Proceedings of the 2005 Winter Simulation Conference
M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds.

OPTIMIZING ROBOT ALGORITHMS WITH SIMULATION

Todd LeBaron

Simulation Consulting
Brooks Automation
5245 Yaeger Rd
Salt Lake City, UT 84014, U.S.A.

Joerg Domasche

Simulation Manager
Infineon Technologies AG
Konigsbrucker Strasse 180
01099 Dresden, GERMANY

ABSTRACT

Maximizing equipment throughput on multi-chambered cluster tools is an ongoing objective for semiconductor fabs. The increasing use of dual-armed robots and the need to process multiple products simultaneously complicates this objective. Typically, when a new processing technology is introduced, one chamber inside the tool is dedicated to the new process, while the other chambers are assigned to run normal production wafers. This results in multiple wafer flows or "parallel routes" within the tool. Determining and implementing optimal robot schedulers to efficiently handle the complexities within the tool is key to maximizing equipment throughput. This paper introduces the components of a multi-chambered cluster tool and discusses how simulation was used at Infineon to develop, test, and optimize efficient wafer selection rules. Several real-world cases are detailed and reported.

1 INTRODUCTION

Cluster tools come in many different configurations but basically include similar components. A cluster tool consists of one or more vacuum frames surrounded by processing chambers, one or more load locks, and one or more robots, as shown in Figure 1. The main vacuum frame is typically evacuated to purge contaminants and improve deposition conditions. The processing chambers attach to the vacuum frame and are typically isolated from the vacuum space by slit valves. A load lock is a temporary chamber in which wafers can be moved from atmosphere to vacuum pressure prior to processing, and back to atmospheric pressure after processing. One or more vacuum robot reside inside the vacuum frame and are used to move wafers between the load locks and processing chambers. Atmospheric robots reside inside the equipment front end module (EFEM) and are used to move wafers within this area.

In normal 300 mm fabs, wafers are transported to cluster tools inside front opening unified pods (FOUP's). A FOUP typically holds 13 or 25 wafers. A FOUP is loaded into one of the load ports in front of the tool, and will remain at the tool until all wafers complete the processing requirements. The EFEM consists of an atmospheric robot (and possibly an aligner and/or buffer chamber), and is responsible for moving wafers between the respective FOUP and the load locks. An atmospheric robot may service up to four load port positions. Figure 1 illustrates the basic components of a cluster tool.

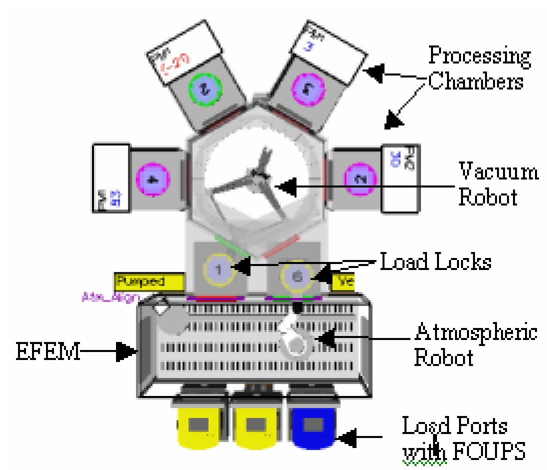

Figure 1: Cluster Tool Configuration

After a FOUP has been loaded at a load port, wafers begin their route sequence. Wafers are removed from the FOUP (one at a time) and placed in one of the load locks by the atmospheric robot. Once the load lock is filled to capacity, it is then pumped down to reach the vacuum pressure. Each wafer is then removed from the load lock by the vacuum robot, and placed inside a processing chamber. Each wafer will follow a route, or flow, defining the sequence and duration of processing within the respective chamber. Inside a processing chamber, material is added to the wafer, or the wafer is enhanced in some way through a series of targets and deposition technology. Processing chambers may also have vacuum requirements for pumping and venting to alter the chamber pressure and temperature. Slit valves are used to isolate the processing chambers and load locks from the vacuum space. These components, working together, constitute the integrated wafer processing system of cluster tools

2 PARALLEL FLOWS

Typically, when a new processing technology is introduced, one chamber inside the tool is dedicated for the new process, while the other chambers are assigned to run normal production wafers. This results in a tool that processes multiple products at the same time, or parallel product flows.

