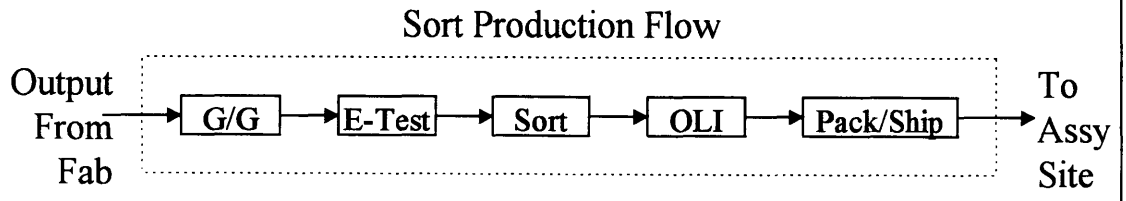
3. Sort Manufacturing

The purpose of this chapter is to discuss the Sort 9/11 process flow and develop the reasons for focusing on the tester equipment.

3.1 Process Flow

A wafer is processed through several steps within Sort (Figure 3). Wafers arrive in lots of up to 25 and are typically loaded and moved between equipment sets by human operators. Often there are WIP (Work In Process) racks before each operation to hold the incoming material. The production operators are grouped according to the processing steps and are responsible for several pieces of equipment at each processing step. The purpose of Sort is to remove out of specification wafers, provide valuable end-of-line data for the Fab, and mark non-functional die in preparation for the Assembly operations.

Figure 3 - Sort Production Flow



The backside of the wafer is received from Fab covered with an electrically insulating passivation layer (oxidized Silicon), but the backside needs to be coated with a layer of Gold to improve adhesion during subsequent packaging in Assembly, and to create a conductive layer since the bulk (substrate) of the wafer will be used as a common electrical ground for each part. The gold layer is applied by first physically grinding the backside of the wafer during the Grind operation and then applying the gold layer during the Gold operation (G/G in Figure 3). The wafer then proceeds to E-Test where the devices and structures created between the die during fabrication (in the "scribe" area) are tested. The data gathered at E-Test is compared to nominal process values with out of specification material either immediately being scrapped or being placed on hold for

further investigation. From E-Test, the lots move to the Sort operation. The Sort testers check that each individual die is functional. Die that fail are classified into "bins" based on the test which caused them to fail. The bin distribution of die on a wafer is another indicator which can cause material to be put on hold for further investigation by Engineering. The wafers then physically proceed to off-line ink (OLI) while the Sort test results are loaded into a local database. Based on the Sort test results, non-functional, or "bad", die are ink marked to allow Assembly to discriminate between die which should be scrapped and die which should be processed further. The final operations in Sort 9/11, Pack and Ship, prepare the wafers for shipment to the appropriate Assembly site.

In most Fab operations, the specific product entering an operation does not substantially affect the processing step. Therefore, the wafer processing times are nearly constant across product type. The E-Test and Sort tester operations, however, require that the equipment be setup differently for each product type. When a different product is run at these two operations, not only is hardware swapped out, but the product's software test program must be loaded. These product specific test programs can be substantially different resulting in highly variable processing times.

FIFO material processing generally holds at the various manufacturing steps although it is not strictly enforced. The exceptions involve the product specific operations (E-Test and Sort) as well as priority lots being released into the manufacturing flow. The priority lots are typically either processing experiments for Engineering or new products. A negligible number of priority lots (~1% of total volume) pass through Sort during any given week. The product types being processed at the E-Test and Sort tester operations influence the product flow as well. For example, assume that testers are setup for product A with product B already waiting to be tested. If a product A lot arrives, it will be loaded onto the equipment ahead of the waiting product B to save a setup changeover from occurring.

3.2 Sort Tester Costs

The increased complexity of Intel's most advanced microprocessors require leading edge test equipment. The advanced Sort testers are capable of providing stimulus to the large number of pins on the microprocessor products at high frequency, as well as having sufficient memory to hold the immense test programs. At $3-5 million per tester (depending on the configuration), the Sort tester equipment is as costly as the most expensive equipment set in the complete Fab/Sort process flow. During the most recent microprocessor generations, capital expenditure on Sort testers has easily exceeded other equipment sets within the Sort area. A comparison of equipment costs (in dollars per unit of capacity) for all of the Sort processes clearly shows that improvement resources being allocated solely within the Sort organization should necessarily migrate to the Sort testers. Improving tester operations would have the largest impact in future capital avoidance. Since capital avoidance for Intel's newest facility was the primary aim of this project, the Sort tester area was an excellent area in which to focus.

3.3 Applying TPM to Sort Testers

TPM, or Total Productive Maintenance, is an equipment maintenance philosophy developed in Japan based on the U.S. concept of productive maintenance. The basic principle is "that equipment improvement must involve everyone in the organization, from line operators to top management" (Nakajima, Development 2). Its key innovation is that "operators perform basic maintenance on their own equipment" (Nakajima, Development 2). The goal of TPM is to increase equipment effectiveness so each piece of equipment can be operated to its full potential and maintained at that level. To maximize equipment effectiveness, three equipment characteristics need to be maximized: the equipment's total availability, the equipment performance rate, and the number of quality products produced by the equipment. Since each of these three characteristics can be quantified, a single unified metric, Overall Equipment Effectiveness (OEE), can be calculated:

Overall Equipment Effectivenss = Availability × Performance Rate × Quality Rate

where,

$$\text{Availability} = \frac{\text{Total Time} - \text{Down Time}}{\text{Total Time}}$$

$$\text{Performance Rate} = \left(\frac{\text{Theoretical Cycle Time}}{\text{Actual Cycle Time}}\right)\left(\frac{\text{Total Output} \times \text{Actual Cycle Time}}{\text{Total Time} - \text{Down Time}}\right)$$

$$\text{Quality Rate} = \frac{\text{Good Output}}{\text{Total Input}}$$

A slight mathematical modification to the OEE equation helps analyze its implications:

Given:

$$\text{Utilization} = \frac{\text{Output} \times \text{Actual Cycle Time}}{\text{Total Time}}$$

$$\text{Speed Factor} = \left(\frac{\text{Theorectical Cycle Time}}{\text{Actual Cycle Time}}\right)$$

$$\text{Yield} = \text{Quality Rate}$$

Then:

$$\text{Performance Rate} = \text{Speed Factor x}\left(\frac{\text{Utilization}}{\text{Availability}}\right)$$

$$\text{OEE} = \text{Availability} \times \text{Speed Factor} \times \left(\frac{\text{Utilization}}{\text{Availabilitiy}}\right) \times \text{Yield}$$

$$\therefore \textbf{OEE} = \textbf{Utilization} \times \textbf{Speed Factor} \times \textbf{Yield}$$

The OEE equation points to three broad areas in which to improve Sort tester effectiveness: speed, yield, and utilization. Of these three areas, the operational aspects embodied in the utilization component do not receive as much attention as the other two areas, and, in general, are not as well understood. The speed component is primarily focused on what occurs when WIP is being processed by the tester. This is an area which already receives much attention from the Intel engineering community. Not only have

there been improvements on the Intel side, but the equipment suppliers themselves are motivated to make improvements to increase the marketability of their product. The other two areas, yield and utilization, are impacted through Intel's operational policies. In Sort, yield indicators contain a relatively large component due to the Fab's processing variability and a smaller component due to the actual Sort operations. There is strong leverage in increasing Sort yields since the material coming from the Fab has used up the vital resources of the factory constraint and any yield improvement would directly impact overall manufacturing capacity. This philosophy needs to be understood by the Sort organizations so that their particular yield issues receive the necessary amount of emphasis. The final component of tester effectiveness, utilization, has probably received the least amount of analysis, yet its associated indicators are the most highly tracked. Modeling this area would improve the understanding of the indicators as well as which operational components have the most leverage in affecting those indicators.

3.4 Sort Tester Utilization

The utilization metric is commonly referred to as the percentage of time that a piece of equipment is processing material, or arithmetically:

$$\text{Utilization} = \frac{\text{Time Processing Material}}{\text{Total Time}}$$

For the Sort testers, the average weekly utilization equation becomes:

$$\text{Average Weekly Utilization} = \frac{\text{Number of Lots} \times \text{Average Lot Processing Time}}{\text{Number of Testers} \times 168\text{hrs/week}}$$

A couple of points should be made regarding this equation. First, of the three variables, a stable, mature Sort floor can only affect utilization by changing the number of testers it operates. The other two variables are determined by what the Fab outputs, which Sort does not currently influence. Second, many Sort improvement projects reduce the average lot processing time. These are referred to as test time reduction (TTR) programs.

Obviously TTR programs are beneficial, but it must be understood that the improvements will have a negative affect on the utilization metric. Although Engineering is rewarded for implementing the TTR programs, production may be wrongly reproached for the decrease in utilization. More appropriate responses would occur if the utilization metric was better understood.

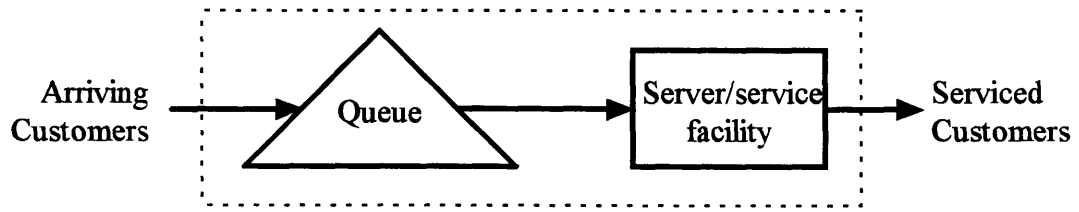
4. Queueing Theory

The purpose of this chapter is to introduce queueing theory, demonstrate how it can be applied to the Sort testers, and perform sensitivity analysis of some key queueing theory parameters. The purpose of applying a queueing model was to provide a framework for analyzing the operational aspects of the Sort tester area and to improve the understanding of the utilization metric.

4.1 Queueing Theory Overview

Queueing theory attempts to describe characteristics of the two element system depicted in Figure 4 (Fine and Wein 1). The system involves customers arriving at a server to be processed. Some examples of a server are a post office worker, the Golden Gate bridge, and widget production equipment. For those server examples, the customers would be (respectively) people desiring the service of the post office, cars wanting to cross the entrance of San Francisco Bay, and widget raw material. A queue is formed when items arrive at a service center and the servers are busy. Queueing theory attempts to describe system characteristics, such as line length and waiting time, based upon the interarrival and service time distributions and the queue type.

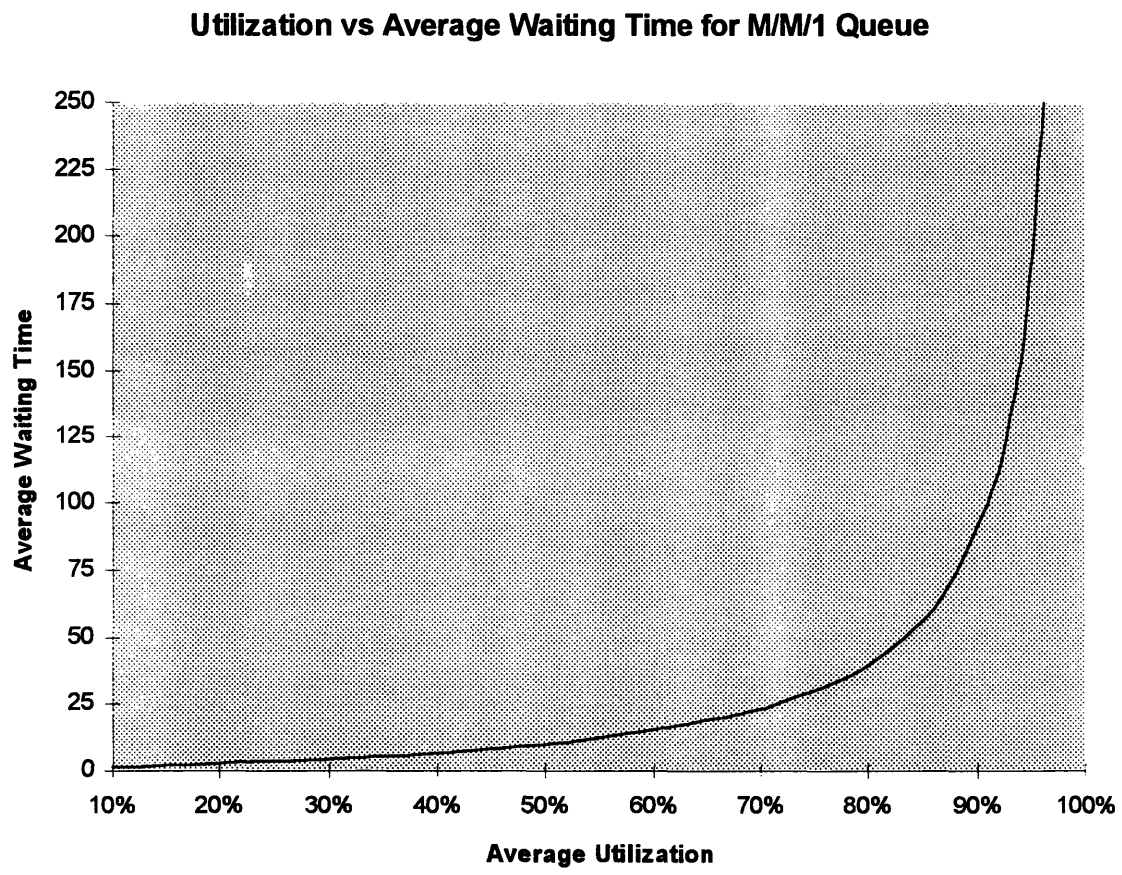
Figure 4 - A Simple Queueing System



System utilization is a key parameter in a queueing system. Utilization is defined as the probability that all servers are busy or, alternatively, that an incoming customer will have to wait. The relationship between average system utilization and average customer waiting time for a typical queueing system is shown in Figure 5. At low system utilization, waiting time is near zero. Therefore, the probability that all of the servers are busy when a

