Proceedings of the 1991 Winter Simulation Conference
Barry L. Nelson, W. David Kelton, Gordon M. Clark (eds.)

AN APPROACH TO MODELING LABOR AND MACHINE DOWN TIME IN SEMICONDUCTOR FABRICATION

Susan S. Baum
Cheryl M. O'Donnell

NCR Microelectronic Products
2001 Danfield Ct.
Fort Collins, Colorado 80525

ABSTRACT

Simulation has become an increasingly popular analysis tool in the semiconductor industry. However, labor resources and equipment downtime have proven difficult to implement correctly in a semiconductor fabrication model. This paper presents a unique approach to modeling labor resources, and emphasizes proper techniques for modeling equipment breakdowns and scheduled maintenance.

1 INTRODUCTION

The semiconductor manufacturing process is extremely complex, consisting of hundreds of process steps and utilizing a large array of equipment and labor resources. A typical Application Specific Integrated Circuit (ASIC) requires over 200 operations, and visits 50-70 different machines, some several times throughout its route. The operation of any wafer fabrication facility (fab) is highly dependent on both labor and equipment. To model a fab ignoring either labor constraints or equipment downtime could misrepresent cycle times, throughput rates, and queues.

The variety of equipment in a wafer fab creates the need for a highly skilled labor force, characterized by a high level of cross-training, and the ability to monitor multiple machines at one time. In addition, because it takes from two to six weeks to train a new operator, depending on the nature of the task, significant lead time is required in making staffing decisions. A ramp up in production cannot be met by adding operators during the peak production period; it must be planned for well in advance. Determining the optimal number of operators trained in specific skills can be a key factor in meeting production requirements.

The equipment required for wafer fabrication is capital intensive, often ranging from several hundred thousand to a million dollars for a single piece of equipment. This equipment will exhibit both random and scheduled downtime. For example, a wafer may break inside a machine resulting in a 'jam' which would halt production on the machine. If machines are down often, the ability to meet production requirements will again be limited.

This paper presents an approach to modeling labor resources, random equipment failures, and preventative maintenance in a wafer fabrication facility, and addresses their inherent complexities. This approach was implemented in a simulation model of an ASIC wafer fabrication facility, developed using the SIMAN[1] simulation language.

2 CONCEPTS AND TERMINOLOGY

This section defines several key concepts and terms used in describing the labor and machine failure models. We define a **lot** to be a collection of wafers which require identical processing steps. The lots are the main entities used in our model. Lots are often grouped into a **batch**, when required to efficiently load a piece of equipment. For example, in loading a furnace which can hold 200 wafers, several lots may be batched together. Each lot of wafers follows a **route** though the fab, consisting of a sequence of process steps. The route specifies which workstations to visit, and the process plan to be used. A **workstation** is defined as a group of one or more identical, interchangeable machines that share the same reliability and throughput characteristics [Najmi]. Individual machines within a workstation can have different states (busy or idle, up or down) at any given time. Examples of workstations include identical aligners, develop tracks, or furnace tubes. Machines are classified by **machine type**, and belong to a designated **fab area**. The machine types used in our model are Basic and Wet-bench, depending on how the machine processes wafers. These classifications are important in

modeling labor delays. The four fab areas are Photolithography, Etch, Thin Films, and Diffusion.

There are several terms associated specifically with labor modeling. A **man-machine-ratio (MMR)** represents the number of machines an operator can monitor at one time. For example, an MMR of 1:4 signifies that one operator can monitor up to 4 machines at one time. A **machine group** is a grouping of stations for the purpose of labor modeling. All machines in a machine group have the same MMR and are in close proximity. An operator at one machine can monitor any other machine in the machine group if the MMR has not been exceeded. All machines with a 1:1 MMR are placed in a single machine group. This is a strategy used to reduce the number of machine groups in the model, while still meeting the criteria for a machine group. A **labor group** is a group of identical, interchangeable labor resources capable of operating any machine which requests labor from this group. Each station (group of like machines) has a labor group assigned to it. Figure 1. illustrates the relationships between these defined categories.