Consider the cluster tool configuration shown in Figure 2. This tool has two load locks (LLA, LLB) and four processing chambers (PM1, PM2, PM3, and PM4). Chambers PM1, PM2, PM3 are set up to process normal production wafers for 60 seconds. Chamber PM4 is set up to run the new processing technology which requires 120 seconds.

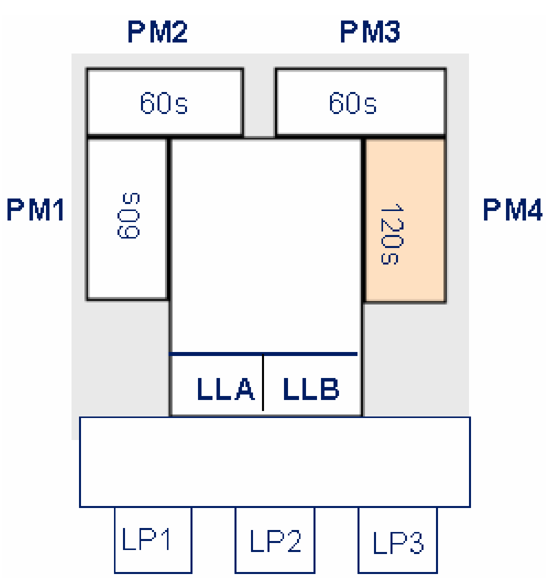

Figure 2: Cluster Tool Layout

Three load ports (LP1, LP2, and LP3) are positioned in front of the tool. FOUPS containing either Product1 or Product2 are loaded into the load ports. When both Product1 (which runs Process1) and Product2 (which runs Process2) are loaded at a load port and ready to process, then the tool is running parallel flows. In addition, each wafer can take many different paths through the tool depending on its respective route, as shown in Figure 3.

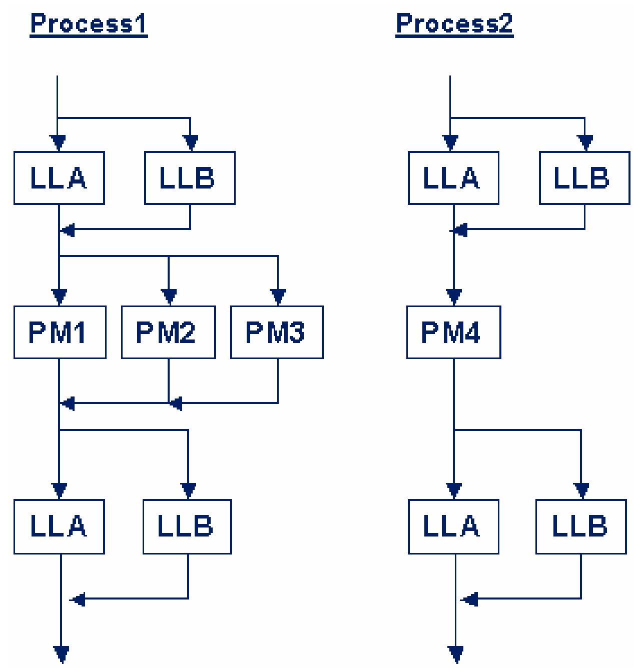

Figure 3: Product Routing Possible Paths

Figure 3 assumes that the load locks are not dedicated to a specific product. Wafers may enter the tool at either load lock LLA or LLB. Product1 wafers will then process at the next available chamber (PM1, PM2, or PM3). Product2 will process at PM4. Once processing is complete, the wafer will return through the next available load lock (LLA or LLB). This example illustrates parallel flows as well as parallel chambers. The scheduling behind parallel flows is complex and will be discussed in the following sections.

3 CLUSTER TOOL SCHEDULER

The cluster tool scheduler (CTS) is the actual software used to schedule the wafer move sequence within the cluster tool. The throughput of the tool can be greatly affected by the efficiency of the scheduler. There are typically two types of schedulers used on multi-chambered cluster tools.

