FabTime Cycle Time Management Newsletter

Volume 6, No. 4 May 2005

Information

Mission: To discuss issues relating to proactive wafer fab cycle time management

Publisher: FabTime Inc. FabTime sells cycle time management software for wafer fab managers. New features in this version (7.0) include a tool dispatch lot list chart that displays (for each tool) the list of lots queued for the tool, sorted in suggested dispatch order according to the selected dispatch logic.

Editor: Jennifer Robinson

Table of Contents

Welcome

Welcome to Volume 6, Number 4 of the FabTime Cycle Time Management Newsletter! We are lacking in community announcements for this issue, but we do have subscriber discussion related to last month's article on overall WIP effectiveness, and a new question about troubleshooting declining moves in a wafer fab. Our FabTime software user tip of the month is about excluding extended holds for marketing purposes from the shipped lot cycle times reported by manufacturing.

In our main article this month, we discuss WIP management, with emphasis on lot dispatching. We begin by defining scheduling and dispatching. We then define several standard dispatch rules, and examine the common factors underlying these rules. Finally, we briefly review some of the fab-specific issues that lend particular complexity to dispatching: batching, setups, and time constraints between process steps. This is far from being a comprehensive description of all of the possible fab dispatch rules. Instead, our intention is to introduce a common vocabulary for dispatching, so that in future discussions we can explore more complex scenarios. We welcome your feedback.

Thanks for reading!—Jennifer

FabTime

Tel: (408) 549-9932
Fax: (408) 549-9941
www.FabTime.com
Sales@FabTime.com

Community News/Announcements

FabTime welcomes the opportunity to publish community announcements. Send them to newsletter@FabTime.com. There are no new community announcements this month.

FabTime User Tip of the Month

Exclude Extended Hold Time from Cycle Time Reports

Generally, you report cycle time as the time from when a lot enters the fab until the lot is shipped by manufacturing, including all queue time, process time, travel time, hold time, etc. However, sometimes lots undergo extended holds for marketing or staging purposes. We've seen these hold areas called "crib", "z" (for sleeping), and "staging", among other names. These types of holds are usually outside of the control of the manufacturing organization. Therefore, what you would sometimes like to do is adjust your reported cycle times by deducting this extended hold time.

In FabTime, you can subtract out this extended hold time on any chart that displays factory cycle time, including the Shipments Lot List chart, the Factory Cycle Time Trend and Pareto charts, and the WIP Lot List chart. (For the WIP Lot List, this only applies when the "Age" control is set to "Factory" rather than the default "Current Opn".) Near the bottom of the main set of filters to the left of the chart are two filters labeled "-Own:" and "-Opn:". These subtract out, respectively, any time that each lot spends with a particular owner code or at a particular operation. The standard filter syntax rules apply (comma separated lists, wildcards, and not filters). Therefore, as long as you can identify the time that the lot (or lots) spent in extended hold through either a special owner code (e.g. the lot changed owner to become "MFGZ"), or a special set of operations (e.g. "Crib" or "Store" operation), you can easily subtract out the extended hold time from your reported shipped lot cycle times. Note that the chart title will, as always, display the list of filters applied to the chart, so that it remains clear exactly which cycle time data you are reporting.

If you have any questions about this feature (or any other software-related issues), just use the Feedback form in the software.

Subscriber Discussion Forum

Issue 6.03: Overall WIP Effectiveness

An anonymous reader submitted the following comments in response to last month's article on Overall WIP Effectiveness. "An item of caution... The theoretical cycle time in proportion to OWE must be factored, as it will be influenced by accommodation to batching rules and lot size on single wafer tools, and may introduce a disturbance for an individual lot.

For example, we watch "super hot" lots very closely for turn rate or days/layer. An aggregate index for this sub-class of lot is widely observed, but the mathematics and reports which generate it can be run for "normal lots" too. A super hot lot which is only 6 wafers long can surpass 1X theoretic cycle time if the theoretic basis is established as a fixed table for a full 25 wafer lot, while a normal lot will typically have much more queue and transport time, etc.

A dynamic theoretical cycle time would make OWE comparable to any other watchdog on cycle time such as turn ratio or days/layer, if the goal is set properly. Most fabs also already have a watchdog on cycle time and hold time. The more interesting aspect of your proposed index is the easy observation and classification of the not-OWE time into actionable events.