customer arrives is very low, and, on average, a customer will proceed to a server in a very short amount of time. As system utilization increases, waiting time increases exponentially. Given a system with stochastic interarrival and service times, 100% utilization can only be attained at the cost of infinite waiting time. A plot of queue length would show the same asymptotic relationship with system utilization.

Figure 5 - Utilization vs Waiting Time



4.1.1 Interarrival and Service Time Distribution

To specify a queuing system, the time between customer arrivals and the time to service customers are two statistical processes which need to be known. Each process can be either a predictable or an unpredictable process (Kleinrock 1: 4). If an arrival process is predictable, the amount of time before another customer arrives is precisely known. If a service process is predictable, the amount of service time remaining once a customer enters a service facility is precisely known. The analysis of a completely predictable system is not as analytically complex as a system subject to stochastic variability. "The assumption in most of queueing theory is that these interarrival times are independent, identically distributed random variables" (Kleinrock, 1: 8). It is the analysis of unpredictable systems that best reflects real world problems and thus dominates queueing theory research.

For queueing theory to be most applicable, closed-form solutions describing the queueing system's waiting time and queue length are necessary. To generate closed-form solutions for particular queue types, simplifying assumptions regarding the interarrival and service time distributions are required. When working with stochastic arrival processes, a common simplification is to assume that arrivals follow a Poisson process. This process "enjoys a number of ... simplifying analytical and probabilistic properties" and the process does in fact model numerous natural physical and organic processes (Kleinrock, 1: 61). The Poisson process is fully specified by one parameter, λ, the average arrival rate of customers. Given a Poisson arrival process with parameter λ, by definition the interarrival times will be exponentially distributed with parameter λ (Nahmias 684-686). The average interarrival time will then be $1/\lambda$. A characteristic of both the Poisson and Exponential distributions that simplifies analysis is that the variability measures of the distributions are related to the mean of the distributions. For the Poisson process, its variance is equal to its mean. For the Exponential process, its standard deviation is equal to its mean. The Exponential distribution also has an interesting property commonly referred to as its "memoryless" property. Given exponentially distributed interarrival times, knowing how much time has passed since the last arrival does not affect the estimate of when the next arrival is to occur, it is still λ, the average interarrival time. When modeling completely

random arrivals, assuming exponentially distributed interarrival times will usually suffice, but there are cases when erlangian, hyperexponential, or another distribution may be more accurate. Again, making a simplifying assumption regarding the interarrival or service time distributions allows for a well-behaved mathematical representation to be derived.

4.1.2 Queue Types

There are many factors distinguishing the different "types" of queueing systems. Maister divides these factors into 7 categories (13):

1) Number of Successive Stages
2) Number of Channels
3) Queue Discipline
4) Range of services offered by each serving facility
5) Behavior of customers in deciding whether to join line
6) Behavior of customers in deciding whether to stay in line
7) "Types" of customers

The number of successive stages are segregated into two general partitions: Single Stage and Multiple Stage Systems. A Single Stage System contains only one queue and one server. Alternatively, a Multiple Stage System contains several queue/server systems connected serially so that customers must proceed through one queue/server pair to another queue/server pair based on a predetermined order. The Multiple Stage System can be thought of as many Single Stage Systems connected together.

The number of channels simply describes whether there are individual queues for each server (Singlechannel) or whether there is a single queue the customer enters and is subsequently routed to an available server (Multichannel). The Parallel Single Stage System is composed of multiple Single Stage Systems in parallel so that the customer must choose which queue to join upon arrival. Conversely, the Multichannel Single Stage System contains multiple servers but with only one queue feeding those servers.

The queue discipline describes the order in which the arriving customers are served. "Queue disciplines are one of the prime managerial tools available to affect the behavior of the system" (Maister 5). Applying some simple policies to the queueing system can greatly enhance system performance. Probably the simplest and most common discipline is to service the customers in the order in which they arrive, otherwise known as FIFO (First-In, First-Out). Opposite of FIFO is Last-In, First-Out, or LIFO. Other disciplines concern the existence of one or more priority levels for incoming customers. The rules regarding priorities usually involve servicing the higher priority customers before any lower priority customer. One final discipline of note is the Multi-line system. A good example of this discipline in practice is the four queue system at a street intersection, where traffic light duration is the primary means available to increase system performance must be determined.

The range of services provided by each server impacts the queueing system as well. By only having certain services available, customers are discriminated against based on their needs. Specialization by individual servers may substantially increase individual server performance and correspondingly increase overall system performance.

Customer behavior also affects queueing system performance. There are two decisions that a customer may be able to make: (1) whether to join the queue at all and (2) whether to remain in the queue that was joined. The first decision may depend on the line length, since the potential customer may not desire to join a long line. The second decision may depend on the queue waiting time, since a frustrated customer waiting in line for a long period of time may have the ability to leave a queue.

Finally, being able to categorize different types of customers and understanding the arrival distributions of the individual categories is important to optimizing system performance. This queue type descriptor is closely related to the range of services provided by each server. If customer categories can be determined, then servers can be optimized to address

the needs of a specific customer category. Server specialization can improve overall system performance.

4.1.3 Notation

A simplified notation has been developed in referring to different queue types and the corresponding model assumptions regarding interarrival and service time distributions. The notation has three parameters indicating the arrival distribution, the service distribution, and the number of servers. M, D, E_k, H_k and G denote the special distributions: exponential, deterministic, Erlang with k phases, hyperexponential (mixture of k exponentials) and general, respectively (Whitt, Approximations 119). The simplest nontrivial system is M/M/1. This notation represents a queueing system having exponentially distributed interarrival and service times and one server. Often times, many assumptions regarding the queue type need to be made in order to apply a queueing model. Some of the typical assumptions are that the processes are independent of each other and independent of the number of customers in the system. Also, unless otherwise mentioned, there is unlimited line length, no priority scheme, and the system is run FIFO.

Finally, regarding notation, there are many variables commonly used in the context of queueing systems:

λ - average arrival rate - the average rate at which customers arrive

μ_a - average interarrival time ($= \lambda^{-1}$) - the average time between arrivals

σ_a - standard deviation of interarrival time - a measure of the interarrival time variability

c_a - coefficient of variation of interarrival time $\left(=\frac{\sigma_a}{\mu_a}\right)$ - a measure of interarrival variability normalized to the mean interarrival variability

μ_s - average service time (also denoted τ) - the average time it takes to service a customer

σ_s - standard deviation of service time - a measure of the service time variability

c_s - coefficient of variation of service time $\left(=\frac{\sigma_s}{\mu_s}\right)$ - a measure of service time variability normalized to the mean service time

m - the number of servers

ρ - utilization $\left(=\frac{\lambda\mu_s}{m}\right)$ - the probability that a server will be busy

4.2 Application to Sort Testers

Whitt's GI/G/m model provided the framework for the Sort tester analysis (Whitt, Approximations). The GI/G/m notation signifies that the arrival and service processes are general distributions, independent of each other, with m number of servers. Whitt's model can be fully described using five of the previously listed parameters: μ_a, c_a^2, μ_s, c_s^2, and m. There were several reasons for choosing this model:

- Closed-form solution
- Interarrival time variance was a variable
- Service time variance was a variable
- Multiple servers
- Simplifying assumptions didn't invalidate its use

The analysis uses wafer lots as the arriving customers, and the Sort testers as the servers. The incoming lot is an aggregate lot having the weighted average characteristics (by product) of all the incoming lots. In examining preliminary production flow data, it was clear that the mean and standard deviation of the two system processes (interarrival and service) were far from equal, so assuming an exponential distribution was very inaccurate. Also, the model was intended to give some insight into the effects of variability changes, so using a distribution that tied the variability to the mean would not permit that analysis. The model assumes that the distributions are sufficiently specified by their first two moments (their mean and variance). The model also assumes FIFO customer flow and that all of the testers are equally equipped to process any incoming lot. The FIFO assumption is probably the largest assumption violation. Incoming lots are not placed on the WIP racks in order so the operators can pull lots off of the rack in any order. This results in some lots waiting for a longer period than a model would predict and other lots not waiting as long as a model would predict. This increases the variance of the waiting time distribution, but the average waiting time predicted by a model would still be accurate.

When modeling complex systems, simplifying assumptions need to be made. These assumptions should not detract from the queueing model accuracy to such an extent that conclusions being drawn are misleading. Although the validity of the assumptions can be argued, the queueing model output was consistent with the basic tenets of queueing theory. As stated earlier, the purpose of the model was to provide a framework for analyzing the operational aspects of the Sort tester area and to improve the understanding of the utilization metric. The model was successful in fulfilling its purpose.

4.2.1 Arrival Process

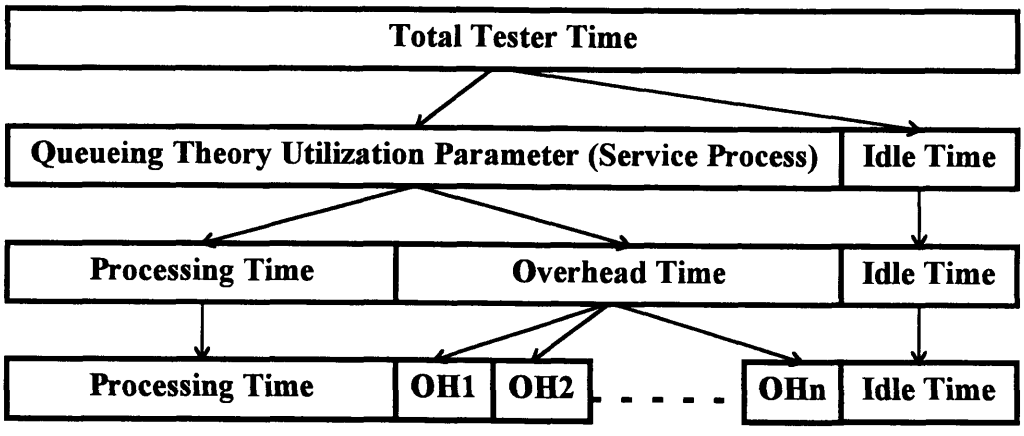
Developing the arrival process is relatively simple. With data regarding the exact arrival time of each incoming lot, every interarrival time can be calculated. Statistical analysis of the interarrival times would reveal the mean and standard deviation of the interarrival times. These two descriptive statistics are directly used by Whitt's model.

4.2.2 Service Process

Queueing theory applications described in the literature typically assume that if a queue is forming, it is due to the servers being busy with customers. In working with equipment in a manufacturing setting, one realizes that there are other occurrences which cause a queue to form. These occurrences range from inoperable, or "down", equipment to required operational activities. All of these activities effectively consume capacity and need to be considered when determining a service process.