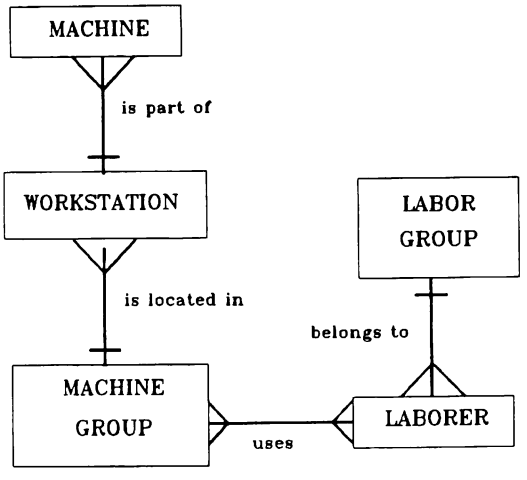

Figure 1 – Entity Relationships

We defined a **use variable** to count the number of machines in a machine group which are currently using an operator. These counts are associated with each machine group and are stored in a global array. In Siman this was done using the global array variable D(X). The number is used to determine if a labor resource is required as each additional machine requires an operator.

Two terms used in connection with machine scheduled and unscheduled downtime modeling are **mean-time-between-failures (MTBF)** and **mean-time-to-repair (MTTR)**. The MTBF is the average interval of time a machine is up before it exhibits a breakdown. This can include both busy and idle time. The MTTR is the average interval of time a machine is down before the repair is complete. This includes both time spent waiting for a repair person to arrive and the actual repair time.

3 LABOR MODELING

Scope

A simple labor/machine model consists of an operator running a single machine with a 1:1 Man-Machine-Ratio. The operator is busy for the entire processing time, and is idled when the machine is idled. In a wafer fabrication facility, several complexities are introduced. The first is the high level of training required to perform each task. While cross training is a necessity, it is unrealistic to have all operators qualified to run all machines. It is more often the case that an operator is trained in several stations within a designated fab area, i.e., etch, diffusion.

A second complication is introduced by the various Man-Machine-Ratios which exist in a fab. The MMR may be 1:1 for an ion implanter, 1:2 for photolithography aligners, and 1:9 for develop tracks. Therefore, rules used to determine when an operator is needed must be machine specific. Furthermore, if there are two machines, each with a 1:2 MMR, they must be located in close proximity to each other for it to be feasible for an operator to monitor both machines at once.

The staffing of operators in each fab area for all shifts also must be addressed as there is not a consistent number of operators in each area during all shifts. The model must be capable of adjusting available resources for each shift.

Methodology

A lot (entity) flows through the model by following a sequence of processing steps designated as a route. When a lot arrives at a processing station, it waits for an available machine and seizes it. As soon as the machine is seized, the model must determine if labor is required. Several pieces of information are required: the station machine group, the machine type, the station Man-Machine-Ratio, the station labor group, and the use variable for the machine group. For all Furnaces and Basic machines, the following calculation is used to determine if it is necessary to seize an additional laborer:

FLAG = modula(D(**count**) , **MMR**)

If FLAG = 0, there is either no labor present at that

machine group, or the laborer(s) are already monitoring the maximum number of machines allowable according to the MMR for the machine group. Therefore, when Flag = 0, a labor resource must be seized. If Flag > 0, there is already a laborer present who has the capability to monitor an additional machine in the machine group, and it is not necessary to seize another labor resource. The flow of the labor model is depicted in Figure 2.