The first type is commonly referred to as a “search optimization method” and is executed before any processing starts. The CTS is fed the tool configuration, and the product flow (or recipe), along with a specified search depth. The CTS program then uses a branch and bound search algorithm to explore all move possibilities and future resulting moves forward to the specified search depth. The move that initiates the sequence that yields the highest process chamber utilization within the search horizon is output to the system and the algorithm is repeated. The actual control software will then use the resulting move sequence to schedule chambers and sequence the wafers while the tool is operating. This type of scheduler will also re-calculate the move sequence if certain events occur, such as a chamber going down or the loading of a new FOUP.

The second type of scheduler is the rule-based scheduler. The rule-based scheduler reacts to conditions in the tool and uses pre-defined logic to determine which wafer to move next. The rule-based scheduler is faster to initialize because it doesn't need to calculate or recalculate the wafer move sequence. However, it is limited by the defined rules, and may not be optimized for specific routes.

The simulation model presented in this paper uses a rule-based scheduler. The rule base scheduler comes with the ability to assign each robot a common scheduling algorithm (such as “push”, “pull”, “closest”, “fifo”, “chamber_priority”). The scheduling algorithm is used to determine which wafer a robot will “pick” or “place”. In addition, user-defined algorithms can be defined, assigned to a robot, and tested. The ability exists to prioritize different “pick” and “place” criteria, and have the robot use this criteria when deciding which wafer it should pick or place.

4 SCHEDULING COMPLEXITIES

Scheduling deals with the logic used to determine when equipment events are triggered, such as when to pump and vent the load locks, open the slit valves, and more importantly, which wafer to move and when to move it. Scheduling applies to the robots that are responsible for moving wafers between the FOUP’s, load locks, and processing chambers. A move consists of two separately scheduled events: a “pick” and a “place”. A “pick” is an event in which the robot extends inside a chamber, lifts the wafer from it’s position, and retracts with the wafer. A “place” is an event in which the robot extends with the wafer into a chamber and places it on the respective chamber position or slot. On a single ended robot, the “pick” must always be followed by the “place”. On dual-ended robots, there is more flexibility that allows for optimization in scheduling.

Several factors contribute to the complexity of the cluster tool scheduler and must be taken into account when developing scheduling rules to determine optimal equipment performance:

- Multiple product flows
- Chamber clean cycles
- Dual ended robots
- Load lock functions
- These factors are discussed in the following sections.

4.1 Multiple Product Flows

Many different products can potentially run in the tool at the same time. When more than one product is loaded and running, the wafer start sequencing can have a significant effect on equipment throughput. The EFEM robot is responsible for pulling wafers from the FOUP and loading them into the load lock. In a system where the load lock has limited capacity, it is critical that the scheduler picks the correct product to start. If not, chamber starvation (idle chambers), may result. In addition, other chambers may be backlogged by filling the load locks with wafers that can not readily be processed. These conditions will certainly result in decreased throughput.

4.2 Chamber Clean Cycles

On deposition and etch process tools, it is common for the chambers to execute a self-clean cycle. A clean cycle is used to purge impurities and other residual material from the chamber. The chamber must be empty during the clean cycle. The frequency and duration of clean cycles are dependent on the process; however, it is not uncommon to clean after processing every wafer.

The robot scheduler must take clean cycles into account. The scheduler should be able to look ahead to avoid pre-positioning the robot in front of a chamber that will be unavailable due to a clean cycle requirement. The scheduler should take clean cycle times into account when deciding which wafer to start next to avoid loading a load lock with a wafer that will not be the next product that can start processing inside a chamber. Also, the scheduler should take clean cycles into account when determining the optimal time to start the pump and vent cycles, sometimes deciding to vent with an unprocessed wafer still occupying a load lock slot.

