FabTime Cycle Time Management Newsletter

Volume 5, No. 5 May/June 2004

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 the current version (6.1) include automatic breakdown of standby time into "standby-WIP-waiting" and "standby-other", and new WIP utilization outputs.

Editor: Jennifer Robinson

Contributors: Katja Schultz (IZET); Dan Fullerton (Eastman Kodak); Jimmy Giles (STMicroelectronics); Peter Gaboury (STMicroelectronics); V.A. Ames (Productivity System Innovations (PSi)); Jimmy Martin (Analog Devices)

Table of Contents

Welcome

Welcome to Volume 5, Number 5 of the FabTime Cycle Time Management Newsletter! We hope that you're having a great summer. In this issue, we have several responses to last month's subscriber discussion question, about breaking up standby time according to whether or not WIP is available. We also have a new subscriber discussion question about the closest-to-completion-time dispatch rule. We extend our thanks to the people who took time to write in this month, and hope to hear from more of you in the future.

In our main article this month, we propose a new metric for tracking shift-level use of individual tools by operators, called WIP Utilization%. This metric was developed jointly by Frank Chance of FabTime and Jimmy Martin of Analog Devices. We define WIP Utilization% as Productive Time / (Productive Time + Standby WIP Waiting Time). This is similar to our definition of Utilization, which is Productive Time / (Productive Time + Standby Time). However, in the denominator, we only include the standby time in which WIP is waiting for the tool. WIP Utilization% will approach 100% if, whenever WIP is waiting, and a qualified tool is available, the WIP is processed as soon as possible. Driving WIP utilization to 100% generally minimizes per-visit cycle times through the tool, and helps to maximize shift-level throughput. This metric overcomes several shortcomings of the standard utilization definition as a shift-level metric for operators. We hope that you find it useful.

Thanks for reading!—Jennifer

FabTime

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

Community News/Announcements

OpVent 2004 Conference Announcement

Katja Schultz of IZET Innovations-zentrum Itzehoe sent us the following announcement about IZET's upcoming OpVent 2004, an International conference on business opportunities, future trends, and financing. The conference will be held June 16th-18th in the Hamburg, Germany Metropolitan Area.

"Your ideas and opportunities are the pivot point of OpVent 2004 at IZET Innovationszentrum Itzehoe. An optimal platform to make topgrade business contacts and pave the way for your success. OpVent 2004 offers exciting opportunities for you and your entrepreneurial visions. You will get the chance to meet international experts, financiers and potential partners and to expand your networks.

A rich programme is awaiting you:

- International business meeting "Community-Treff"
- Lectures on the topics of markets and trends in the metropolitan area of Hamburg as well as expansion and financing
- Sportsevents: Golf tournament – Innovation Cup of Itzehoe – or a sailing trip on the Elbe River

More information and registration information are available at www.opvent.com, by phone +49.4821-778540 or by emailing schultz@izet.de"

Benchmarking for Probe Cycle Times

FabTime has been approached by a probe manufacturing facility that is seeking to benchmark their cycle time performance with other probe areas. They would like to better understand where they are in terms of cycle time performance, relative to other probe areas, and to mutually share ideas for performance improvement. They would also like to get an understanding of the metrics that different probe areas use to monitor performance. If you are a probe manufacturing area, interested in undertaking this type of one-on-one benchmarking, please contact Jennifer Robinson (Jennifer.Robinson@FabTime-.com) to be put in contact with the other company. Note that benchmarking would be contacted directly between your companies – FabTime will simply provide the introduction.

FabTime Adds PROMIS Support

We are pleased to announce data link support in our FabTime cycle time management software for sites using the PROMIS manufacturing execution system.

New Mailing Address for FabTime

FabTime's physical mailing address has changed to:

815 Greystone Place
San Luis Obispo, CA 93401.

Corporate phone (408-549-9932) and fax (408-549-9941) remain the same.