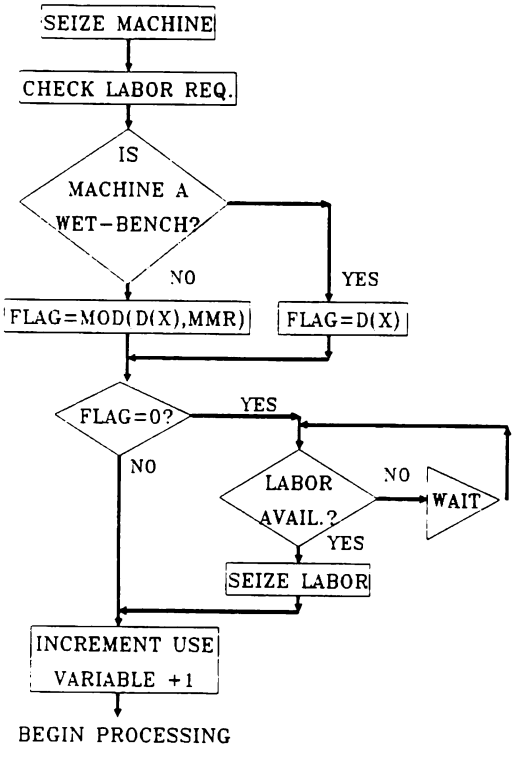

Figure 2. Labor Model Flow

Labor for a Wet-benches requires a unique calculation because this type of equipment behaves like a "batch conveyor". Batches take a fixed duration for processing through the machine, yet after a specified lag time and before one batch is done, a second batch can be started in the machine. A Wet-bench in a fab consists of a series of tanks filled with different chemicals into which small batches of lots are successively dipped and held for predetermined time periods [Najmi]. A second batch can start processing at a wet-bench as soon as the first batch is out of the first tank. Because more than one batch can be actively processing at one bench, the previous equation is not accurate in this case. Instead, we need only to know if any lots are present at the workstation(s) that a single operator is monitoring. Therefore, the equation is:

FLAG = D(**machine_group**)

If FLAG = 0, there are no lots present, and therefore no labor present. A labor resource must be seized to process the lot. If FLAG > 0, there is already an operator present. Traditional wet-benches require an operator to be present during the entire time a lot is being processed.

When labor is required, it must be seized from the appropriate labor group use at that workstation. The labor groups in this model are broken down by fab areas: Diffusion, Thin Films, Etch, and Photolithography. If a labor resource from the appropriate group is not available, the lot entity will wait in a queue associated with the labor group until an operator becomes available.

When the operator is seized from the labor pool, he is delayed for the total operator delay required. The operator-time-per-wafer and operator-time-per-batch fields represent the number of minutes the operator is required to be present at each station to process wafers. These values are stored as attributes of each entity, and are updated by the SIMAN SEQUENCE element. The delay time per batch occurs once for each batch at a station. The operator delay time per wafer is multiplied by the number of wafers in each batch. These numbers are then added to calculate the total operator delay. The model allows the operator delay to be different from the machine delay. For example, during a furnace run, the machine may be delayed for 8 hours, but the operator is only delayed for 30 minutes.

When the operator delay is over, the previous calculations are again used to determine if the labor resource may be released. The global use variable is decremented, and FLAG is calculated. If FLAG = 0, operator is released. If FLAG >0, the operator is retained.

Alternate Approaches

Several other approaches to the labor modeling problem were analyzed, but found inappropriate. One method was to divide the operator time required at each station by the MMR. For example, if an operator was required for 60 minutes at a workstation with an MMR of 1:2, then the actual time the operator is delayed is 30 minutes (60 minutes divided by 2). Using this approach, a bottleneck piece of equipment could be kept waiting for 30 minutes for an operator who is actually available to watch more than one machine. This delay could happen many times during the day creating an artificial backlog at a critical piece of equipment. Likewise, if machine groups are not used, the model could incorrectly allocate an operator to two machines which are not physically located close

together. If labor groups are not used, it implies that an operator is trained on every piece of equipment in the fab, which is unrealistic. Lastly, labor modeling could be disregarded altogether. In our experience we found that modeling a fab without labor constraints misrepresented cycle times and throughput rates.