Be cautious that the proposed new index does not just come full circle to an item we already watch under a different name."

FabTime Response: We think that this reader makes a good point. Of course in practice fabs run into these issues with theoretical cycle time quite often (e.g. when calculating traditional OEE, etc.). But we agree that the most useful part of what we're doing lies in the categorization into WIP States.

Identifying the Cause of Declining Moves

An anonymous subscriber wrote: "We have been experiencing a declining trend in our wafer moves and are trying to understand the root cause. Can you provide me with a brief set of charts that you could think of that would help diagnose why the quantity of moves is declining? We know it's not tool down related since we haven't have many tool down issues recently."

FabTime Response: We would recommend starting by looking at the trend in your starts. Could your moves have gone down simply because the starts have gone down? If the starts haven't gone down, then we would look at WIP, sliced by area or toolgroup. The question is: where is your WIP piling up?

You might also look at a moves pareto, to see if you have some area that isn't pulling its weight, so to speak. Or, generate moves trend charts for each production area, displayed in 12-hour periods, and compare performance to goals for each area. Alternative, a WIP turns chart would show, for each period, moves during the period, divided by starting WIP at the beginning of the period. This tells you, for areas that don't have much WIP, whether they're working proportionally or not. In general, comparing moves to WIP can be very informative. If you have a moves problem in an area with plenty of WIP, then there must be some other behavior going on.

Once you narrow in on the area or toolgroup, we would still recommend looking at the tool state data. Even if you haven't had significant down events, perhaps you're having extra engineering time, or perhaps you have idle time even when there's WIP waiting (suggesting operator issues). At this point, it's probably time to talk with someone in the area to see what's happening (e.g. look at WIP by operation to see where it's piling up, see if there is a reason for this, see how it can be resolved). All of the charts that we have mentioned here are standard in our FabTime cycle time management software, and most exist in other in-house reporting systems.

Lot Dispatch for Wafer Fabs

As many of you know, FabTime has been conducting an informal survey of the causes of cycle time problems in wafer fabs. To date, we have tabulated nearly 400 responses. Over the two years since we started this survey, the three most frequent responses have always been: "downtime", "tool utilization", and "one-of-a-kind (single path) tools". However, a fourth response has been climbing in the ratings over the past few months, and now stands a close fourth on the list. This response is "WIP management (dispatching and scheduling)".

This response is not so surprising. Fabs are becoming ever more complex in terms of product mix and short product life cycles. Due dates to customers are becoming more and more aggressive. Reticles are ever more expensive, while increased product complexity means that there are more different reticles required all the time. And so on. All of this means that simply managing the WIP in the fab – deciding which lot should be run next on which tool – is a bit more difficult every day. In this article, we offer a brief introduction to the topics of scheduling and dispatching, though our primary focus is on dispatching. We also discuss some of the particular requirements of wafer fabs that make dispatching more of a challenge.

Scheduling vs. Dispatching

The terms scheduling and dispatching are often used interchangeably in conversation. However, they refer to two different things. Scheduling requires looking at a set of lots, and planning ahead of time (perhaps at the start of each shift) exactly which lots will be processed on which tools, and in which order. This requires looking at all of the lots at one time, and making decisions that are best for the fab as a whole, even if they appear sub-optimal from the perspective of individual tools or areas.

Potential schedules are evaluated in terms of an objective function, which may include various weighted goals (due date performance, minimum queue delays for all lots, minimum queue delays for hot lots, etc.). The optimal schedule is the one that has the best value for the objective function. This can be difficult to determine, however, because fabs are highly complex, and have many possible schedules. A schedule is considered "good" (though not optimal) if it achieves a reasonable value for the objective function, and can be generated quickly. Because there is so much variability in a fab, schedules often need to be re-generated or adjusted during the shift (for example, if a key tool goes down unexpectedly).

Dispatching, on the other hand, involves looking at the set of lots that are in queue and qualified to run on a particular tool, and applying a rule to determine which order to use for processing the lots. Dispatching logic is often local (rather than looking at the entire fab), although some dispatch rules take into account nearby operations and expected arrivals to a tool. Dispatch scores for each lot are calculated on the fly, whenever a tool becomes available for processing. The lot with the highest score is processed next.

