FabTime Cycle Time Management Newsletter

Volume 6, No. 7 August/September 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 (7.5) include an alert based on the number of new arrivals to a production area.

Editor: Jennifer Robinson

Contributors: Professors Ad de Ron and Koos Rooda (Technical University of Eindhoven)

Table of Contents

Welcome

Welcome to Volume 6, Number 7 of the FabTime Cycle Time Management Newsletter! We hope that the end of the summer finds you all well and ready to think about improving cycle time. This month we have a conference announcement about ISSM, to be held in San Jose next week. Our software user tip of the month is a primer on using FabTime's Excel export functionality. We also have subscriber discussion related to identifying the cause of declining moves in a wafer fab and improving lot tracking in less automated fabs (both from Issue 6.06), and the fundamental drivers of fab cycle time (from Issue 6.05). A new subscriber discussion topic about formalizing methods for setting operation cycle time goals is also included.

In our main article this month we discuss setup avoidance policies for lot dispatching. Where present, setups reduce the available capacity of tools, and tend to increase cycle time. This has led to frequent use of setup avoidance policies in fabs. However, running a pure setup avoidance policy can lead to long cycle times for low volume recipes. Various methods exist for forcing a setup to occur to prevent long queue times – several are discussed below. We also discuss integration of setup avoidance into a dispatch factor paradigm, and parallels between setup avoidance and batch size formation decisions. 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

International Symposium on Semiconductor Manufacturing (ISSM) – San Jose, CA – September 13-15

ISSM will be held in San Jose next week. The conference website (www.issm.com) says that "ISSM is the industry's largest forum of semiconductor manufacturing professionals dedicated to sharing technical solutions and opinions on the advancement of manufacturing science. This symposium has been held in Japan and in the U.S. on alternate years since 1992. ISSM aims to establish new concepts for semiconductor manufacturing technologies and to promote them as systemized and universalized technologies."

If you are planning to be in San Jose for ISSM, and would like to arrange for a demo of FabTime's web-based digital dashboard software, send an email to Jennifer.Robinson@FabTime.com.

FabTime welcomes the opportunity to publish community announcements. Send them to newsletter@FabTime.com.

FabTime User Tip of the Month

Export Chart Data to Excel

FabTime's data tables can each be sorted by up to three variables, using the sort controls located above the data table. Sometimes, however, you want to do more in-depth analysis of the raw data. To export the chart data to Excel, simply click on the "Excel" button located above the data table. In most cases, Internet Explorer will pop up with a message box stating: "An ActiveX control on this page might be unsafe to interact with other parts of the page. Do you want to allow the interaction?" Click "Yes". FabTime will then open up Excel on your system, and pull in the data table. Hyperlinks will be removed, so that the result is sortable, editable data. A few specific pointers:

- The Excel export can only export the data that is currently displayed on your chart page. This is limited to some maximum number of rows by the "Rows" text box (located right above the Excel button). If you want to export more rows than you usually display on your chart pages, you'll need to first increase the number of rows displayed (by entering a larger value in the "Rows" text box, and pressing the associated "Go" button), and then press the Excel button.
- Because this is an ActiveX control, the Excel export only works if you use Internet Explorer as your FabTime browser. And of course it can only open up the Excel spreadsheet if you are

working on a computer that has Excel installed.

■ The Excel files that are generated by FabTime are stored in a temporary directory. If you wish to make changes to a file, the safest thing is to do a "SaveAs" from Excel's File menu, and save/rename the file to some location on your own computer.

■ If you press the "Excel" button on your home page, FabTime will export pictures of the charts that are displayed there. Although these are not editable Excel charts (they are pictures), this can be useful if you need to send someone a snapshot of all of the charts on your home page. Any chart notes that you have entered on the "Manage Tabs/Notes" interface will be displayed below each chart in a text box. This may affect the formatting of the workbook, if you have long chart notes. However, once you have the charts as graphics in Excel, you can format as you like.

■ Depending on the configuration of your system, you may get a message the first time you click the Excel button that says "I'm sorry, but I was unable to start Excel. Click OK and I'll display a help page that discusses the most likely solution to the problem." This help page walks you through one Internet Explorer setting that you may need to change in order to enable Excel export on your system. If you have trouble with this, just use the feedback form in FabTime to ask for help.