In applying Whitt's queueing model to Sort 9/11, the service process was constructed from two underlying activities, the *processing* activity and the *Overhead* activities (Figure 6). The Overhead activities simply cover all of the other activities occurring on the testers other than the processing of the incoming lots. The relationship to queueing theory's utilization parameter is fairly straightforward. Utilization is the probability that an incoming customer will have to wait. Therefore, the percentage of time that is spent either on processing or on Overhead activities is analogous to utilization as it is defined in queueing theory. Breaking down tester time further, total Overhead Time is composed of several individual components representing the non-processing activities occurring on the tester. Each of these Overhead components needs to be defined such that they are mutually exclusive and sum to the total Overhead Time.

Figure 6 - Total Tester Time Components



Processing Time

The processing time distribution can be calculated based on the product mix being run. Each product has a die yield dependent average test time. Die yield (DY) is the number of good die on the wafer after Sort functional testing, expressed as a percentage of total good die possible on the wafer. There is a linear relationship between DY and wafer test time. (This is discussed further in section 5.1.1.) Based on the product's die yield, an average wafer test time can be calculated. Weighting these average test times by the proportional number of lots entering the area, a probability distribution can be generated. The mean of this distribution is the average lot processing time, and the standard deviation of this distribution is the standard deviation of lot processing times. This is summarized mathematically below:

$$\mu_p = \sum_{i=1}^{n} \frac{l_i}{L} \mu_{p_i}\{dy_i\}$$

$$\sigma_p^2 = \sum_{i=1}^{n} \frac{l_i}{L} \left(\mu_{p_i}\{dy_i\} - \mu_p\right)^2$$

Where: μ_p = average lot processing time

σ_p^2 = lot processing time variance

dy_i = die yield of product i

$\mu_{p_i}\{dy_i\}$ = average lot processing time for product i (as a function of its die yield)

l_i = number of lots of product i

L = total number of lots

n = total number of products

This method of generating a distribution neglects the inter-product wafer-to-wafer variability. This assumption is justified when the spread of the average wafer processing time is much larger than the spread of any of the individual products wafer-to-wafer variability. This is true for the products at Sort 9/11.

Overhead Time

The mean Overhead Time is simply calculated by adding up the average amount of time each of the Overhead components consumes tester capacity. The queueing model incorporates this Overhead Time into the service process by applying an equal amount of Overhead Time to each of the incoming lots. This is not a completely accurate representation of what is occurring, but it is a means of compensating for this non-available time. Therefore, the average service time equals the sum of (1) the average lot processing time and (2) the total Overhead Time divided by the number of arriving lots.

$$\mu_{loh} = \frac{\text{Total Overhead Time}}{\text{Total \# of Lots}}$$

$$\mu_s = \mu_p + \mu_{loh}$$

Now that the Overhead Time is being applied in equal amounts to all of the incoming lots, a variability for the individual lot Overhead Times needs to be determined. As described previously, the typical distribution used to model a completely random process is the exponential process. It has the characteristic of the standard deviation being equal to the mean. So, an assumption is made that the standard deviation of the individual lot Overhead Time equals the mean amount of Overhead Time being applied to the incoming lots.

To summarize, there are two assumptions being made in applying Overhead Time to the service process. The first assumption is that the total amount of Overhead activities can be divided and applied equally to each incoming lot. The second assumption is that these small pieces of Overhead Time being applied to individual lots each represent an exponential process. The validity of these assumptions needs to be considered when evaluating the soundness of the queueing models results.

Finally, the service time variance needs to comprehend the variance in the processing time and the variance of the Overhead Time. Assuming that the lot processing time and individual lot Overhead Time are independent, the service time variance will equal the sum of (1) the lot processing time variance and (2) the square of the average individual lot Overhead Time.

$$\sigma_{loh} = \mu_{loh}$$

$$\sigma_s^2 = \sigma_p^2 + \mu_{loh}^2$$

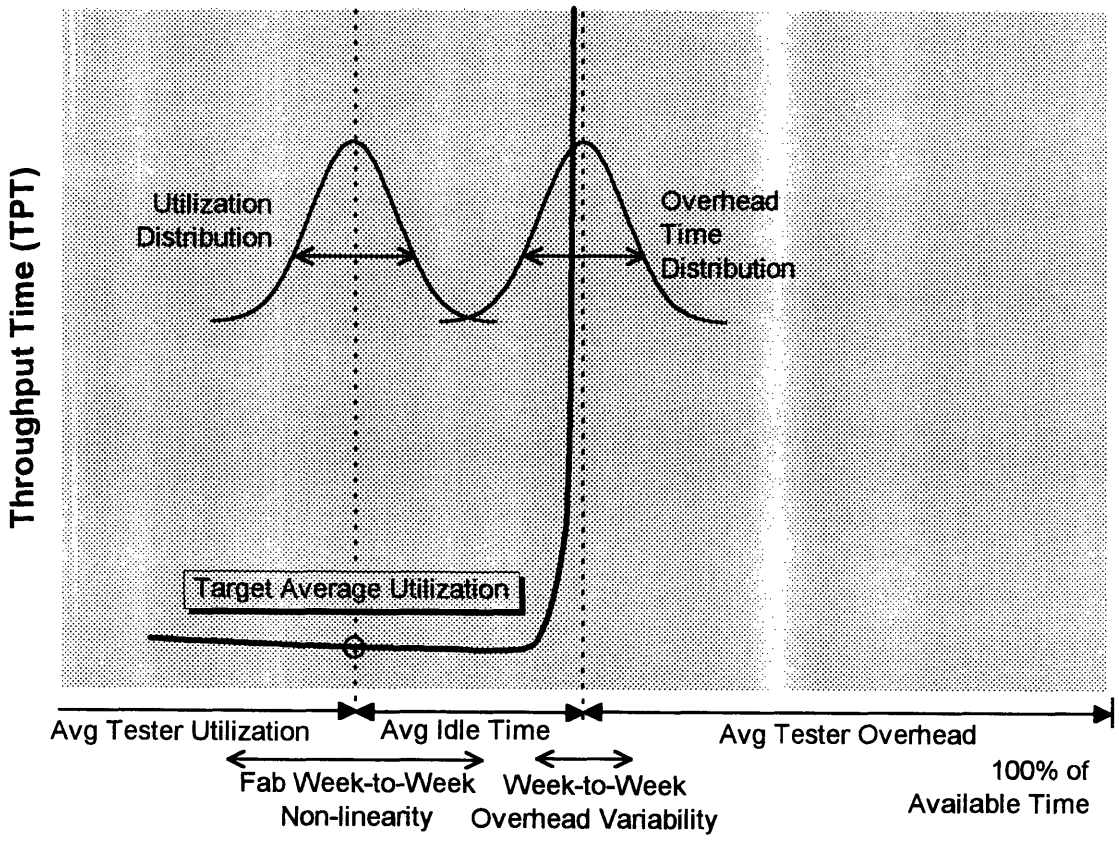
4.2.3 Utilization Framework

Figure 7 provides a graphical framework for applying queueing theory to the Sort tester area. There are a few operational measures which need to be considered when determining the maximum achievable average utilization. The tester utilization and Overhead Time levels will vary from week to week based on the workload entering the area and on Production performance. Variations in Overhead Time change the location of the graph's asymptote. The gap between the average utilization level and the average amount of Overhead Time needs to be large enough to accommodate the variability in those two measures since the total amount of time consumed by Overhead activities acts as a "wall" against high utilization levels. As the utilization level approaches the asymptote, throughput time increases dramatically, while utilization reaches its maximum level. If the floor attempts to move past the asymptote, beyond the maximum utilization level, material will build up faster than it can be worked off. The longer this operating point is sustained, the longer it will take to remove the accumulated WIP. A gap between the average utilization and average Overhead levels must be planned for to minimize the probability of the utilization level moving past the asymptote.

The curve in Figure 7 has a slightly different shape than the curve in Figure 5. At low utilization throughput time is shown as decreasing slightly as utilization increases (up to the "knee" of the curve). This phenomenon can occur when the queueing model contains

an Overhead component which is inversely related to the utilization level. When the utilization level is low, the Overhead component is large, and when the utilization level is high, the Overhead component is very small. If the magnitude of the Overhead component changes faster than the utilization level, throughput time can decrease as the utilization level is increasing.

Figure 7 - Queueing Theory Framework for Sort Testers



4.3 Sensitivity Analysis

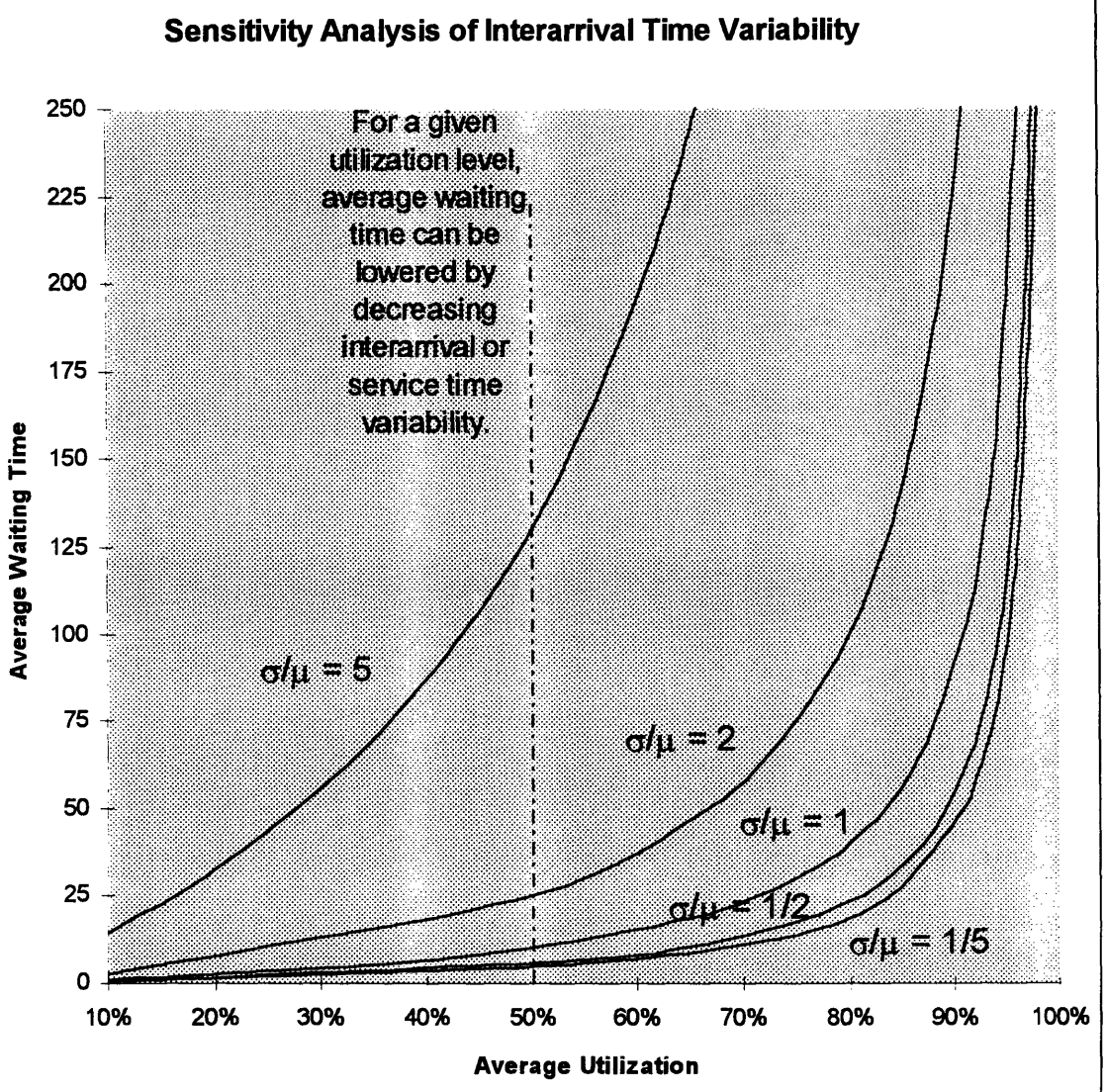
It is important to understand how the other input parameters to Whitt's model affect the queueing system. This section describes the results of performing sensitivity analysis on

the interarrival time, service time, and on the number of servers operating within the system.

4.3.1 Interarrival and Service Time Variance

Increased variability in interarrival or service times at a given utilization level increases the average waiting time of the incoming customers (Figure 8). In the context of the Sort tester area, interarrival variability refers to the "lumpiness" of the material flow into the area, while the service variability relates to the spread of the average lot test times for the different products being processed. Therefore, if the testers are operating at a particular utilization level, waiting time can be reduced either by smoothing the flow of lots into the area or by decreasing the spread of lot test times. Alternatively, these variability reductions can be thought of as harnessing lost tester capacity since higher utilization levels can be attained with the same waiting time.