Dispatch Rules vs. Dispatch Factors

A full review of dispatch rules (and associated articles) would run to hundreds of pages. We won't attempt such a feat here. Instead, we'll cover a few of the standard rules, and along the way, point out common dispatch factors – the component pieces of the rules. Individual rules are nice subjects of study, but in practice it is often desirable to blend the underlying factors in a new way to match the situation at hand.

Several Standard Dispatch Rules

In this section, we discuss some standard dispatch rules. Many other dispatch rules have been proposed for wafer fabs – several are included in the "Further Reading" section of this article. However, the rules described here are a good starting point.

PriorityFIFO (Priority, first-in-first-out)

This is a common dispatch rule seen in fabs, particularly in fabs that produce to stock (rather than to individual customer orders). Lots are ordered first by priority, so that higher priority lots are processed first, and then by elapsed time at the current operation (so that the lots that were waiting for the tool first are processed before later-arriving lots). Fabs that manage by looking at "inactive" lots (lots that have been at their current operation for more than some period of time), are generally driving towards FIFO.

PriorityCR (Priority, critical ratio)

This is a dispatch rule that is frequently used in fabs where due date performance of individual lots is important. As above, lots are first grouped by priority. Then, within each priority they are ordered in decreasing order of their critical ratio – a ratio of work remaining (for the lot before it leaves the fab) to time remaining before the lot's due date. The exact calculation for critical ratio varies, but the basic idea remains the same. Lots with much work remaining and only a short time to get it done have a high critical ratio, and thus are favored by this rule.

PriorityEDD (Priority, earliest due date)

Simpler than the critical ratio rule, this dispatch rule groups lots by priority, and then orders them in increasing order of due date (so that the lots with the earliest due date are always processed before lots with later due dates). However, this rule does not take into account where a lot is in the line relative to its due date. Therefore, it can result in some lots finishing well ahead of schedule, at the expense of other lots which have been started more recently.

PrioritySPT (Priority, shortest processing time)

Under this rule, lots are grouped by priority, and then in increasing order of the expected process time of the current operation on the tool. This rule has been shown, at a local level, to be good at reducing average cycle time per visit across all the lots through a tool. However, it can result in long waits for lots at long process time operations, and can be detrimental to due date performance.

PrioritySRPT (Priority, shortest remaining processing time)

Under this rule, lots are grouped by priority, and then in increasing order of total remaining process time (across the entire flow). This rule tends to pull forward lots that are near completion, though at the expense of lots earlier in the process flow. You sometimes see this rule applied when there is a need to increase outs over a short time period (e.g. the end of the quarter is approaching).

For all of the dispatch rules discussed above, non-priority versions exist (that is, the rule is applied independently of the individual lots). However, these are probably not as relevant for fabs as the priority versions of the rules. We've never yet visited a fab that didn't have multiple lot priorities.

Dispatch Factors

From the list above, we see several factors that recur in various forms:

1. Lot priority
2. Lot queue time (at the current step)
3. Lot due date
4. Lot processing time (at the current step)
5. Lot remaining process time (to shipment)

Even with just these factors, it is possible to compose a very large number of dispatch rules – all 5-way combinations of ordering the rules (leaving aside shorter combinations and weighted combinations) would be: 5 * 4 * 3 * 2 * 1 = 120 potential rules. It's easy to come up with at least five other plausible factors (see our next section), and all 10-way combinations would be 10 * 9 … * 1 = 3,628,800 potential rules.

Our point is that it's difficult to imagine a single dispatch rule from this large universe being provably best for any one fab, let alone for all fabs. Rather, we suggest you focus on the underlying dispatch factors, and choose factors that match your particular operating goals.

A Few Additional Dispatch Factors

Other dispatching factors that may prove useful include:

1. Downstream tool priority – is it a constraint?

2. Downstream batch efficiency – should we favor full batches?

3. Time constraints – how can we keep lots from timing out?

4. Production planning move targets – do we need to focus on a particular line segment?

5. WIP targets – can we send WIP to where it will be processed, rather than wait in queue?

6. Setup time – is it worthwhile to delay a lot in order to minimize setups?

7. Tool qualification – how many tools can run this lot?

Again, this is not an exhaustive list – and it will only grow larger with time. In our FabTime software, we are migrating from dispatch rules to dispatch factors to avoid the problem of rule multiplication as more factors are defined. The site decides, at the tool level, what rankings and weights to use for the various factors. We believe that this approach holds great promise in balancing the simplicity of dispatch rules against the need to develop more global solutions.