As always, 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 6.06: Identifying the Cause of Declining Moves in a Wafer Fab and Improving Lot Tracking in Less Automated Fabs

An anonymous subscriber sent in responses to two open topics mentioned in last month's issue:

"Identifying the cause of declining moves in a wafer fab:

I suggest revising the topics aggressively to "Improve the fab baseline to increase movement". Specific ideas to do this are:

a.) Identify the bottleneck/key tools, analyze these tools' availability categories, and then take action for these categories (from either an engineering or an automation point of view).

b.) Perform operator motion & time studies, to break down the contents of what the operator did during fab working hours. Take action to let the operator focus on production activity only.

At our site, by accomplishing a), we continuously reached record high movements twice within one month. By accomplishing b) also, I think that another record high will not be far away.

Improving lot tracking in less automated fabs:

a.) Requires an integrated CIM solution, including not only product data but also tool data and engineering data collection.

b.) Requires as much detail as possible regarding process step definition in the MES.

c.) Requires more frequent report generation than just daily. (The data should include month-to-date, week-to-date, and day-to-adhoc hours)

d.) Requires customization to search by lot ID instead of having to search the database as a whole (after shipping out tracking, i.e. RMA or other customer concern follow-up)"

Issue 6.05: The Three Fundamental Drivers of Fab Cycle Time

Professors Ad de Ron and Koos Rooda of the Technical University of Eindhoven wrote: "With interest we have read your article entitled "The Three Fundamental Drivers of Fab Cycle" in the FabTime Newsletter. We think it is a good initiative to show the drivers of fab performance. We have written a note about fab performance drivers and our result is that the stochastic variables interarrival time and effective process time (with their variabilities) are the basic drivers for fab performance. The difference with your approach is that, in our view, the interarrival time is more basic than utilization, which can be derived from it. The same can be remarked about the effective process time instead of number of qualified tools."

Professors de Ron and Rooda sent us their technical note on the subject, as well as a related paper which has been accepted for publication in IEEE Transactions on Semiconductor Manufacturing. The abstract for the IEEE TSM paper (titled "Fab Performance") follows:

Abstract—To remain competitive, and to boost profitability, manufacturers in capital-intensive and highly competitive industries want to maximize throughput and minimize flow time. Achieving high throughput conflicts with achieving low flow time. In order to unhide the trade-off between throughput and flow time a performance measure, called manufacturing performance, has been developed. The manufacturing performance is defined by the quotient of the ratio between throughput and flow time of an actual manufacturing system and this ratio of a reference system. The reference system can be adapted by the user in correspondence with objectives.

By applying the manufacturing performance to one workstation and using analytic approximations for this workstation, manufacturing performance can be expressed analytically. It seems that manufacturing performance has an optimal value that is given by equipment availability and coefficient of variation. Manufacturing performance is applied also to a 4-workstation manufacturing line. Results from analytic approximations show the practicability of the manufacturing performance. Comparison of manufacturing performance with OFE, an earlier proposed metric, showed that the manufacturing performance is a more clear metric. This conclusion was based upon simulations with a two-stations manufacturing line.

The manufacturing performance is a technical performance metric for manufacturing lines that supports, for instance, economical considerations to obtain optimal throughput flow time combinations under economical optimal results. This is a useful addition to the existing metrics, which may benefit manufacturers in their operations. The authors consider this contribution as a discussion paper and demand for comment.

FabTime Response: We believe that Professors de Ron and Rooda are working in the same general direction that we are - trying to understand the primary drivers of fab performance. We agree with their focus on variability, and consider their focus on interarrival time and effective process time (rather than our stated focus on number of qualified tools, utilization, and variability) to be a useful parallel approach. We don't feel that the two approaches are inconsistent; more that they focus on different estimations of underlying data. FabTime's focus remains on data that we feel can be collected relatively easily by wafer fabs using their current systems. For those interested in more information about this Technical University of Eindhoven research on the fundamental drivers of fab performance, please contact Jennifer.Robinson@-FabTime.com.

Setting Target Cycle Time Goals by Operation

Another subscriber wrote: "One of my current projects involved setting target cycle time goals for various operations. I've used TCT and X factor for this purpose so far but my approach has not been very scientific. I'm trying to find out if there is a scientific approach for determining the target CT, incorporating factors such as UPH, Equipment Quantity, Batch size, arrival rate, etc. Do you have any recommendations on what method to use? Are there any articles out there about this matter?"