4.3 Dual Ended Robots

When a dual-ended robot is used, additional scheduling factors need to be considered. Dual-ended robots can use one end of the robot as a temporary wafer buffer. This allows the robot to pre-position in front of a processing chamber waiting to “swap” wafers once processing is finished. It can also be used as a buffer position to hold a wafer whose next position is not yet available, but whose previous chamber could start its next scheduled task once the wafer is removed. Examples of this include removing a wafer from a load lock so that it can start the vent cycle, or removing a wafer from a chamber so that it can start a clean cycle. The scheduler must make smart decisions with regards to pre-positioning and when to commit one blade as a wafer buffer, otherwise throughput will decrease.

4.4 Load Lock Functions

Equipment scheduling also includes deciding when to pump and vent the load locks. A load lock that is in the process of pumping or venting can not be accessed by either robot. In addition, the vacuum robot can access the load lock only when it is in the "pumped" condition. The atmospheric (EFEM) robot can only access the load lock when it is in the "vented" state.

5 FLEXIBLE SIMULATION SCHEDULER

The simulation model was built with the ability to define and rank different "pick" and "place" event criteria, and assign the definition as an algorithm for the robot to use. An Excel front end facilitates the defining and assigning of custom algorithms.

As the simulation model is running, the robot makes decisions as to what it will do next based on current conditions inside the tool. The robot's job is to move wafers between chambers, load locks, and load ports. When there is nothing for the robot to do, it goes idle. When there is one or more wafer ready to be picked up, or one or more wafer on the robot waiting to be placed, the robot will use the assigned algorithm to determine what to do next.

The robot will rank all of the possible wafers that are ready to be picked up according to the "Pick Criteria" values associated with the algorithm. "Pick Exceptions" are also used to adjust the score as they apply to each wafer. In addition, each wafer (if any) that is on the robot waiting to be "placed" is ranked according to the "Place Criteria" and "Place Exceptions". The wafer with the highest possible score will be the next wafer the robot "picks" or "places". After the robot makes the required "pick" or "place", the robot re-evaluates the conditions in the tool, assigns scores to each wafer, and continues with the next task as dictated by the algorithm (scheduling rules).

There are 20 different "Pick Criteria" events to which values can be assigned. Not all events are applicable to every case. Examples of "pick criteria" include events such as "The Wafer is in the load lock and it's destination chamber is available", and "The Wafer is finished processing and the chamber will execute a clean cycle after it is removed."

There are five different "Place Criteria" events to which scores can be assigned. Examples of "Place Criteria" include "Wafers destination chamber is a bottleneck chamber" and "Wafers destination chamber is the priority chamber".

When wafers are being scored, if the criteria is true for the wafer then the score is increased by the value assigned to the criteria. This design provides great flexibility and speed in developing and optimizing "wafer selection rules."

6 SIMULATION SCENARIOS AND CASE RUNS

Two Simulation scenarios were set up and run to determine the effects of different tool configurations and scheduler options. Equipment throughput, given in wafers per hour, will be used as the key performance metric. Both scenarios use the same fixed parameters with the exception of load lock assignment. Scenario 1 uses dedicated load locks for each route (LLA is used for Product1, LLB is used for Product2). Scenario 2 assigns load locks based on availability.

6.1 Scenario 1 – Dedicated Load Locks

Each case will process two products (product1 and product2). Product 1 is assigned LLA and Product 2 is assigned LLB. Table 1 lists the fixed configuration values used for scenario1 case runs.

Table 1: Scenario1 Fixed Parameters

Scenario 1 Fixed Parameters	Product 1	Product 2
Number of Load Lock Slots	2	2
Align Time	4 sec	4 sec
Processing Chambers	PM1 or PM2 or PM3	PM4
Processing Time	60 sec	120 sec
Chamber Clean Frequecy	After every 1 wafer	After every 1 wafer
Chamber Clean Time	53 sec	109 sec
Load Lock Assignment	LLA	LLB
Load Lock Pump Time	14 sec	14 sec
Load Lock Vent Time	12 sec	12 sec

The Scenario will include several case runs in which different algorithms and robot types are evaluated. Table 2 identifies the case experimental parameters.