Fab Complexities Related to Dispatching

Entire dissertations have been written about various complexities related to dispatching for fabs, particularly batch processing, setups, and time constraints between process steps. We will touch on the issues related to each of these here, and provide you with some references at the end of the article for further information.

Batching

There are two basic approaches to dispatching decisions for batch tools. One approach is to look at all possible combinations of each allowable batch size, examining each one individually in order to find the "best" batch right now. This requires total enumeration of all feasible batches, which very quickly becomes difficult due to the combinatorial explosion in the number of possibilities.

The other approach, and the one that we use in FabTime, is to score the lots individually, and pick the lot with the highest score to be processed next. We then look down the dispatch list to find other lots (in decreasing order of dispatch score) that share the same batch ID, in order to fill the batch as much as possible. Thus a batch tool's dispatch list can look like the following (assume the earlier in the alphabet, the higher the lot's dispatch score and that the current tool has a maximum allowable batch size of six lots):

Batch 1) Lot A, Lot F

Batch 2) Lot B, Lot D, Lot E, Lot G, Lot H, Lot I

Batch 3) Lot C

If the tool's maximum batch size were four lots, the dispatch list would look like this (given our assumptions about lot priority):

Batch 1) Lot A, Lot F

Batch 2) Lot B, Lot D, Lot E, Lot G

Batch 3) Lot C

Batch 4) Lot H, Lot I

In this way, although the individual lot ranking was A, B, C, and so on, A can only be batched with F, so they together represent the "recommended" or top scoring batch to load. Minimum batch size requirements can be included at this point, such that if the minimum batch size required by the tool was four lots, Batch 2 would be processed first. (See Volume 2, Number 1 of the newsletter for a discussion of minimum batch size policies and their impact on cycle time.)

Dispatch scoring may contain particular logic with regards to batching. For example, in FabTime we have included the following:

- Batch efficiency (actual load size divided by maximum batch size) can be weighted heavily to give preference to lots going into full batches at the downstream step.

- Upstream batching is controlled so that lots exiting the upstream step in a batch do not violate a downstream timer if sent to a non-batch tool.

- Upstream batching is controlled by the downstream batch size if they are both batch tools and there is a batch size mismatch.

Setups

Tools with setups between recipes (e.g. implanters) may require modification to the dispatch rules described above. A common rule used on tools with significant setups between recipe types is a "setup avoidance" policy. Under a setup avoidance policy, the dispatch list is ordered according to the dispatch rule. However, the operator then looks down the dispatch list, and selects the highest ranking lot that does not require a setup. Only if there are no matching lots in the queue is a setup performed. In that case, the lot at the top of the dispatch list is generally selected.

Setup avoidance policies can be quite successful in reducing the amount of time that a tool spends in setup. However, they can lead to extremely long wait times for lots with infrequently occurring recipes. In many cases, a cap is included on the wait time of lots in the queue, such that the operator is forced to do a setup if any lot waits for more than some time interval (e.g. 12 hours). Determining what this time interval should be is an excellent application of simulation analysis.

Time Constraints between Process Steps

A particular dispatching problem for many fabs is the presence of time constraints between process steps. For example, a lot may undergo a clean operation, and then be required to complete the next operation, a bake step, within some time window. If the time window elapses without the bake step being completed, the lot must return to re-do the clean operation. This results in a capacity loss at the clean operation, because additional process time is incurred, without processing any new lots. Therefore, logic that looks downstream at the bake operation prior to completing the clean step may be needed.

Conclusions

WIP management is a critical task faced by people running fabs today. In this article, we have defined dispatching and scheduling, two primary approaches for WIP management. We have concentrated primarily on dispatching, which requires ranking all of the lots in queue for a particular tool according to some dispatch score, and processing them accordingly. We have described several of the standard local dispatch rules in use today, and proposed a focus on the common factors underlying dispatch rules, rather than on the individual rules themselves. Finally, we have briefly outlined some of the particular dispatching-related issues that face fab personnel, including batch tools, setups, and time constraints between process steps. This is by no means a comprehensive article. Much research has been undertaken in the area of dispatching and scheduling. Instead, our purpose has been to lay out some common vocabulary in this area, to facilitate further, more detailed discussions. We hope that you will care to participate.