Figure 8 - Interarrival Variability Sensitivity Analysis

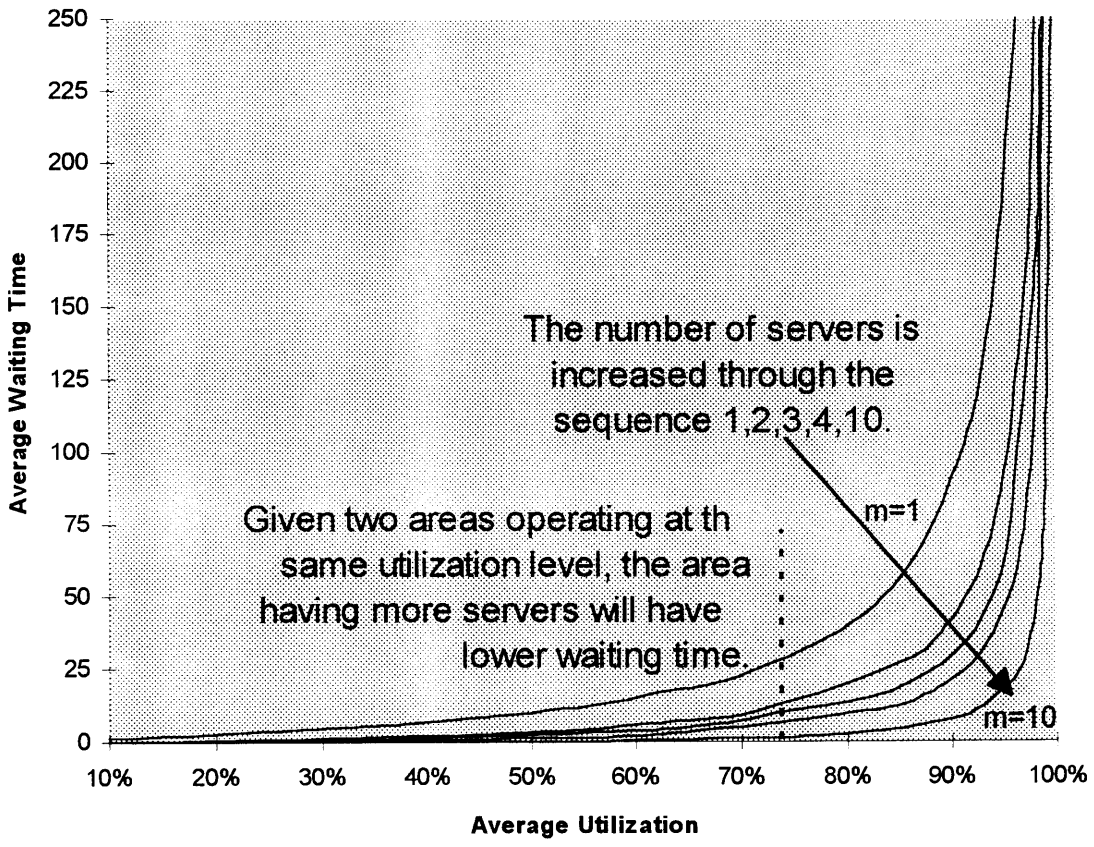


4.3.2 Number of Servers

Having a larger number of servers in a queueing system (for a given utilization level) improves the performance of the system. "The appropriate server utilization typically increases as the number of servers (and the arrival rate) increase" (Whitt, Understanding

708). This analysis assumes the system has the same utilization before and after changing the number of servers. This necessarily implies that the arrival and/or service rate be appropriately adjusted to maintain the same utilization level. The result is that at a given utilization level, the average waiting time will be lower in systems with a larger number of servers (Figure 9). This is very pronounced when increasing above one server, but there are diminishing returns for each incremental server.

Figure 9 - Server Number Sensitivity Analysis



5. Sort 9 Study

The queueing model framework previously presented was applied to Intel's Sort 9 tester area. (Sort 9 refers to a subsection of the total Sort 9/11 floor.) Sort 9 is representative of Intel's logic Sort floors and there is much historical data on which to base a study. Fourteen weeks of data were gathered and input into the queueing model. The model calculated an average lot waiting time which was combined with the average processing time to generate an average throughput time. This chapter first discusses the specific data collection methods used for Sort 9 and then proceeds to review the queueing model results as well as the important insights gained from the study.

5.1 Data Collection

The first step in understanding how to improve tester utilization is to quantify the components of the testers' 168 hours per week. The tester time line can be divided and classified into four mutually exclusive categories:

1. Utilization Time
2. Overhead Time
3. Idle Time
4. Development Time

Utilization Time consists of time the tester is actively processing an incoming, first run lot. This does not include time to re-test lots, assist stalled testers, or any other time that the tester is not progressing towards completing an incoming lot.

Overhead Time is a very large category consisting of any activities or states which take processing time away from a tester. This includes unscheduled downtime, re-tests, PMs, setups, etc., as well as time when a tester is not processing incoming lots because its current setup does not match that of a waiting incoming lot.

Idle Time consists of time when there is no incoming material to process.

Development Time consists of time set aside for development work by Engineering. Sort 9 Engineering is typically allotted a set number of hours per week on one of the production testers. These hours are removed from the time available to process incoming lots and does not play any part in the analysis.

Each of the tester time components needs to be understood as fully as possible. For the Sort 9 tester area, the duration of some activities was estimated due to either lack of duration tracking for the activity, or the available data didn't fit cleanly into one of the above categories. If estimates were required, interview data from Production and Engineering personnel was used to generate typical values.

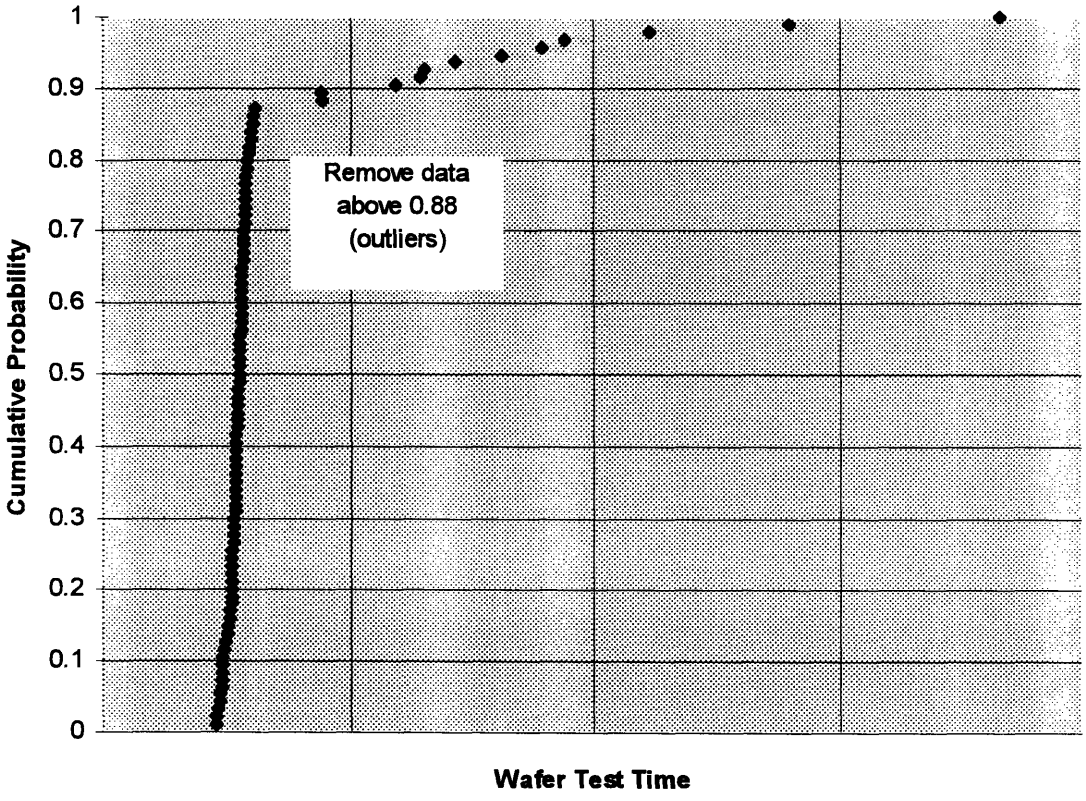
5.1.1 Utilization Time

In calculating total utilization time, the first step is to calculate, on a per product basis, the average time required to test an average lot. This average lot contains the particular product's average number of wafers. With each product's average lot test time and knowing the total number of lots, by product, processed during a week, the total utilization time for the week can be calculated by summing, for each product, the average lot test time by the total number of lots.

Calculating the average lot processing time for a particular product requires "filtering" and then regressing the raw wafer test time data. The factory's database contains time stamp information regarding when a tester began testing a wafer. Subtracting the start time of one wafer from the start time of the next results in the actual time it took to process that particular wafer. This time difference may contain components other than those defined for Utilization Time. Quite often, the time to perform a required operational activity or assist the equipment is captured as well. These other activities would by definition fall into the Overhead Time category since the tester is no longer processing incoming wafers. By collecting many data points (> 200 wafers) for a particular product, a plot of Wafer

Processing Time versus Cumulative Probability can be generated. This was done for all of Sort 9's products processed during the 14 weeks under study. In each case, the top 5-20% of the wafer processing times were much larger (even orders of magnitude larger) than the majority of the data (Figure 10). These high wafer processing times are outliers compared to the normal processing of a lot and contain pieces of Overhead Time (which is being accounted for separately). Therefore, to determine a *true* wafer test time, the outliers are removed, leaving a "filtered" data set.

Figure 10 - Wafer Test Time Cumulative Probability Chart

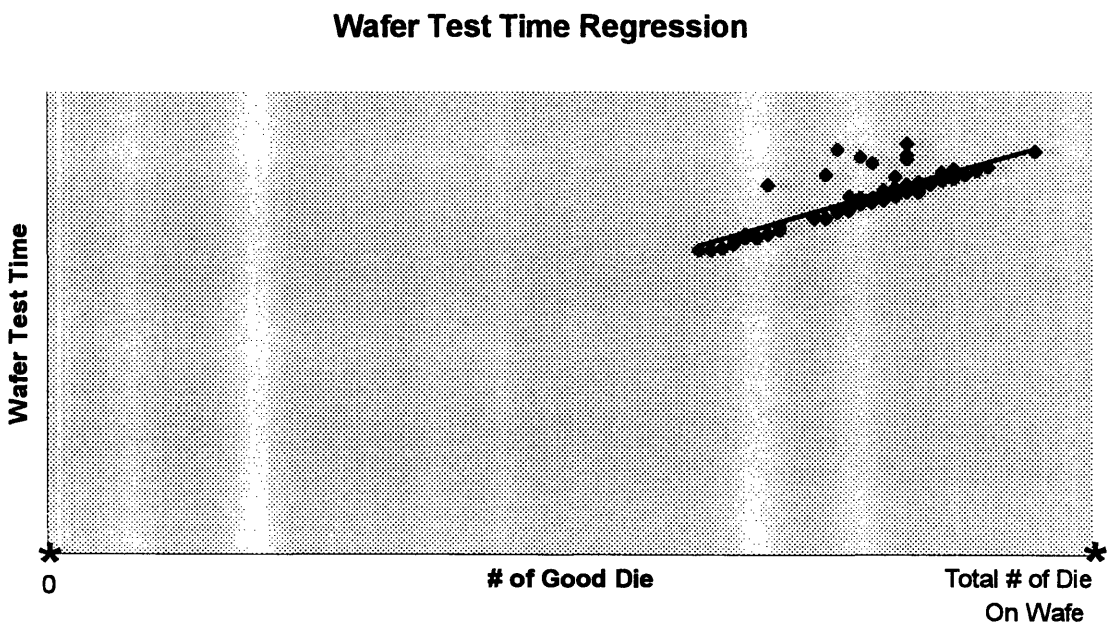


There are two important points regarding the wafer test time calculations which should be noted. Based on knowledge about the test programs in Sort 9, there was a high probability that high wafer test times were the result of other tester activities which the Overhead

Time category captured, thus justifying their removal. If there is reason to believe that large wafer test times are true processing time data points, the filtering process should not remove them from the data set. This can be the case when test programs are written to collect much more information when a failure occurs or when they are written to periodically collect detailed engineering data. Secondly, it was stated that calculating wafer test time information needs to occur for every product. This is not precisely correct since individual products can be tested using different test programs, significantly changing the test time distribution for that product. As a product matures, the test program, or *test tape*, is improved and a new revision is added to the library of available test tapes. It is the product/test tape pair that is the lowest common denominator on which the wafer test time analysis should be performed.