Table 2: Scenario1 Case Run Matrix

Scenario1: Dedicated Locks		Vacuum Robot
		Wafer Selection Rule
Case Run	Vacuum Robot Type	(Algorithm)
Case1	Dual_Same (Leapfrog)	Pull
Case2	Dual_Same (Leapfrog)	Push
Case3	Dual_Same (Leapfrog)	Prioritize_loadlock
Case4	Dual_Same (Leapfrog)	Prioritize_Chamber
Case5	Dual_opposite (Bisymmetric)	Pull
Case6	Dual_opposite (Bisymmetric)	Push
Case7	Dual_opposite (Bisymmetric)	Prioritize_loadlock
Case8	Dual_opposite (Bisymmetric)	Prioritize_Chamber

Appendix A provides a brief description of the various wafer selection rules (algorithms) that were defined. Appendix B provides a description of the vacuum robot types that were used.

6.2 Scenario 2 – Products Use Either Load Lock

Scenario 2 is identical to scenario 1, with the exception of the load lock assignment. Wafers run under scenario 2 can pass through either LLA or LLB, depending on availability. Table 3 lists the fixed configuration values used for scenario2 case runs.

Table 3: Scenario 2 Fixed Parameters

Scenario 2 Fixed Parameters	Product 1	Product 2
Number of Load Lock Slots	2	2
Align Time	4 sec	4 sec
Processing Chambers	PM1 or PM2 or PM3	PM4
Processing Time	60 sec	120 sec
Chamber Clean Frequecy	After every 1 wafer	After every 1 wafer
Chamber Clean Time	53 sec	109 sec
Load Lock Assignment	LLA or LLB	LLA or LLB
Load Lock Pump Time	14 sec	14 sec
Load Lock Vent Time	12 sec	12 sec

The Scenario will include several case runs in which different algorithms and robot types are evaluated. Table 4 identifies the case experimental parameters used for scenario2.

Table 4: Scenario 2 case Run Matrix

Scenario2: Shared Locks		Vacuum Robot
		Wafer Selection Rule
Case Run	Vacuum Robot Type	(Algorithm)
Case1	Dual_Same (Leapfrog)	Pull
Case2	Dual_Same (Leapfrog)	Push
Case3	Dual_Same (Leapfrog)	Prioritize_loadlock
Case4	Dual_Same (Leapfrog)	Prioritize_Chamber
Case5	Dual_opposite (Bisymmetric)	Pull
Case6	Dual_opposite (Bisymmetric)	Push
Case7	Dual_opposite (Bisymmetric)	Prioritize_loadlock
Case8	Dual_opposite (Bisymmetric)	Prioritize_Chamber

7 RESULTS

The simulation scenarios were set up and the case runs were made. Throughput results for each case run was reported. Both Product Throughput, and Total System Throughput was reported. Throughput is given in wafers per hour (WPH). Each case ran a total of 20 FOUPs, each with 25 wafers. Table 5 lists the results for scenario 1. Table 6 lists the results for scenario 2.

Table 5: Scenario 1 Throughput Results

Scenario 1:	Product1	Product2	Total
(Dedicated Load Locks)	Throughput	Throughput	Throughput
Case Run	WPH	WPH	WPH
Case1	51.0	13.8	64.8
Case2	55.8	12.4	68.2
Case3	49.3	14.2	63.6
Case4	57.7	14.5	72.2
Case5	51.8	13.8	65.6
Case6	54.2	12.5	66.8
Case7	46.4	14.5	60.9
Case8	57.6	14.5	72.0

Table 6: Scenario 2 Throughput Results

Scenario 2:	Product1	Product2	Total
(Either Load Lock)	Throughput	Throughput	Throughput
Case Run	WPH	WPH	WPH
Case11	60.4	12.1	72.3
Case12	53.3	12.3	65.6
Case13	65.7	11.6	77.4
Case14	81.2	13.5	94.7
Case15	58.3	12.7	70.9
Case16	53.4	12.4	65.8
Case17	65.4	11.7	77.1
Case18	80.5	13.6	94

7.1 Configuration Comparison

Figure 4 illustrates the side-by-side results when comparing the dedicated load lock cases (scenario 1) against the non-dedicated load lock cases (scenario 2).