FabTime Response: The most common approach that we have seen to setting operation cycle time goals is to take some x-factor of process time, and apply that globally across the fab. FabTime has done work on taking a more scientific approach to calculating operation-specific targets in terms of specific underlying factors. We've taken our simple operating curve spreadsheet (available from www.fabtime.com/charcurve.shtml) and expanded it to include multiple tools per tool group, hot lots, and batch arrivals. We also created a route-level version in which users can enter inputs for each operation as a separate row (with average utilization as another input). The spreadsheet then uses queueing formulas to generate cycle time estimates for each of these operations, and adds them up across routes.

The expanded version of the operating curve spreadsheet, and the route-level version, are currently only available to customers of our cycle time management course or our software. Our experience has been that although these queueing formulas can be quite useful in understanding how factors like arrival variability, number of hot lots, number of tools, and downtime characteristics will influence expected operation cycle time, the limitations of queueing formulas make it difficult to use this for setting goals for all operations. For instance, we don't model operator delays, or time constraints between process steps, or batch processing, because these are more difficult to handle in queueing formulas. To include such complexities, one generally needs to go beyond queueing models to simulation. And there we get into issues of collecting and maintaining very detailed data.

What our software customers can do in this area is use FabTime to collect information for them on actual x-factors by operations, and then look at where they believe that cycle time can be improved. We believe that this is the best approach in the long-term – to use as much actual data as possible, collected automatically so that people don't have to maintain it, and combine that with an understanding of how underlying factors like variability are likely to drive performance.

We would be interested to hear from other subscribers on this new topic.

Setup Avoidance and Dispatching

Introduction

Setups, both large and small, add day-to-day complexity to semiconductor wafer fabrication. A setup occurs when a tool requires some configuration change in order to be able to process lots of a particular recipe or operation. Setups are sometimes sequence-dependent, meaning that the length of the required setup depends on both the current recipe ID configured on the tool and on the new recipe ID. For example, implanters commonly require sequence dependent setups when changed to a new species, where the length of the setup time depends on the previous species value. Many setups, however, are sequence-independent. That is, before the first lot of any recipe ID can be processed, a setup of some fixed length must be performed.

If we were to ignore setups in dispatching decisions, and process lots in, say, first-in-first-out order, setups would lead to a significant capacity loss for some tools. That is, time spent doing setups would cut down on the standby time for each tool, and would thus drive up cycle time. Remember: cycle time is proportional to 1 / (1 - utilization), where utilization = ((productive time)/ (productive time + standby time)). Tools with little or no standby time have utilization approaching 1, and hence have high cycle times. Therefore, fabs frequently undertake setup reduction initiatives, with the goal of significantly reducing or eliminating setups and their attendant complexity.

For this article, we treat setups as a given for the time being, and consider ways of organizing work in the fab to minimize their impact.

Setup Avoidance Dispatch Policies

Setup avoidance policies are commonly used in fab dispatching decisions, either explicitly or implicitly. Under a pure setup avoidance policy, the operator continues processing lots of the same recipe ID (the same operation), in order to avoid changing recipes and having to perform a setup. Only when there are no lots remaining in queue requiring that recipe ID does the operator perform a setup to allow processing of a different recipe. This type of setup avoidance policy can dramatically reduce the percentage of time that a tool spends doing setups.

However, there is a problem with implementing a pure setup avoidance dispatch policy, particularly for small toolgroups. Higher volume recipes will tend to dominate the dispatch list, leaving lots at lower volume operations to wait in queue for extended periods of time. This is a direct consequence of dispatch rules that say "don't perform a setup if there is any matching lot in queue." What sometimes happens is a kind of "soft" dedication, in which a tool ends up spending nearly all of its productive time processing lots of a single recipe type, even as other lots wait. In extreme cases, operators may be so reluctant to lose capacity by performing setups that they will hold a tool idle until other lots of the target recipe type arrive. This leads to unplanned single path operations (see Issues 6.05 and 3.03), and higher than expected average cycle times. This can sometimes be seen even in cases where the setup time is relatively short, and/or where there is sufficient spare capacity on the tool to perform the setups.

To avoid this problem (low volume recipes waiting too long), fabs have imposed various caps on the setup avoidance policy. For example:

- **Maximum queue time.** Avoid setups if possible. However, if any lot in queue has been waiting more than X hours, force a setup, so that the lot is processed next. Here X might be something like 24 hours, but would vary according to the loading of the tool.

- **Maximum number of lots processed.** Avoid setups if possible. However, once N lots of the same recipe ID have been processed in a row force a setup to a new recipe ID. Here N might be determined by using simulation or some other model, or by past experience.