The test programs at Sort 9 have a linear relationship with the number of good die on the wafer. Since a bad die is defined as one the fails a test program and a good die is one that passes all of the tests, it takes longer to test a good die as compared to a bad die. This also means that as the number of good die on the wafer increases, the wafer test time will increase. Using the filtered data, the number of good die can be regressed against the wafer test time to give a die yield dependent wafer test time (Figure 11). This analysis results in a core database of die yield dependent wafer test time equations for each product/test tape pair.

Figure 11 - Estimating Die Yield Dependent Wafer Test Times



T = Total Wafer Test Time

ε = iid Random Variable

N_{GD} = # of Good Die on Wafer

N_{BD} = # of Bad Die on Wafer

t_{GD} = Average Good Die Test Time

t_{BD} = Average Bad Die Test Time

N_{TD} = Number of Total Die on Wafer

DY = Die Yield

$$N_{TD} = N_{GD} + N_{BD}$$

$$DY = \frac{N_{GD}}{N_{TD}}$$

Derivation of Regression Equation:

$$T = N_{GD}\, t_{GD} + N_{BD}\, t_{BD} + \varepsilon$$

$$T = N_{GD}\, t_{GD} + \left(N_{TD} - N_{GD}\right) t_{BD} + \varepsilon$$

$$T = \left(t_{GD} - t_{BD}\right) N_{GD} + N_{TD}\, t_{BD} + \varepsilon$$

$$\hat{y} = \quad m \quad x \quad + \quad b$$

$$b = N_{TD}\, t_{BD} \qquad m = t_{GD} - t_{BD}$$

$$t_{BD} = \frac{b}{N_{TD}} \qquad t_{GD} = m + \frac{b}{N_{TD}}$$

Relating the Regression to DY: $T = \left(t_{GD} - t_{BD}\right) DY\, N_{TD} + N_{TD}\, t_{BD} + \varepsilon$

In determining Utilization Time from historical data, the die yield of a product is known. When forecasting, a die yield can be predicted from historical data and/or based on planned improvements. In either case, using the die yield, the quantity of wafers per product, and the die yield dependent wafer test time equations, a total *expected processing time* (EPT) can be calculated. Total EPT gives an approximation to the total amount of time that will be needed to process the given amount of product at those die yields. EPT only quantifies the uninterrupted amount of processing time, thereby removing the time for any operational activities which are occurring. EPT is therefore equivalent to utilization time.

5.1.2 Overhead Time

The category of Overhead Time contains the duration of all tester activities (except running incoming lots) that prevent another incoming lot from being loaded onto the tester. Overhead Time is a difficult category to estimate since much of the information is not currently tracked. Therefore, this category was quantified in a bottoms-up fashion by first defining each of the constituent parts of Overhead Time and then quantifying them based on available data or on interviews with the Production or Engineering personnel. The sub-categories of Overhead Time need to be mutually exclusive and sum to the total Overhead Time. To determine total time for an activity requires knowing the average frequency of the activity and the average duration of the activity. Of the ten Overhead components defined for the Sort 9 testers, most had actual data for either the frequency or the duration element, reducing the overall estimation error.

In an environment such as the Sort tester area, it is difficult to model the true capacity of the equipment since it is highly dependent on the efficiency of the operators. When a workforce is highly motivated, the capacity of a work area can be significantly greater as compared to periods when the workforce does not feel motivated to operate efficiently. Fine and Graves applied a Tactical Planning Model developed by Graves (1986) which captures this idea by assuming "a control rule in which the production output from the sector is proportional to the WIP level at the sector" (Fine and Graves 21). Data from

Sort 9 (both numerical and anecdotal) suggested a similar relationship. To capture this decrease in efficiency during low workload periods, the queueing model contained a workload dependent component as part of the Overhead Time category. It is a very large part of Overhead Time at low utilization levels and drops to zero at moderate utilization levels. It had been noted by a Sort 9 manager that, to his surprise, he found during some periods throughput time actually decreased as tester utilization increased. This type of phenomenon can be demonstrated by a queueing model that comprehends workload dependent capacity.

5.1.3 Idle Time

Idle Time occurs when there is no incoming WIP to run. At Sort 9, idle time (as defined) does not occur very often - there always seems to be something waiting to be processed. Idle time can be calculated by subtracting the other three categories from the total tester time. In a dynamic environment, where lots arrive and are processed in a truly random and independent fashion, a gap of idle time is necessary. Without this gap, the arrival and service variability will cause throughput time to continually increase.

5.1.4 Development Time

Sort 9 Engineering requires tester time to prepare new products for high volume manufacturing and to improve the processes on the floor. They are allotted a set number of hours per week on one of the production testers for engineering development work. In the queueing model, this time is subtracted from the total weekly tester time to give a true total production availability time.

5.2 Queueing Model Results

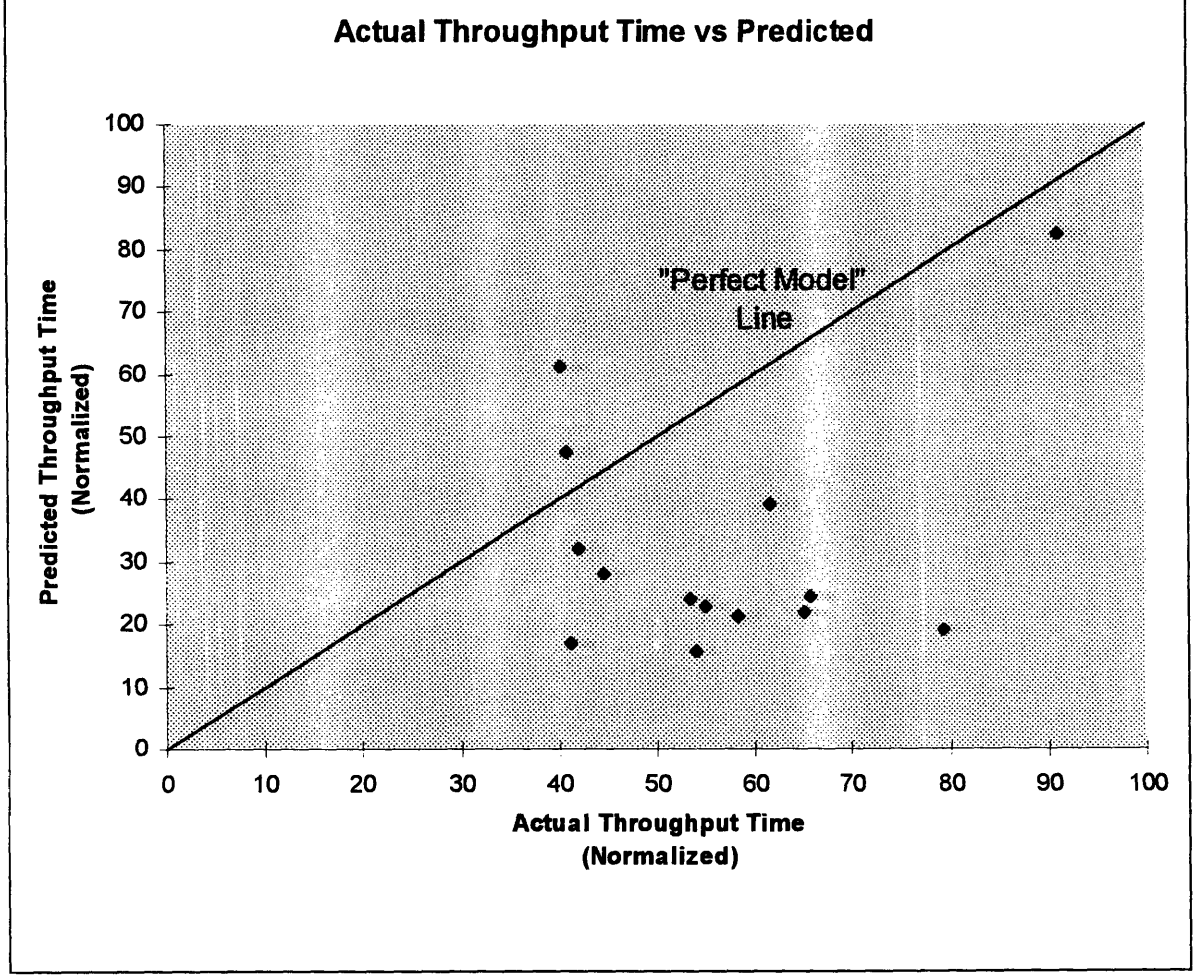
This section will briefly cover the model results followed by a discussion of the insights gained from the collected data.

5.2.1 Queueing Model Performance

A plot of expected versus predicted throughput time for the fourteen weeks under study is shown in Figure 12. The model predicted throughput times which were on average lower

than the true throughput times. Given that many of the input variables were estimated from anecdotal or incomplete numerical data and that some of the basic queueing model assumptions are in slight violation, it would not be expected that the model generate exact results. The accumulated data for each of the fourteen weeks revealed that the testers were busy with wafer testing or Overhead activities for a large percentage of their available time. This corresponds to operating near the "knee" of the curve in Figure 7; a very sensitive area of the model. Small errors in estimating total processing time and/or Overhead Time result in large errors in the resulting throughput time calculated by the queueing model. The inaccuracies in the input data is too large when performing analysis in this region of the curve. Increasing the accuracy of the input data would help the resulting output, but as mentioned previously, the purpose of the model was to provide a framework for analyzing the operational aspects of the Sort tester area and to improve the understanding of the utilization metric. Although more accurate model output was desirable, it's inaccuracies do not invalidate the purposes of the model.

Figure 12 - Queueing Model Results



5.2.2 Limitations to Increased Utilization

The goals of analyzing Sort 9 using queueing theory were to provide insight into the utilization metric and to understand the limitations to increased utilization of the Sort testers. Creating the queueing analysis framework provided a fresh and improved understanding of the utilization metric. Pulling together the necessary data for the queueing model provided the insights to uncovering the two fundamental limiters to increased tester utilization: expected processing time (EPT) variability and Overhead Time. Figure 13 graphically depicts the EPT for the wafers processed in Sort 9 during the 14 weeks of analysis. The week-to-week EPT *variability* limits the achievable average

utilization level. To understand this, a shared basic assumption held by the Sort organization needs to be discerned (Schein 21-22). The organization primarily operates under the belief that Sort cannot limit Fab output. Therefore Sort must be able to process the planned Fab output each week. The problem is that Fab loadings are primarily based on wafer quantity and not dependent on the product mix. So the Fab treats wafer run times for different products equal, while this is far from the truth for the Sort area. The EPT metric is a much more appropriate indicator of Sort loading since it compensates for product mix variations (Figure 14). In statistical terms, stating that Sort is supposed to handle almost anything Fab delivers is equivalent to a $+3\sigma$ Fab output, where σ can be calculated from the 14 data points. Sort needs to have enough capacity available to process this $+3\sigma$ EPT level, but given the high variability of the Fab output, this extra equipment will not be needed during most weeks. Being required to run whatever comes out of the Fab, given its variability, requires that testers will quite often not be used. This effectively limits the average utilization level that can be maintained.