4 EQUIPMENT DOWNTIME MODELING

Scope

Just as labor plays a significant role in a fab simulation, modeling of equipment downtime is also necessary to ensure model validity. There are two types of equipment downtime modeled: unscheduled breakdowns and scheduled preventive maintenance. Downtime due to misprocessing, jams, and contamination occur at unplanned times. To help ensure less unplanned, random machine failures, management may impose a preventive maintenance schedule. By checking the equipment at various intervals and performing routine maintenance, equipment uptime performance can be improved. Individual pieces of equipment should be modeled independently, allowing each machine to go down according to its own random schedule. Both unscheduled (breakdowns) and scheduled (preventive maintenance) downtime should be modeled to represent an accurate picture of fab.

Methodology

A collection of entities are created to control equipment failure and repair intervals. The number of entities created is equal to the total number of stations existing in the fab. Each station may contain more than one machine. Therefore, the number of failure entities must be duplicated to accurately represent the number of machines in the fab. This is accomplished using the SIMAN DUPLICATE command.

The scheduled and unscheduled downtime routines are modeled separately within the machine downtime subroutine. Thus, each failure entity is again duplicated. The SIMAN BRANCH command allows one entity to handle scheduled maintenance activities and the other to handle random breakdowns. Figure 3 illustrates the entity duplication process using an example of an implanter station consisting of two like implanter machines. Each of these implanters experiences scheduled and unscheduled downtimes. Therefore, to adequately model the implanters' downtime, four failure entities are required.

To model unscheduled breakdowns, failure entities are assigned distributions corresponding to their workstation. The distributions are used to calculate the intervals between failures and time to repair. Some failures and repair times are modeled deterministically when insufficient data exists to determine the distribution. When the failure entity signifies that it is time for a particular machine to go down, that failure entity seizes, with priority, one machine in the workstation (group of like machines). The number of available machines within that workstation is altered to reflect a decrease in the number that are now available. The machine is then delayed by another calculated distribution or mean of its MTTR. After the delay, the number of machines in the workstation will be altered again to show that the machine is once again available.

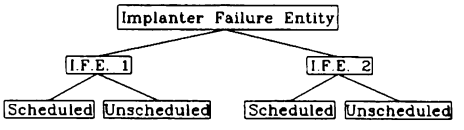

Figure 3 - Duplication Process of Implanter Failure Entities

Failure entities are also created to model preventive maintenance schedule. A machine remains up until the time for its scheduled maintenance. At this time, the failure entity seizes the workstation and alters the number of machines available. After a delay of its MTTR, the number of machines available is altered and the machine is again available, indicating that preventive maintenance is complete. The entities will continue to control the scheduled and unscheduled machine downtime events, taking machines down and bringing them back up, for the duration of the simulation.

If a machine is considered up all the time or data is not accurately maintained on a machine, that machine may not have a MTBF or MTTR. If this is the case, then the failure entity should be disposed, as it is not performing any function in the model.

Alternate Approaches

In order for a model of machine failure to accurately represent a fab, extensive time must be spent on gathering the appropriate data. Overall percentage uptime or downtime for each machine is not sufficient. The historical equipment data for machine up and down time must be used to accurately model equipment breakdowns. The following examples illustrate the inadequacies of knowing or using only overall average percentage uptime or downtime for each machine.

Consider a fab which has a station consisting of two aligners. The historical data indicates that they are up for an average of 16 hours and down for an average of 4 hours. This is an average uptime percentage of 80% for the aligners. This scenario is represented in Figure 4.

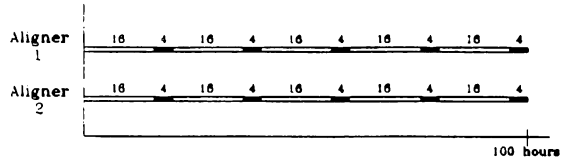

Figure 4 - Aligner Example with 80% Uptime

The simulation analyst may decide to simulate the aligner scenario by taking the total average up and down hours and dividing by the number of machines in that station, i.e. 16 hours uptime divided by 2 aligners results in one aligner being up for 8 hours and down for 4 hours and the second aligner being up for 100% of the time. This scenario is illustrated in Figure 5.