- **Maximum number of lots in queue.** Avoid setups if possible. However, if there are more than Y lots of any recipe ID waiting in queue, force a setup to that recipe. Here the value of Y would depend on the utilization and average queue length of the tool.

- **Minimum or maximum number of tools dedicated to each recipe ID.** Here setup-avoidance is still the goal, but only lots that can be run without violating a minimum or maximum tool limit are eligible for processing. In general, specifying a minimum number of tools forces at least that many tools to be dedicated to a particular recipe ID, while specifying a maximum number of tools limits the number of tools that can be setup simultaneously for a particular recipe ID. Selecting the right values for these parameters can be quite complex.

- **Operation-level moves goals.** Avoid setups if possible. However, operators must also meet operation-specific moves goals during each shift, including moves goals for low volume recipes, including rework recipes. This policy can lead to conflicts, however, between avoiding setups and meeting moves goals.

The problem with all of the above approaches to capping the setup avoidance policy is that each requires calculation of one or more parameters (e.g. maximum queue time). Finding the best value for these parameters can be challenging in and of itself, and may require simulation to explore possibilities. More of a problem is the fact that things change rapidly in a wafer fab, and the right value to use today might not be the right value to use next week. Also, use of more than one of these methods for overriding setup avoidance can lead to very complex dispatch rules.

Managing Setups with Dispatch Factors

As discussed in Issue 6.04, FabTime's approach to dispatching is to allow flexible combinations of dispatch factors to be specified for each tool group, with different weightings granted to each factor according to the requirements of the tool group. The simplest way to implement setup avoidance under this scenario is to first use whatever other underlying dispatch factors are selected (for example, lot priority and due date) to score the eligible lots. Once the lots are sorted in dispatch factor order, pick the lot with the highest dispatch score to be processed next. Then check the queue to look for other lots of the same recipe ID (or operation), and move those up on the dispatch list to be processed following the first lot. Setups can be further reduced under this paradigm by adding additional factors, such as:

- **Quantity of lots with the same recipe ID.** Weighting this factor heavily favors lots that can be run as a large group without an intervening setup. Another option would be to allow a minimum quantity threshold, so that no weight is applied for lots that don't have at least N lots of the same recipe ID in queue.

- Recipe ID match with most recent track-in. Weighting this factor heavily favors lots with recipe IDs that match the most recent track-in for the tool and thus can be run immediately without a setup.

Left alone, however, these approaches are subject to the problem described above: some cutoff is needed to keep from running high-volume recipe IDs forever, and to force a changeover to lower volume recipes. To combat this problem, we need factors or constraints that force setups under certain conditions. In a perfect world, these factors and constraints would be as simple as possible, to reduce the complexity of dispatch troubleshooting.

Aside: Parallels between Setup and Batching Decisions

There are parallels between deciding when to changeover a tool to a new recipe ID and deciding when to form a batch. When we choose a recipe ID under a setup avoidance policy, we are implicitly forming a batch from the matching lots in queue. We're saying that we will process these lots all together, in order (though not all at the same time). This is similar to the decision that we make when we decide to run a particular batch ID, and we select lots with a matching batch ID from the queue.

In issues 2.1 and 3.8 we talked about the batch size decision, and concluded that it is a mistake to force the running of full batches on lightly utilized tools. The reason for this is that when you wait for a full batch before processing on a tool that is already available, the lots that are in queue incur additional cycle time while they wait for other lots. Then, when the full batch is finished and moves downstream as a group, there is often additional queue time at the next operation, while the batch is worked off. We have advised running batch tools under a greedy policy, which states that if the tool is available and has lots waiting, you should start the batch, even if not full. The greedy policy is robust, in that if you run a greedy policy on a heavily loaded tool, even if you run one or two smaller batches here or there, most of the time you will have enough lots in queue to end up running batches that are large enough.

The analogy here to a full batch policy is a strict setup avoidance policy, in which you never force a setup. The result is large "batches" of lots with the same recipe ID processed in sequence. However, the consequence is long queue times for the other lots. Unfortunately, the situation with setups isn't as clear as the situation for batching. That is, a pure "greedy" policy, in which you always do a setup to process the next lot waiting, is likely to lead to too much setup time for the tool. However, for lightly loaded tools, which have extra capacity that can be used to perform setups, it may be better for the average cycle time of all lots to force more frequent setups. The idea behind this article is to discuss systematic ways to help operators decide when to force setups, with a goal of reducing average cycle time for all lots.