Figure 13 - Expected Processing Time Distribution

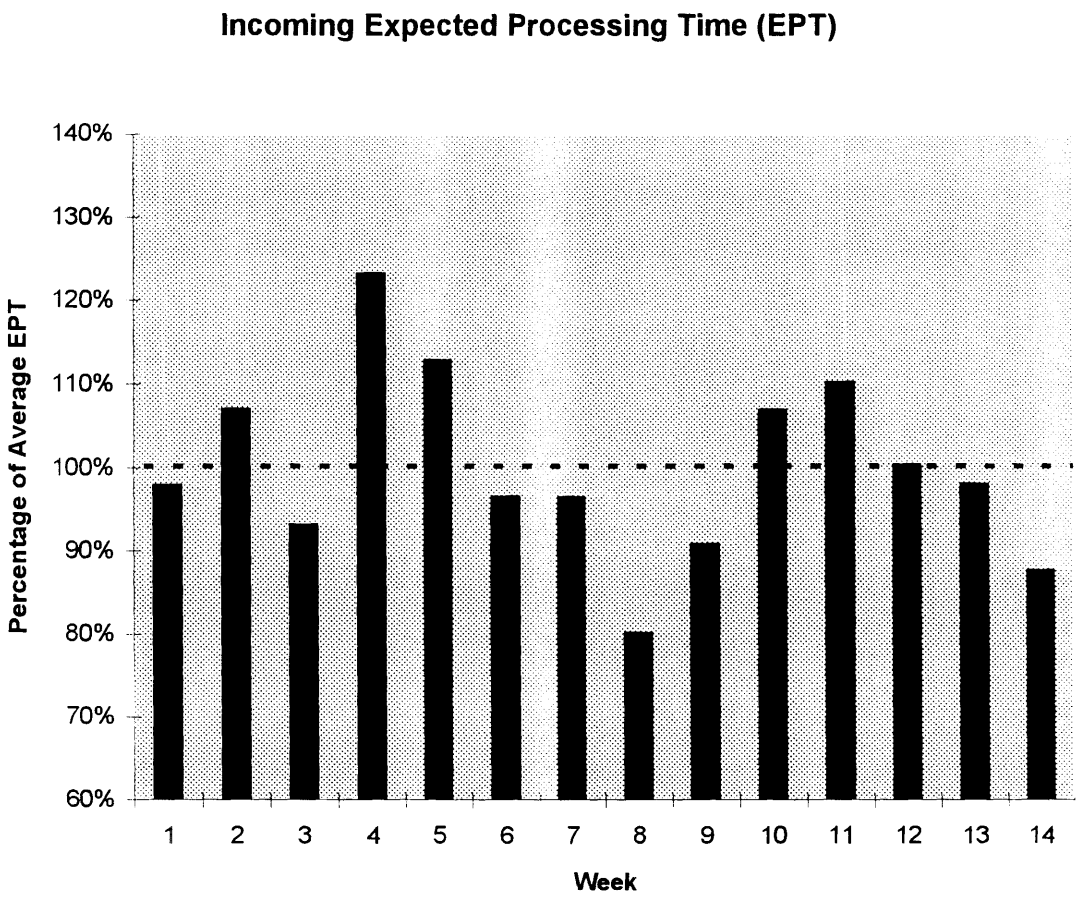
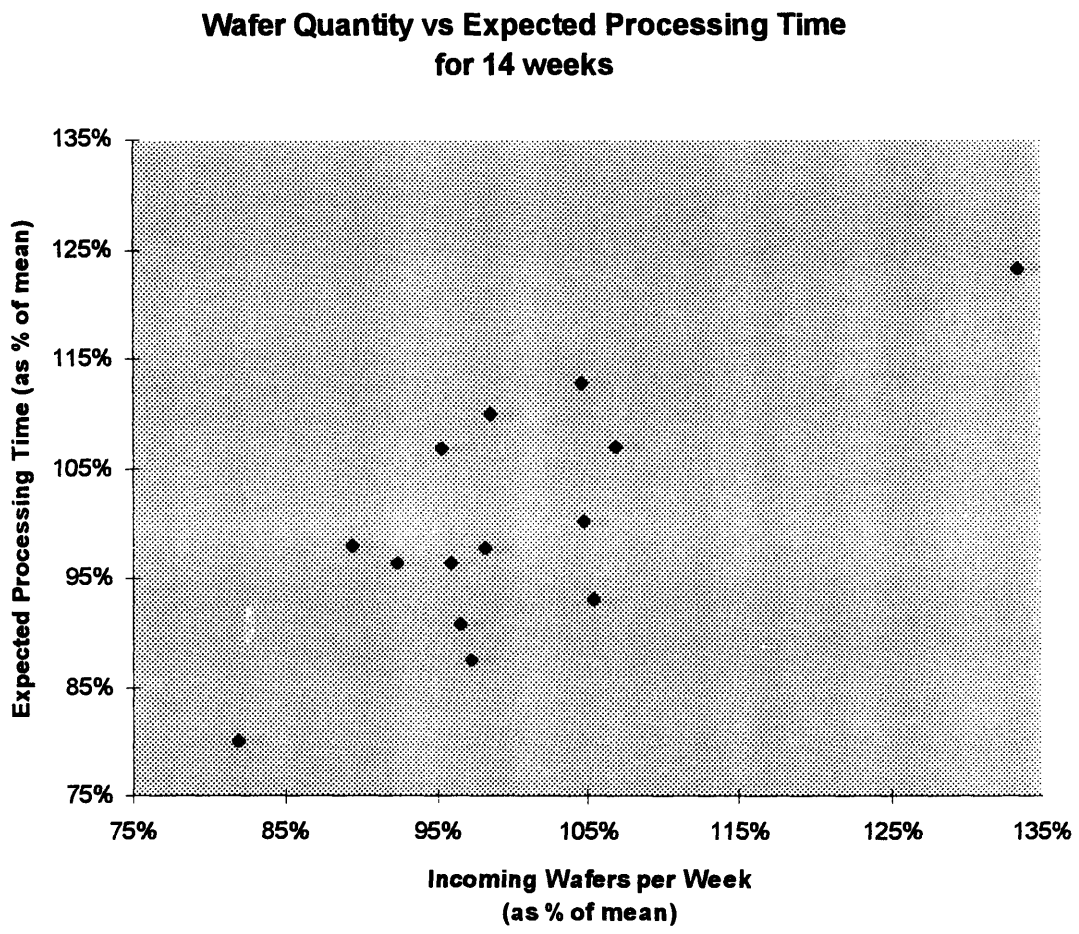
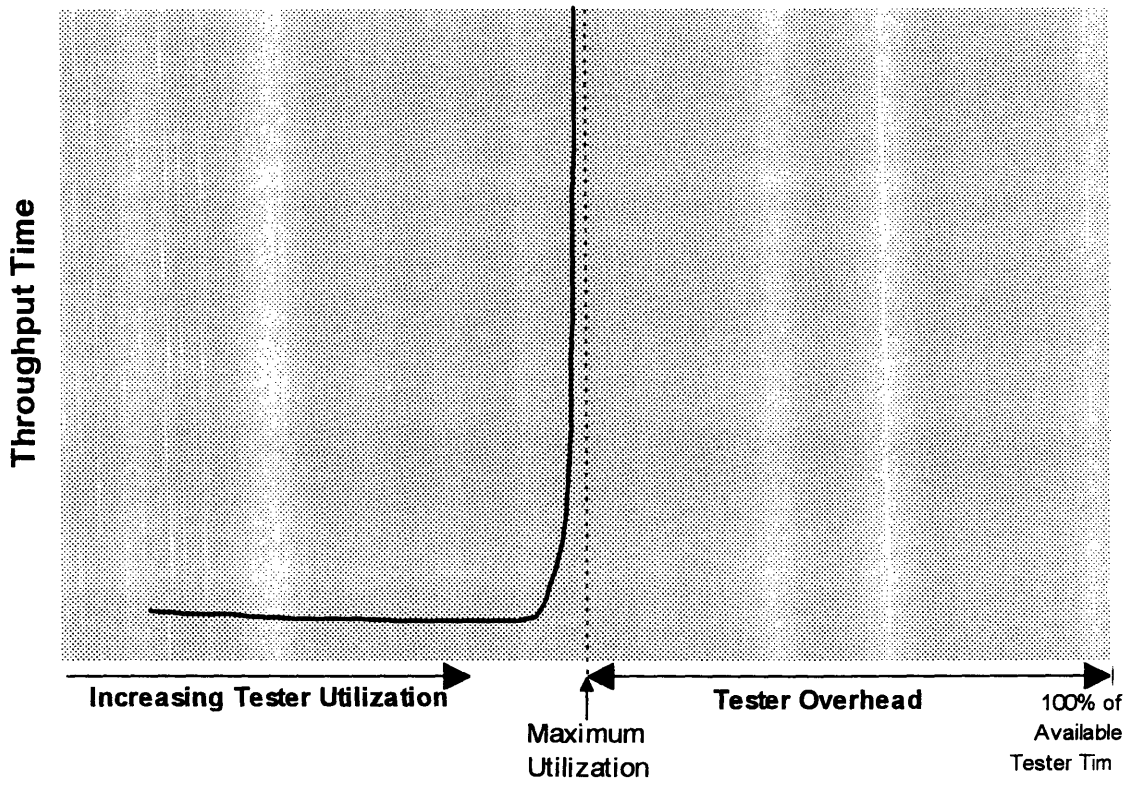


Figure 14 - Wafer Quantity Poor Predictor of Total Processing Time



The second limiter to increased utilization is the amount of weekly Overhead Time. The bottoms-up approach used in calculating weekly Overhead Time for this study highlighted the magnitude of these capacity consuming activities. Most of the Overhead Time components were not being tracked so the analysis was instrumental in determining the major consumers of tester capacity. Given highly variable Fab output, during a heavy workload week the amount of time available to process the incoming lots needs to be at its maximum. Overhead Time is what limits that maximum amount (Figure 15). By continually tracking and reducing the Overhead components, the maximum attainable utilization can be increased. This also increases the average utilization level that can be maintained.

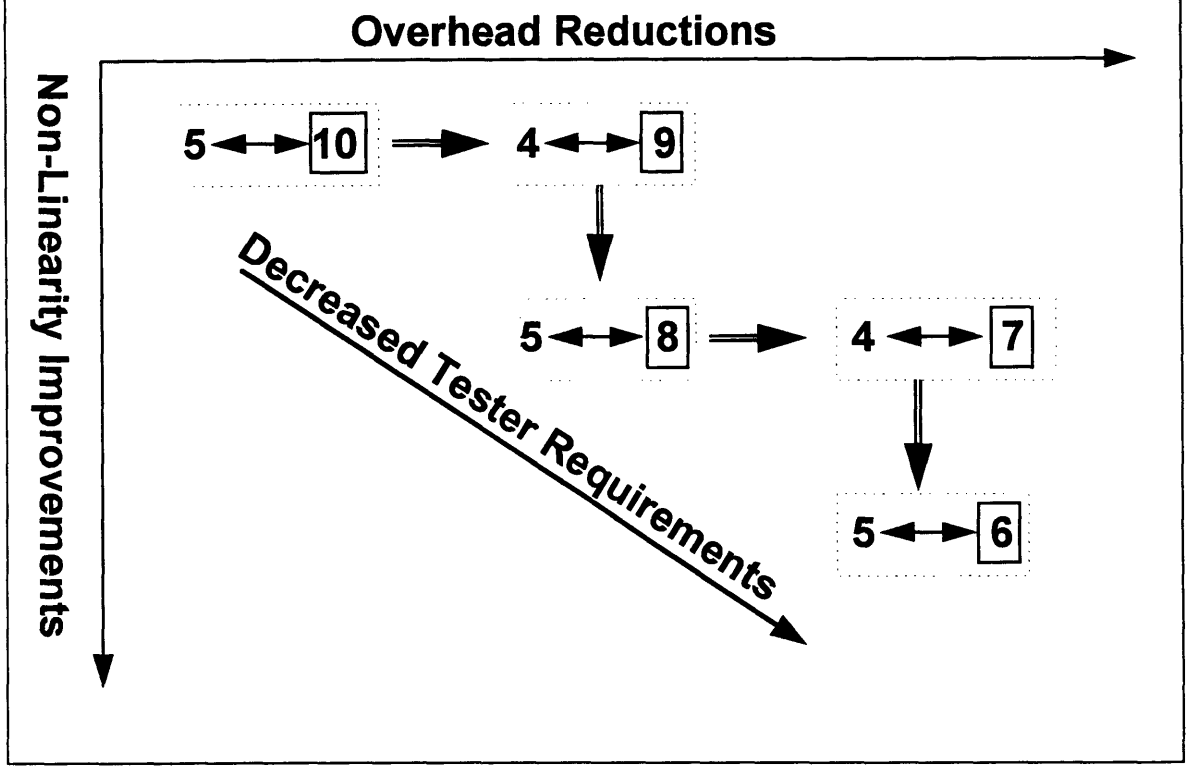
Figure 15 - Overhead Time Limits Maximum Utilization



6. Addressing Limitations to Increased Utilization

Recommendations on how Sort 9/11 can reduce the limiters to increased utilization (EPT variability and Overhead Time) are presented in this chapter. Improvements in either one of these areas effectively creates tester capacity by increasing the maximum attainable average utilization. Figure 16 graphically displays an example utilization improvement road map. Beginning in the upper left hand corner, the number of required testers swings from 10 during heavy workload weeks to 5 during light workload weeks due to weekly Fab output variability. Therefore, the average number of required testers is 7.5 (= (5+10)/2). Since Sort needs to be able to handle the heavy workload weeks, 10 testers are required on the floor. Reductions in Overhead Time would reduce the average number of testers required to 6.5, but the week to week variability would remain the same, causing a shift from 4 to 9 testers depending on the weekly workload. The Overhead Time reduction has reduced the number of testers needed on the floor from 10 to 9. Improvements in EPT variability would not affect the average number of testers required on the floor, but instead would decrease the maximum number of testers needed during the heavy workload weeks. In the Figure 16 example, the number of testers needed on the floor has been reduced to 8 by the EPT variability reduction. The graphical representation of the improvement road map depicts the improvements as discrete events, yet it is more accurate to think of improvements as continuous events that would follow the arrow from the upper left to the lower right hand corner of the graph.