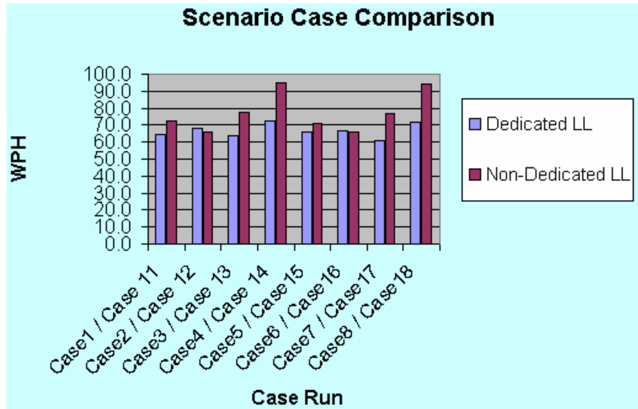

Figure 4: Scenario Case Comparisons

This graph illustrates the throughput advantage of not dedicating a load lock to a specific product. Throughput capability is on average 15.6 percent higher when either product can pass through either load lock

7.2 Robot Type Comparison

Figure 5 illustrates the side-by-side results when comparing the robot types used.

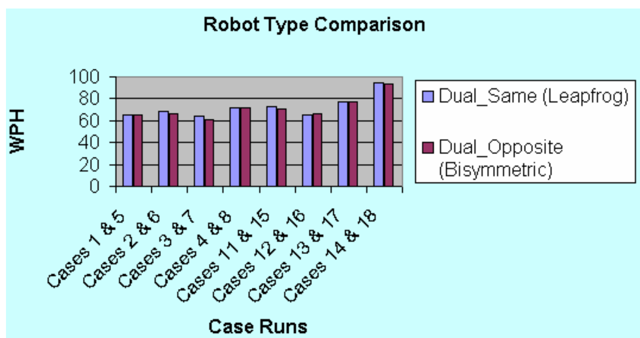

Figure 5: Robot Type Comparisons

This graph illustrates that both types of dual ended robots achieve similar results. There is little difference (less than 1percent on average) in throughput when comparing the two robot types (the leapfrog robot has the small advantage).

7.3 Algorithm Comparison

Figure 6 illustrates the side-by-side results when comparing the four robot algorithms (wafer selection rules).

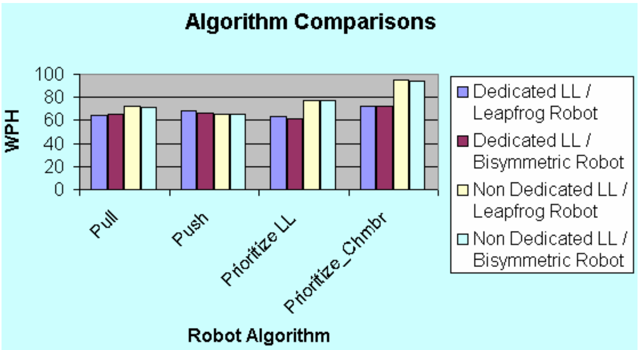

Figure 6: Algorithm Comparisons

This graph illustrates the effect the algorithm has on throughput. It is clear that the algorithm that prioritizes the moves to and from the processing chambers, provides a higher throughput. The algorithm improvements range from 8.9% to 30.7% increase depending on the configuration as shown in Table 7.

Table 7: Robot Algorithm Improvements

Configuration	Worst Algorithm	Best Algorithm	% Improvement
Dedicated LL / Leapfrog Robot	Pull	Prioritize_Chamber	10.2%
Dedicated LL / Bisymmetric Robot	Prioritize LL	Prioritize_Chamber	8.9%
Non Dedicated LL / Leapfrog Robot	Push	Prioritize_Chamber	30.7%
Non Dedicated LL / Bisymmetric Robot	Push	Prioritize_Chamber	30.0%

8 CONCLUSIONS

The increasing complexity and versatility of semiconductor equipment has enhanced the need for simulation. The initial investment of time and resources for constructing a flexible simulator of the equipment design concept is minimal when weighed against the benefits it provides.

In the Infineon study, the simulator accurately identified the scheduling rules and options that would have a positive effect on tool performance. Using simulation at Infineon has provided a benchmark for measuring the performance of the cluster tool control system and wafer move sequence. It helped answer many difficult and complex questions, enabling control software programmers to focus on impact areas.