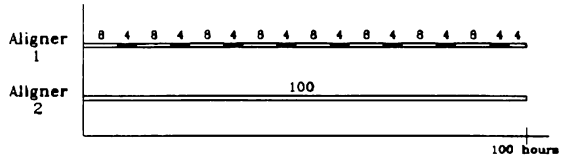

Figure 5 - Aligner Example with 84% Uptime

The above two scenarios are not statistically equivalent. The average uptime in the second scenario is calculated by noting that the first machine is up 68 hours and down 32 hours. Thus, the first aligner is up 68% of the time and the second aligner is up 100% of the time. The overall average uptime for the aligner station in the second scenario is 84%, which does not equal the 80% uptime achieved in the first scenario. Another problem associated with the second scenario is that there is never the possibility of both machines being down simultaneously. This is not realistic and may result in queues being smaller than they should be.

If the simulation analyst were to decide to only use his knowledge of the percentage uptime in his code he may be inaccurately representing the true uptime and downtime. For example, one machine may have an MTBF of 80 hours and an MTTR of 20 hours. A second machine may have an MTBF of 8 hours and a MTTR of 2 hours. Both these machines have an average uptime of 80%. Yet, it is obvious that these two machines are dissimilar. If there were two of the machines that were up 80 hours and down 20 hours and both were to break down at the same time, a very large queue could be built up waiting 20 hours for these machines to come back up. The queue may not grow as large if the machines had an average of 8 hours up and 2 hours down.

Thirdly, it is inaccurate to model unscheduled machine breakdowns by reducing the processing rate by the percentage of a machine's downtime. For example, if it was known that the downtime for a particular machine was 10%, it would be incorrect to simply reduce the processing time of that machine by 10% to account for the downtime. Once again this stresses the importance of knowing both the MTBF and the MTTR. Only knowing the overall percentage of downtime does not give the true picture of a machine's downtime. Therefore, to simply reduce a machine's processing time by its downtime percentage will affect both cycle times and queues [Law, 1990].

Statistical Analysis

As illustrated in the three examples in the previous section, it is important to obtain as much data about each machine as possible. The value of the data must be weighed against the time spent gathering it. It is necessary not only to obtain overall downtime or uptime percentages, but to gather more detailed data. The type of data found in machine uptime logs can often be used to fit a distribution.

After the individual data points are collected, those points are then divided into intervals and a histogram is obtained by plotting the intervals. The histogram suggests what the possible distribution may be. Figure 6 illustrates a histogram which appears to demonstrate exponential characteristics for the unscheduled uptime of a machine.

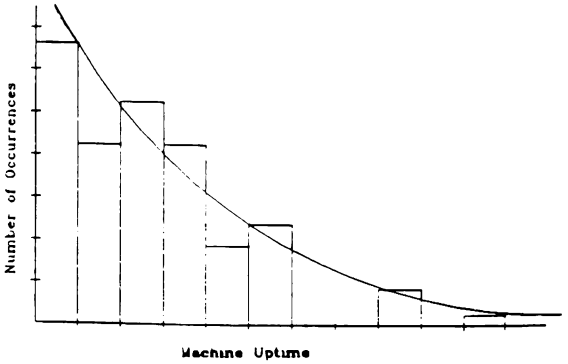

Figure 6 - Machine Uptime Following an Exponential Curve

To test the hypothesis that the histogram shown in Figure 5 is truly exponential, the Kolmogorov-Smirnov (K-S)

goodness of fit test was performed. First, the estimated Beta parameter was found using the maximum likelihood estimator (M.L.E.). In the case of the exponential distribution, this is the mean. The distribution function was then calculated for each individual data point in the sample. The K-S test measures the greatest difference D_n between the distribution function and the data points in the sample. The test statistic was calculated using the formula described in Law and Kelton (1990), as follows:

$$D_n^+ = \max_{1 \le i \le n} \left\{ \frac{i}{n} - \hat{F}(X_{(i)}) \right\}, \quad D_n^- = \max_{1 \le i \le n} \left\{ \hat{F}(X_{(i)}) - \frac{i-1}{n} \right\}$$

where $D_n = \max\{D_n^-, D_n^-\}$

The null hypotheses H_0 is rejected if D_n exceeds some critical value, determined using a specified alpha level of uncertainty.