Figure 16 - Utilization Improvement Model



6.1 Reducing EPT Variability

The primary concern of Fab 9 Production Planning is to fill Fab 9 with wafers. They tend to make the assumption that if the Fab is able to process a weekly wafer amount, Sort should be able to process it as well. And if Sort has difficulty processing the wafer quantity, it is a Sort production problem. Although this may be the cause, it is not the only possibility. As discussed earlier, wafer quantity is not a good predictor of the workload imposed on Sort (Figure 14). Product mix has a very large impact on Sort's weekly loading. Calculating weekly EPT levels in the planning process would expose the true workload level imposed on Sort, since the planned weekly wafer starts become the weekly line items for Sort (after being processed through the Fab). The Production Planning time horizon is several times larger than the factory throughput time. If weekly EPT were calculated across the planning horizon, EPT variability could be tracked. Sort should set EPT limits (both high and low) so it is clear when a trouble week is approaching. At this

early point in the planning cycle Production Planning has some leeway to make adjustments to the planned product mix. Long test time products scheduled during a high EPT week can be traded for short test time products scheduled during a low EPT week. This approach need not reduce Fab loadings since the quantity of wafers will not change substantially, only the product mix will change. Reducing the EPT variability at the input of the Fab is a major step towards reducing one of the main causes of low tester utilization: EPT variability.

6.2 Reducing Overhead

This project's analysis has determined two causes of low tester utilization: EPT variability and tester Overhead Time. These two are not completely independent of one another. The EPT variability has played a part in making Overhead Time a large problem. The variability combined with Sort's interest in not limiting Fab output has resulted in excess tester capacity being placed in the area. The full capacity available is actually only needed during high EPT periods, yet the testers are physically available for production every minute of the week. This creates numerous periods during which the production organization is not under pressure to perform efficiently. Having additional capacity available allows inefficiencies to develop in the operation of the equipment. The excess capacity will act to hide those inefficiencies since the required production items are processed and the schedule is met. Occasionally, there are high EPT periods and the floor has trouble meeting schedule so the problems are investigated, but the high EPT period quickly fades away and the problems are no longer important. This cycle happens occasionally, but infrequently enough to lose continuity between the improvement efforts. During the period of this project at Sort 9, one of these high EPT weeks occurred, and WIP levels in front of the testers grew dramatically. During the following weeks, record utilization levels were achieved on the testers, but not nearly as high as was expected. Both Production and Engineering felt a great amount of pressure to improve the operation of the test equipment. The Overhead Time was limiting the testers' maximum utilization level, but the employees were unable to quantify where all of the tester time was going. They soon realized that they don't have adequate systems in place to quantify the

Overhead Time components and, more importantly, that *you can't improve something unless you measure it.* The people working on the problem began making progress in understanding the Overhead Time breakdown, but another solution to the problem quickly surfaced - send the material to another site for processing. With that solution in hand, the pressure to understand the root cause of the problem was removed. Within a few weeks the Sort area was running as it had before the incident.

The pressure placed on the organization during the above incident created a desirable motivation for improvement. Ideally, any system put in place to address Overhead Time should attempt to foster a continuous improvement environment. The Just-in-Time (JIT) manufacturing philosophy gives some insight into achieving a continuous improvement mentality within the workforce. A fundamental operational aspect of the JIT philosophy is to continuously reduce the inventory level throughout the manufacturing process in order to expose problems in the production system. The typical analogy is water running in a river. The water is the inventory, and as it is reduced, the rocks in the river (problems in the process) are exposed. After the exposed problems are addressed, the inventory level is once again reduced. Always stressing the system in this manner develops a competent and efficient workforce which is continuously improving and increasing their understanding of the production operations.

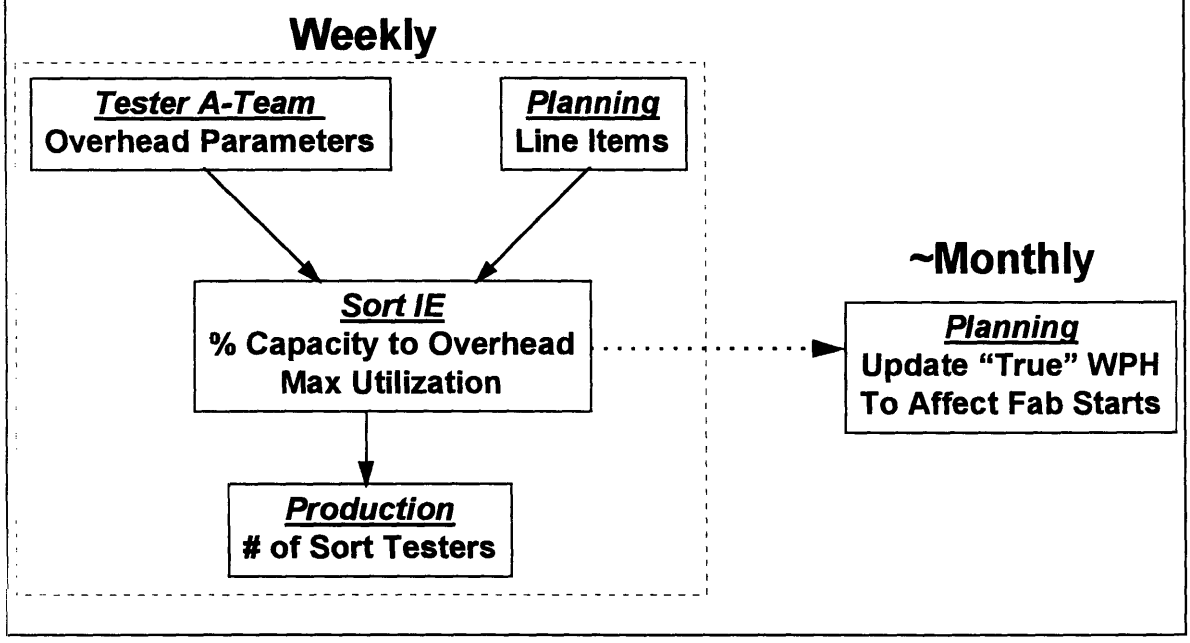
Sort 9 needs to foster the continuous improvement mentality by continuously stressing the system as JIT promotes, but instead of decreasing inventory to stress the system, decrease the available capacity. "Flex" the tester capacity to match what should be needed so the floor always feels they are the bottleneck to increased floor performance. Limiting capacity on low EPT weeks keeps the pressure applied so the floor feels a sense of urgency to rectifying problems. By forecasting the EPT of the weekly line items and the Overhead Time required, the number of needed testers can be calculated. Through this process, the number of production tester hours available will match the EPT of the weekly line items, and the testers which are "active" for production will be running at their maximum utilization - only limited by the amount of Overhead Time. It is an efficient use

of Intel's resources since the testers not actually needed by the Production floor can be used for other purposes. They can always be used for development work by Engineering since new products and processes are continually introduced. Also, other Sort floors needing capacity for the week can off-load work to the "non-active" Sort 9 testers. *Any* use of the excess capacity is better than having inefficiencies consume the valuable tester resources.

6.3 Developing an Operational System

For Sort 9, addressing the fundamental limitations to increased utilization requires support from Sort Industrial Engineering (IE), the Tester Improvement Team, Fab 9 Production Planning and Sort Manufacturing. The proposed process is depicted in Figure 17.

Figure 17 - Operational Activities Supporting Improvements



6.3.1 Sort Industrial Engineering

The coordinating point for the process is Sort IE. Each week Sort IE receives next week's forecast of the Overhead parameters and the EPT of the line items. The Overhead

parameters (used to derive the predicted Overhead Time amount) are delivered by the Tester Improvement Team, while the EPT figure is reported by Planning. Based on these two quantities, the Sort IE can calculate the estimated number of testers required and the maximum attainable utilization (as dictated by the total Overhead Time). The individual Overhead parameters, tester capacity devoted to Overhead activities, weekly EPT, and estimated number of testers are tracked for trending improvement progress. The Sort IE is the owner of the die yield dependent wafer processing time equations used by Planning to calculate weekly EPT. Sort IE is responsible for updating this information on a regular basis (approximately monthly) to insure Planning's forecast EPT's are as accurate as possible. As previously discussed, the die yield dependent wafer processing time equations are used to calculate EPT's, so the equations need to be derived from *true* wafer test times. Any Overhead Time information being captured in the wafer test time data needs to be filtered (Figure 10).

6.3.2 Tester Improvement Team

The cross-functional Tester Improvement Team is composed of the appropriate membership to understand and improve total Overhead Time. Ideally, the team would quantify all of the Overhead Time components so a pareto displaying percentage of capacity lost to each component could be created. Some of these components will be difficult to quantify, but it is important to remember that *any* estimate is better than *no* estimate. Sub-teams could proceed with a root cause analysis of the dominant loss components. Proposed solutions would then be discussed and an improvement plan implemented.

The magnitude of the Overhead Time components are typically dependent on wafer quantity, lot quantity, or the number of physical testers running production material. Therefore, Overhead Time components can be quantified and tracked in terms of Overhead Time parameters which are independent of Sort loading. The Sort IE needs to receive the parameter values from the Tester Improvement Team so the Sort IE can calculate total Overhead Time using the weekly variables of wafer, lot, and tester quantity.

6.3.3 Production Planning

Production Planning has a large influence over one of the fundamental limiters to higher tester utilization - EPT variability. They need to be concerned with the variability of the workload being placed on the Sort tester area. This requires that the group include weekly EPT calculations over their planning time horizon. Planning will then be exposed to the impact of product mix on the Sort tester area. Sort IE will provide upper and lower weekly EPT bounds which Planning cannot exceed. The upper bound represents the maximum amount of Sort tester time available to run incoming production material. The lower bound is not a hard boundary but rather encourages Planning to keep the Sort floor loaded just as it tries to keep the Fab loaded. But instead of using wafer quantity to load the Sort floor, Planning can use product mix adjustments. Sort IE is also responsible for supplying and periodically updating the die yield dependent wafer test time equations necessary to calculate EPT. Since the planning process is already heavily developed around spreadsheet programs, implementing the calculation of an additional weekly EPT value should be a relatively minor enhancement.

6.3.4 Sort Manufacturing

The support of the production organization is needed in two particular areas: (1) using only the number of testers dictated by Sort IE, and (2) supporting and following the Overhead Time improvement programs generated by the Tester Improvement Team. Assuming that the Sort IE calculation of the required number of testers is correct, Production's disregard of the process will provide excess capacity on the floor which will have the negative result of removing the pressure to be efficient. The excess capacity will act as a buffer to perpetuate the inefficiencies. Also, as a side effect, the utilization level of the "active" testers will be lower than desired. Production needs to be actively involved in the Tester Improvement Team. Their personal experiences with the equipment will direct operational improvements that reduce Overhead Time. Once improvements in processes (or otherwise) are implemented, Production has the responsibility to insure that the improvements are followed and maintained. Without Production's follow-through, the improvements are useless.

6.4 Fundamental Enablers

The success of the operational system presented in the last section hinges on two fundamental enablers that have already been touched upon. The first concerns the organization's ability to accurately predict weekly EPT and Overhead Time. Given inaccurate predictions, Sort IE will incorrectly estimate the number of testers for production to use. This will make it nearly impossible for the floor to process the week's line items. This will not create the continuous improvement environment that flexing tester capacity was intended to promote. Accurate forecasts are generated by tracking past data, but the current systems in Sort 9 do not track the needed information. To enable the operations system proposed, the Sort operations need to become predictable, which requires data systems that accurately track the true wafer test times and Overhead Time components. Once these systems are in place, accurate data can be used by Planning and the Tester Improvement Team, and fed to Sort IE for accurate tester capacity forecasts.