Conclusions

In wafer fabs, setups are sometimes required when changing a tool from one operation to another. Setups are non-value-added time, and reduce the available capacity of the tool. Because any reduction in the available capacity of a tool tends to drive up cycle time, fabs are incentivized to avoid setups as much as possible. However, running a pure setup avoidance policy (in which a setup is only performed if there are no lots in queue with a matching recipe ID) can lead to long cycle times for low volume recipes. Various methods exist for forcing a setup to occur to prevent long queue times. However, these methods typically require the use of some tool-specific parameter, the appropriate value for which may change over time. We welcome your feedback on the most effective setup avoidance strategies.

Closing Questions for FabTime Subscribers

What is the bare minimum of complexity that can be incorporated into dispatch rules to account for setups, which will still yield good results for all lots? How do you handle this in your fab? Do you focus your efforts on reducing/eliminating setups?

Further Reading

- Y. Iwata and S. C. Wood, Effect of Fab Scale, "Process Diversity and Setup on Semiconductor Wafer Processing Cost," IEEE 2000 Advanced Semiconductor Manufacturing Conference (ASMC '00), 237-244, 2000.

- D. Rohan, "Machine Dedication under Product and Process Diversity," *Proceedings of the 1999 Winter Simulation Conference*, 897-902, 1999. (All WSC papers since 1997 are available for free download from www.informs-cs.org/wscpapers.html).
- L. Solomon, J. W. Fowler, M. Pfund, and P. H. Jensen, "The Inclusion of Future Arrivals and Downstream Setups into Water Fabrication Batch Processing Decisions," *Journal of Electronics Manufacturing*, Vol. 11, No. 2, 149-159, 2002.
- R. Sunkara and R. Rao, "A Heuristic to Determine Equipment Setup Changes Based on Estimated Lot Arrivals in a Semiconductor Fab," *Proceedings of the 2004 Winter Simulation Conference*, Washington, DC, Dec. 5-8, 2004. (All WSC papers since 1997 are available for free download from www.informs-cs.org/wscpapers.html).

Subscriber List

Total number of subscribers: 1898, from 420 companies and universities. 25 consultants.

Top 10 subscribing companies:

- Intel Corporation (99)
- Analog Devices (79)
- ATMEL Corporation (62)
- Infineon Technologies (60)
- STMicroelectronics (57)
- Freescale Semiconductor (55)
- Philips (47)
- Micron Technology (45)
- Texas Instruments (42)
- TECH Semiconductor (39)

Top 3 subscribing universities:

- Virginia Tech (10)
- Arizona State University (9)
- University of California – Berkeley (7)

New companies and universities this month:

- Molnlycke Health Care
- Sensors Unlimited

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

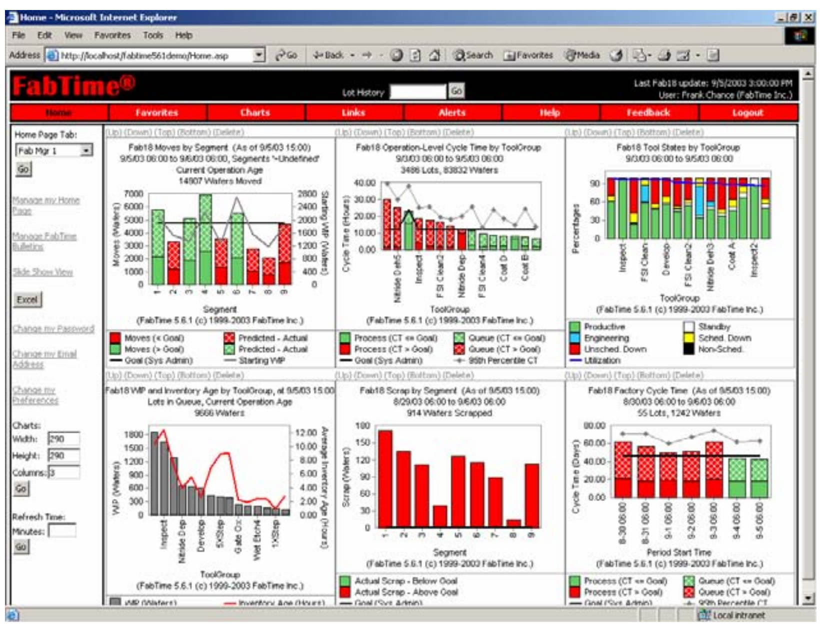
Do you have the best possible information?

- 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%.