Closing Questions for FabTime Subscribers

What types of dispatching do you use in your fab? What are the characteristics that make dispatching particularly complex for your site?

Acknowledgment

FabTime would like to thank Professor Scott Mason of the University of Arkansas for his work with us on dispatch rules, and in particular for his input to the section on batching.

Further Reading

■ S. Barman, "Simple Priority Rule Combinations: An Approach To Improve Both Flow Time And Tardiness," *International Journal of Production Research*, Vol. 35, No. 10, 2857-2870, 1997.

■ J. W. Fowler, S. Brown, H. Gold, and A. Schoemig, "Measurable Improvements in Cycle-Time-Constrained Capacity," *Proceedings of the 6th IEEE/UCS/SEMI International Symposium on Semiconductor Manufacturing (ISSM)*, October 6-8, 1997, San Francisco, A21-A24.

■ C.-E. Lee and C.-W. Chen, "A Dispatching Scheme Involving Move Control and Weighted Due Date for Wafer Foundries," *IEEE Transactions on Components, Packaging, and Manufacturing Technology. Part C: Manufacturing*, Vol. 20, No. 4, 268-277, 1997.

■ Y. H. Lee and M. Pinedo, "Scheduling Jobs On Parallel Machines With Sequence-Dependent Setup Times," *European Journal of Operational Research*, Vol. 100, No. 3, 464-474, 1997.

■ C. Rippenhagen and S. Krishnaswamy, "Implementing the Theory of Constraints Philosophy in Highly Reentrant Systems," *Proceedings of the 1998 Winter Simulation Conference*, Washington, DC, D. J. Medeiros, E. F. Watson, J. S. Carson, and M. S. Manivannan, eds, 993-996. (All 1997 to 2003 WSC papers are available for free download from http://www.informs-cs.org/wscpapers.html).

■ J. Robinson and F. Chance, "A Simple Rule of Thumb for Batching Decisions," *FabTime Newsletter*, Volume 3, Number 8, 2002.

■ J. Robinson and F. Chance, "Impact of Batch Size Decision Rules on Cycle Time," *FabTime Newsletter*, Volume 2, Number 1, 2001.

■ O. Rose, "Some Issues of the Critical Dispatch Rule in Semiconductor Manufacturing," *Proceedings of the 2002 Winter Simulation Conference*, 2002. (All 1997 to 2003 WSC papers are available for free download from http://www.informs-cs.org/wscpapers.html).

■ O. Rose, "Accelerating Products under Due-Date Oriented Dispatching Rules in Semiconductor Manufacturing," *Proceedings of the 2003 Winter Simulation Conference*, S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds., 2003. (All 1997 to 2003 WSC papers are available for free download from http://www.informs-cs.org/wscpapers.html).

■ R. Sturm, J. Dorner, K. Reddig, J. Seidelmann, "Simulation-based Evaluation of the Ramp-up Behavior of Waferfabs," *Proceedings of the 14th Annual IEEE/SEMI Advanced Semiconductor Manufacturing Conference and Workshop*, Munich, Germany, 2003.

■ J. C. Tyan, J. C. Chen, and F. K. Wang, "Development of a State-Dependent Dispatch Rule using Theory of Constraints in Near-Real-World Wafer Fabrication," *Production Planning & Control*, Vol. 13, No. 3, 253-261, 2002.

Subscriber List

Total number of subscribers: 1859, from 420 companies and universities. 25 consultants.

Top 10 subscribing companies:

- Intel Corporation (92)
- Analog Devices (78)
- ATMEL Corporation (62) – New this month to the top 10 list!
- Infineon Technologies (57)
- STMicroelectronics (55)
- Freescale Semiconductor (50)
- Philips (48)
- Micron Technology (47)
- Texas Instruments (42)
- AMD/Spansion (37)

Top 4 subscribing universities:

- Virginia Tech (10)
- Arizona State University (8)
- Nanyang Technological University (6)
- University of California – Berkeley (6)

New companies and universities this month:

- Epson
- Semitool
- Tata BP Solar

Note: Inclusion in the subscriber profile for this newsletter indicates an interest, on the part of individual subscribers, in cycle time management. It does not imply any endorsement of FabTime or its products by any individual or his or her company.