The second fundamental enabler concerns escalating the importance of increased tester utilization throughout the Sort organization. The proposed operational system relies on the premise that the people involved are motivated to make the system work. This is true for any new process to be successful. Incentives need to be in place to gain the desired actions. A utilization goal for the Sort testers needs to be committed to up front and the employees reviewed on their progress toward that goal. At a more basic level, understanding of the utilization metric itself needs to be promoted. The employees need to comprehend the metric before they can be expected to purposely influence it.

7. Supplementary Recommendations

The limiters to increased utilization have been presented along with recommendations to address those limiters. This final chapter discusses other recommendations regarding Sort.

7.1 Tracking Tester Time

The "fundamental enablers" discussed in the previous chapter touched upon an important problem with the current state of data collection from the Sort testers. There are many tester activities which are difficult if not impossible to quantify. Only by quantifying the tester activities can improvement priorities and goals be set. Given the advanced processing and controlling equipment, it should be possible to categorize every moment of time on the testers. The systems being designed for the Intel's newest Sort floors should have this ideal as a goal. For older generations of equipment, or at least to gain exposure to activity time in the short term, simple manual systems should be implemented. This would require extra effort from the equipment operators. They would need to understand the importance of tracking tester activities. In fact, since the operators are the most knowledgeable concerning operation of the equipment, they should be heavily involved in determining how the manual systems will work. As long as they understand what is trying to be tracked, and the importance of tracking it, they should be able to come up with the most optimal manual systems which negligibly impact their duties. Although completely automatic tracking will not occur overnight, at some point the manual systems should give way to automatic equipment tracking with improved data accuracy and resolution.

7.2 Consistent Metrics

This project focused on utilization, only one of many metrics used in the production environment. Within Intel, information is commonly shared between sites to develop improved operational methods, but the metrics being used are often calculated in different ways or based on different assumptions. One of the most frustrating activities when trying to communicate metrics across sites is attempting to normalize the metrics so a comparison can be made. There is a need for one set of agreed upon metric equations to facilitate communication and accurate comparison.

7.3 Within Week Scheduling

Sort's production schedule is generated on a weekly basis. This provided the unit of measure for the analysis. Weekly workload variability was determined to be one of the main causes of low tester utilization. Within the week, lots arrive in a highly variable manner as well, limiting average tester utilization. Within week variability needs to be controlled, just as weekly variability does, to improve tester utilization. Within week production scheduling can also benefit the setup strategies for the Sort testers. Setup strategies dictate which testers are going to be setup for which products and when the setups will be changed. There are three levels of sophistication for setup strategies: highly controlled, heuristics, and simple rules.

- The highly controlled level tracks each lot, knowing when it will arrive into the area, and determines which tester it will go to and when it will approximately be complete. This level of sophistication needs to be tightly integrated into the information systems on the floor and may use a linear program.
- At the heuristics level, a set of rules is developed based on past experiences or simulation models. The rules dictate how to react to different weekly product mix situations.
- The "simple rules" level can be compared to a very simple heuristic.

Even though Intel would be capable of developing the highest level of sophistication, it is probably over-kill for the problem. Developing one of the lower two levels would generate the most efficient system based on effort spent developing it.

An interviewee related his knowledge of another company's tester setup strategy. The policy stated that when a tester completed processing of a certain product type, the testers' setups would almost immediately be changed to the waiting product type. At first glance this may seem to be a very inefficient approach, but once it was implemented, it developed into a very efficient system because of a couple of side effects created by the policy. First, product setup's became much more efficient because the operators were required to do them much more often. Operators had always disliked performing a tester setup and therefore had not practiced or improved the setup process. The new policy

forced them to improve. The second side effect is also related to the operators aversion to performing setups. Once the new setup policy was in place, the operators began communicating with upstream operations on a much more continuous basis to modify the product flow so the number of required setups would be reduced as much as possible. The optimized flow of product types greatly reduced the amount of wasted capacity due to unnecessary setups.

7.4 Sort Tester Buffer

Maintaining a buffer in front of the Sort tester area would decrease Overhead Time by reducing the number of unnecessary setups which occur during the week and also protect against "starvation" due to upstream variability. The trade-off to creating a buffer is increased throughput time through Sort. The intention of a buffer is to improve the efficiency of the "bottleneck" operation. Previous recommendations involved "flexing" the number of Sort testers used for production to match what is needed for each week's line items. This has effectively makes the testers a bottleneck operation. The buffer improves the efficiency of the testers allowing the equipment to run at a higher maximum utilization. Specifically, the buffer reduces the amount of time wasted from unnecessary setups and mismatched setup time. These two inefficiencies occur on the floor when a tester no longer has material of a certain product available to run, yet not all of the weekly line-items for that product have been met. The operator then needs to decide whether (1) to wait for more of that particular product to arrive (mismatched setup time) or (2) to change the setup to a product which is available (unnecessary setup). The ideal situation is to have neither occur. The number of weekly setups should equal the number of products listed in the weekly line items, and the products should already be available to run, eliminating the mismatched setup waiting time. A buffer is a simple approach to achieving this ideal versus the tracking/optimizing approach, which would involve matching the expected arrival time of the incoming lots to the tester's run schedule. This is not an easily realizable approach due to the variability of upstream operations and the limited response time for the Sort tester operation. The most successful manufacturing philosophies seem to be the

most simple manufacturing philosophies. Strategically placing buffers at bottleneck operations comes from a very simple manufacturing philosophy: Constraint Management.

The size of the buffer in front of the testers should be large enough to ensure efficient bottleneck operation while not significantly degrading the time for "information-turns" between Sort test results and Fab yield engineering. This information delay is a major concern since the more time that passes between finding and solving a problem, the more scrap product that is being produced. One possible solution is to sample the product entering the area to uncover any problems as early as possible. The tradeoff for delayed information is improved tester setup efficiency. To recover *all* of the setup inefficiencies the buffer size would need to equal the line item time frame (currently 7 days). This would guarantee that the floor had every product that they needed, when they needed it, regardless of the setup strategy. Obviously, smaller buffer sizes would have less of an impact, but it would still be fairly substantial. An analysis of when lots arrived versus when they were needed would provide insight into the benefits of different buffer sizes.

As Constraint Management promotes, all of the upstream operations need to be focused on delivering what the bottleneck (the Sort testers) needs. Sort, Grind, Gold, and E-Test need to know the appropriate product composition for the buffer. They then work to maintain that buffer so the testers do not starve. Sort would need to work more closely with the Fab back-end to assure that they know what Sort desires.

One of the fundamental assumptions typically made in queueing theory is that the arrival and service processes are independent. If the buffer replenishment rate is tied to the product departures from the buffer, then the independence is lost, and applying a queueing model is not useful. With a properly sized buffer and upstream operations working to maintain the buffer, the issues surrounding the arrival rate variability disappear. Then the only limiter to increased utilization becomes the total amount of Overhead Time. Sort resources can then become focused on reducing the amount of capacity lost to the Overhead activities.

7.5 Sort Capacity Optimization Across Intel Sites

An option for reducing the upper EPT limit is to plan weeks of high EPT, over the full Sort capacity limit, and arrange to have another Intel site test the wafers. Statistically, it would be expected that one floor's high EPT week would coincide with another floor's low EPT week. Combining each of the individual Fab's output results in an overall lower variability than each of the Sort floor experiences. If all Intel Sort floors had accurate forecasts of future weeks' capacity needs, Intel's Sort floors could minimize the total number of testers necessary within the corporation. The added transportation cost would be negligible compared to the high capital savings from fewer testers.

7.6 Organizational Structure

Chapter 2 presented Sort 9/11's organizational structure in relation to Fab 9 and Fab 11. At a typical fabrication site, although each Fab has its own Sort floor, the two areas are still organizationally separated. Since the Sort floor does not require the same level of cleanliness as the Fab floor, Sort is also physically separated from the Fab. This organizational and physical separation creates a wall between the two organizations which inhibits information flow. The site should be focused on improving the overall process flow which does not end until the lot leaves Sort. Given the similarities between the Fab and Sort organizations, it would not seem difficult to integrate the Sort organization into the Fab hierarchy. Sort 9/11 is a special case which enjoys some synergy and economies of scale from combining two Sort floors. These benefits need to be traded off against the detriment to the improved information flow to upstream operations. If reexamination shows that a separate organization is more beneficial than a combined one, then incentives should be put in place to better integrate the two sections of the process. The Fab and Sort organizations need to be working together to improve the overall process flow.

8. Summary

The results of the project analysis were recommendations regarding improvements to Sort operations supporting the goal of decreased capital expenditure through increased equipment utilization. Although the project was focused on the Sort tester equipment set, the results are easily applicable to the management of other equipment areas.

Summary derivation of the limiters to increased equipment utilization:

1. Utilization refers to the time during which equipment is solely processing incoming WIP (Work In Process).
2. Overhead refers to the time during which other activities are consuming equipment capacity because they are not value-added equipment processing activities and, therefore, are effectively preventing incoming material from being processed.
3. Excess capacity is necessary for the occasional large quantities of arriving WIP ("WIP bubbles").
4. Since this excess capacity is operated infrequently, it lowers (and limits) the average utilization level.
5. When WIP bubbles arrive, the equipment will operate at its maximum attainable utilization level.
6. The Overhead activities limit the maximum attainable utilization level.
7. Operational systems need to be put in place to reduce (1) the size of the WIP bubbles and (2) the duration of the Overhead activities.

Recommendations to address the limiters to increased equipment utilization:

- Reduce the size of the WIP bubbles by smoothing the total expected processing time of the start material.
- Reduce the amount of equipment capacity lost to Overhead activities by sustaining a continuous improvement work environment. Maintain the pressure to improve by flexing equipment capacity to match the amount of capacity that should be needed given the incoming workload.

Fundamental enablers of the recommendations:

- Improve the data collection to accurately quantify utilization and Overhead Time.
- Elevate utilization as a priority within the organization by setting utilization goals and continually reviewing the progress towards those goals.

Section 6.3 presented an implementation model for Sort 9/11 which can be easily transferred to other organizations. Increasing equipment utilization throughout Intel's manufacturing sites would substantially reduce capital expenditures on new processing technologies. These cost reductions would improve Intel's competitive position in the microprocessor industry and continue the profitability and growth enjoyed during the last several years.

Works Consulted

Bitran, Gabriel R. and Devanath Tirupati. "Tradeoff Curves, Targeting and Balancing in Manufacturing Queueing Networks." Operations Research 37.4 (1989): 547-64.

Burman, M. H. "A Survey of Current Issues in Production Planning Research." Laboratory for Manufacturing and Productivity: LMP-93-003. MIT (1993): n. pag.

Fine, C. and L. Wein. "Managing Capacity and Throughput in Environments Subject to Variability." material from Introduction to Operations Management course.

Fine, Charles H. and Stephen C. Graves. "A Tactical Planning Model for Manufacturing Subcomponents of Mainframe Computers." Journal of Manufacturing Operations and Management 2 (1989): 4-34.

Graves, Stephen C. "A Tactical Planning Model for a Job Shop." Operations Research 34 (1986): 522-33.

Kleinrock, Leonard. Queueing Systems. 2 vols. New York: Wiley, 1975.

Maister, David H. "Note on the Management of Queues." Boston: HBS Case Services, 1979.

Nahmias, Steven. Production and Operations Analysis. 2nd ed. Boston: Irwin, 1993.

Nakajima, Seiichi, ed. Introduction to TPM. Cambridge: Productivity Press, 1988

Nakajima, Seiichi, ed. TPM Development Program. Cambridge: Productivity Press, 1989

Raghavachari, M. "Queuing Theory." Salvendy 13.7.

Salvendy, Gavriel, ed. Handbook of Industrial Engineering. New York: Wiley, 1982.

Schein, Edgar H. Organizational Culture and Leadership. 2nd ed. San Francisco: Jossey-Bass, 1992

Stecke, Kathryn E. "Machine Interference: Assignment of Machines to Operators." Salvendy 3.5.

Whitt, Ward. "Approximations for the GI/G/m Queue." Production and Operations Management 2.2 (1993): 114-161.

Whitt, Ward. "Understanding the Efficiency of Multi-Server Service Systems." Management Science 38.5 (1992): 708-23.