APPENDIX A: ALGORITHM DESCRIPTION

Four Algorithms were defined and used in the simulation study. The following is a brief description of each:

Pull
The Pull algorithm is a common algorithm used by single or dual ended robots. As the name implies, the algorithm "pulls" wafers through the system. This is done by allowing the robot to pick a wafer only if the destination chamber of the wafer is available (not processing or cleaning). Otherwise, the robot will not pick the wafer. If more than one wafer qualifies to be picked, then the wafer that has been waiting the longest will be selected.

Push
The "Push" algorithm is a common algorithm for dual ended robots (typically running serial flows). As the name implies, the robot will pick a wafer with the free end, and push it to the destination chamber. If the destination chamber is currently processing another wafer, then it will pre-position in front of the chamber and wait to "swap" the processed wafer with the unprocessed wafer. This algorithm works well with serial flows as it tends to isolate the dynamic bottleneck of the system. However, it is complicated when multiple product flows and clean cycles are introduced.

Prioritize Load Lock
The "Prioritize Load Lock" algorithm was defined to prioritize wafer loading and removal at the load locks. If the robot has both blades open, and a wafer inside the load lock is available to be picked, then it will pick this wafer regardless of chamber availability. The robot will also prioritize placing processed wafers inside the load lock. This algorithm works well when the load locks are the system bottleneck.

Prioritize Chamber
The "Prioritize Chamber" algorithm was defined to prioritize wafer loading and unloading at the processing chambers. If the robot has both blades open, and a wafer inside a processing chamber is available to be picked, then it will pick this wafer regardless of load lock availability. It will use one end of the robot as a temporary storage position for the wafer. This allows the chamber to empty and start the clean cycle sooner, making it available for the next wafer sooner. This algorithm works well when the system is limited by the processing chambers (processing time and clean time) as is evident in the results.

APPENDIX B: ROBOT TYPE DESCRIPTION

Two robot types were defined and used in the simulation study. The following is a brief description of each:

Dual Same (Leapfrog)
The "Dual Same" robot has two end-effectors (blades) for moving wafers. Each blade can extend and retract independently. The blades are on the same side of the robot, one directly over the other and coupled around the same rotational axis. The robot does not need to rotate 180 degrees when doing a swap, as does the dual_opposite (bi-

symmetric) robot. A common "Dual Same" robot type is shown below.

Dual_opposite (bisymmetric)

The "Dual_opposite" robot also has two end-effectors (blades) for moving wafers. The blades are on opposite sides of the robot, coupled around the same rotational axis. When making a wafer move, the robot can select the blade that is closest to the destination chamber. The robot needs to rotate 180 degrees when doing a wafer swap. A common "Dual Opposite" robot type is shown below.

REFERENCES

Aybar, M. and Potti, K, 2002. Using Simulation to Understand capacity constraints and improve efficiency on process tools. In *Proceedings of the 2002 Winter Simulation Conference.* Available via <www.wintersim.org>.

Hendrickson, R. 2000. Using Emulation to Validate a Cluster Tool Simulation Model for the *Proceedings of the 2000 Winter Simulation Conference*.

Pare,K, 2002. Using Simulation to Understand and Improve Process Tool Efficiency for the *Proceedings of the MASM 200 conference.*

AUTHOR BIOGRAPHIES

TODD LEBARON has worked for Brooks Automation since 1990 as a simulation analyst. He has conducted numerous simulation studies over the past fifteen years in a variety of applications. He has lead the development of the ToolSim™ Simulation Software. He also teaches AutoMod training courses and provides consulting support. Mr. LeBaron received a B.S. in Manufacturing Engineering from Brigham Young University in 1988. His email address is todd.lebaron@brooks.com.

JOERG DOMASCHE is a Senior Staff Engineer for Modeling and Simulation and Equipment Productivity within Infineon Technologies since 1996. He received his Masters of Science from the University of Wuerzburg in 1996 with major studies in queuing theory and simulation. His email address is joerg.Domaschke@infineon.com.