Often the Weibull and the Gamma are good distributions for MTBF and MTTR respectively [Law, 1990]. A distribution is appropriate when modeling up and down time if the data can be obtained. A theoretical distribution is a more exact fit of the machine up and down times with an empirical distribution being less exact and a mean even less [Law, 1990]. However, even a mean with specific up and down time data is better than using an overall percentage of the machine's downtime. A distribution instills the necessary randomness for machine breakdowns. Consequently, when only a mean time is used for MTBF and MTTR, an offset time can be used so all the machines in a station do not go down simultaneously. It is important to have the offset time be random otherwise the possibility of more than one machine being down at a given time is eliminated.

When gathering data, the simulation analyst must be careful to gather only appropriate and accurate data. For example, if a furnace broke down Monday night and it was scheduled for a preventive maintenance on Tuesday, it would make more sense to correct the initial problem and then to perform the preventive maintenance work immediately following. This would be better than bringing that same machine down again in less than 24 hours. However, the simulation analyst needs to know that when the furnace was down for 4 hours that 3 hours were fixing the unscheduled breakdown and the last hour was performing the preventive maintenance. When calculating that furnace's unscheduled downtime distribution, that data point should be 3 hours rather than 4 hours, because one hour of the downtime is already being modeled in the preventive maintenance routine.

5 CONCLUSION

A working model of an NCR wafer fabrication facility incorporates the concepts described in this paper. The important features of the model are its ability to accurately model labor, separately from pure equipment processing time, and to model both scheduled and unscheduled maintenance activities.

The labor model considers operator skills, man-machine-ratios, and proximity of resources being monitored. This allows the simulation analyst to consider the effects of changes in production starts on labor requirements. The impact of more or less cross training can be evaluated, as well as the impact of dedicating labor resources to specific equipment. The equipment downtime model accurately reflects the impact of random machine failures, when the data is available. This allows the analyst to evaluate the impact on cycle times and throughput of improving the uptime performance of a machine. The effect on throughput with changes to the preventive maintenance schedule can also be evaluated. Both labor and equipment downtime modeling have been critical elements of our modeling success.

1. Siman is a registered trademark of Systems Modeling Corporation.

REFERENCES

Law, A.M., "Models of Random Machine Downtime for Simulation", *Proceedings of the 1990 Winter Simulation Conference*, December 1990.

Law, A.M. and W.D. Kelton (1990), *Simulation Modeling and Analysis*, Second Edition, McGraw-Hill, New York, NY.

Najmi, A. and S.J. Stein, "Comparison of Conventional and Object-oriented Approaches for Simulation of Manufacturing Systems", *Proceedings of the IIE Integrated Systems Conference*, November, 1989.

Pegden, D.C. (1986), *Introduction to SIMAN*, Systems Modeling Corporation, State College, PA.

AUTHOR BIOGRAPHIES

Susan Baum is Project Leader for Advanced Manufacturing Engineering at NCR Microelectronic Products Division in Fort Collins, Colorado. She received an M.S. in Industrial and Systems Engineering from the Georgia Institute of Technology and a B.S. in Industrial Engineering from the Pennsylvania State University. She is currently working on methods and tools to support decision making in the semiconductor manufacturing and test environments.

Cheryl O'Donnell is an Industrial Engineer in Advanced Manufacturing Engineering at NCR Microelectronic Products Division in Fort Collins, Colorado. She received her B.S. in Industrial and Management Engineering from Montana State University. She is currently working on discreet event simulation models of semiconductor manufacturing and decision support tools for production control.