There is no charge to subscribe and receive the current issue of the newsletter each month. Past issues of the newsletter are currently only available to customers of FabTime's web-based digital dashboard software or cycle time management course.

To subscribe to the newsletter, send email to newsletter@FabTime.com, or use the form at www.FabTime.com/newsletter.htm. To unsubscribe, send email to newsletter@FabTime.com with "Unsubscribe" in the subject. FabTime will not, under any circumstances, give your email address or other contact information to anyone outside of FabTime without your permission.

FabTime® Cycle Time Management Software

"Instead of spending time preparing reports, shift facilitators can get the data they need quickly from FabTime, and then spend their time making real improvements."
Mike Hillis
Cycle Time and Line Yield Improvement Manager
AMD Fab 25

FabTime Installation

One fixed price includes

- Site license, unlimited users.
- Implementation & training.
- Software maintenance.

Pilot Project – Analyze your data with FabTime

For $4950, FabTime will

- Identify key contributors.
- Benchmark common metrics.
- Review results at your site.

Interested?

Contact FabTime for technical details or a pilot project quote.

FabTime Inc.
Phone: +1 (408) 549-9932
Fax: +1 (408) 549-9941
Email: Sales@FabTime.com
Web: www.FabTime.com

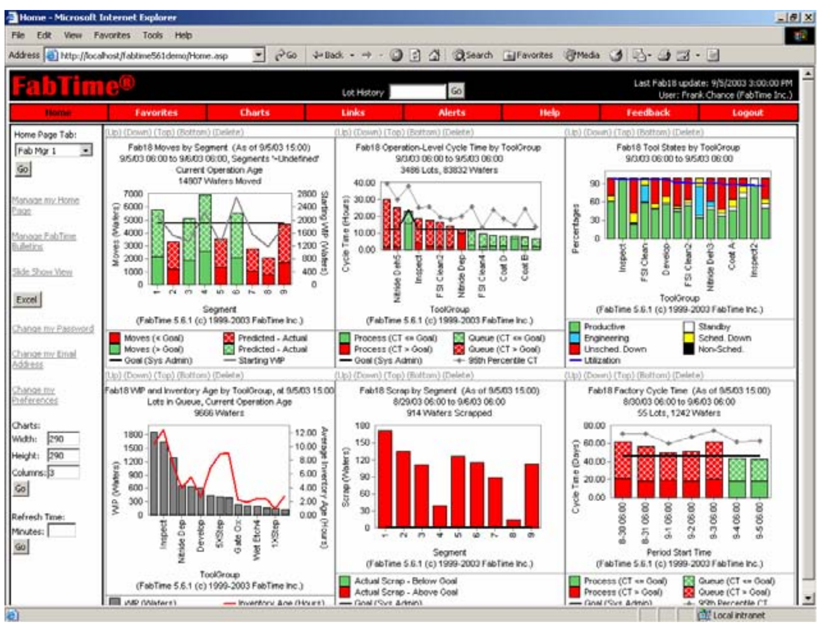
Do you have the best possible information?

- Are your supervisors swamped with daily reports, but lacking real-time information?
- Is it difficult to link equipment performance to cycle time?
- Does each new cycle time analysis require IT resources?

FabTime is a digital dashboard for your fab. In real-time, it provides a comprehensive view of fab performance data – everything you need for proactive management of cycle time. FabTime is designed for hands-on use by managers and supervisors, unlike traditional reporting tools, which were designed for programmers. FabTime also now includes **lot dispatching** (via dispatch rules) and static **capacity planning**.

A Web-Based Digital Dashboard

"I use FabTime every day, and so do the supervisors who report to me. The data that I need is right on my home page where I need it when I come in every morning."
Jim Wright
Production Manager
Headway Technologies



FabTime Benefits

- Cut production cycle times by 10%, hot lot cycle times by 20%.
- Focus improvement efforts on the tools that inflate cycle time.
- Improve supervisor productivity – cut reporting time by 50%.