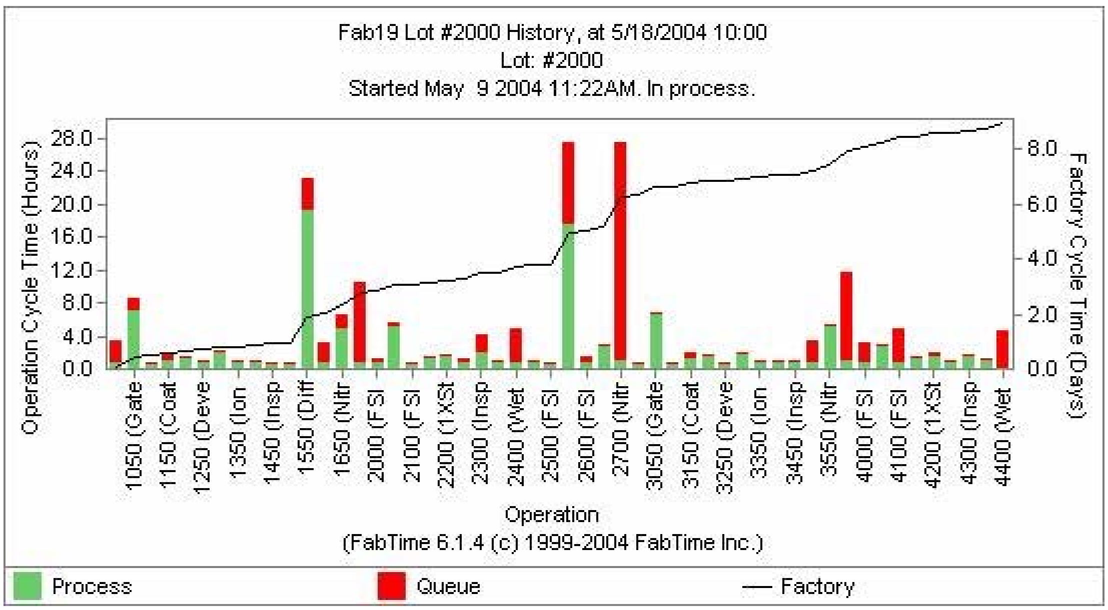
FabTime welcomes the opportunity to publish community announcements. Send them to newsletter@FabTime.com.

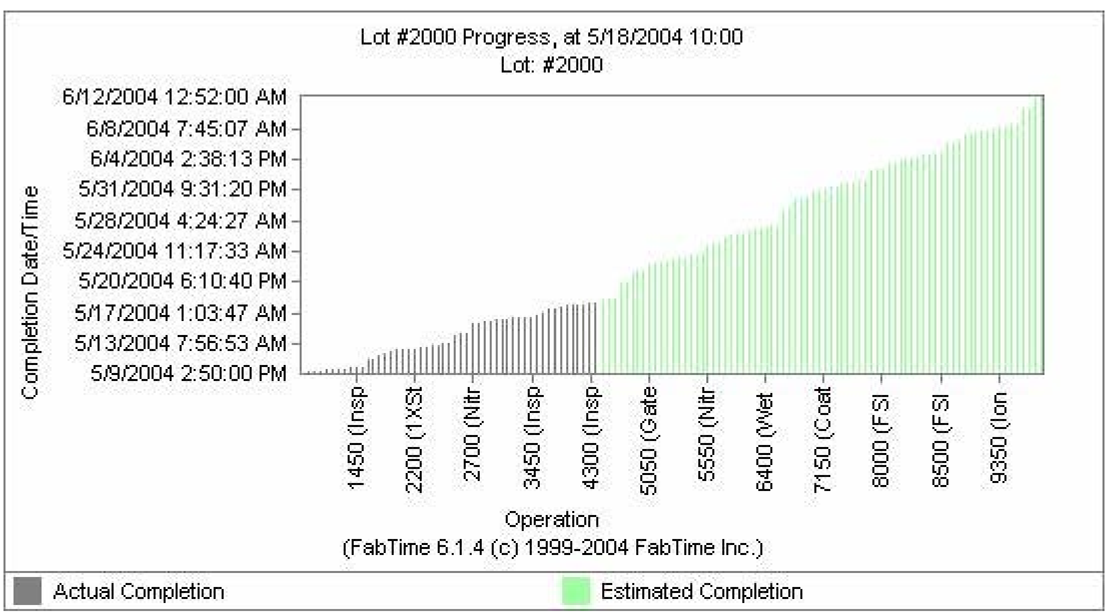
FabTime User Tip of the Month

Access a Lot's History from Anywhere in the Software

In response to a recent customer request, we've added the capability to quickly access a Lot History chart for any individual lot from anywhere in the software. Simply type the lot name in the "Lot History" input box, located in center of the black bar above the red FabTime toolbar, and either hit "Enter" or press the "Go" button. FabTime will immediately take you to the Lot History chart. From the lot history chart you can see where the lot is now, as well as how long the lot took to complete each previous operation (broken out into queue time vs. process time). You can also "Quick Jump" to the Lot Progress chart, which gives you a rough estimate of completion time for future operations (based on planned cycle times for the remaining operations on the lot's route). Examples of the both lot charts are shown on the next page.

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

Closest to Completion Time Dispatch

Dan Fullerton (Eastman Kodak) wrote: "As we embark on another round of cycle time reduction efforts here at Kodak in Rochester, NY, one of my engineers relayed an experience he had many years ago. In his fab, they achieved excellent theoretical cycle time multipliers by implementing a queueing strategy where the run with the highest completion percentage always went to the top of the queue. Philosophically, it sounds to me like a reasonable way to operate at low fab utilizations where a very simple queueing system is required, but intuitively I would think this could disrupt line balance and be grossly inefficient at higher utilizations. I was wondering if you had any experience or thoughts around such a system that you might be willing to share? Thanks in advance for your time and any assistance/insight you may be able to provide."

FabTime Response:

We have certainly heard of this dispatch rule, in which you always run the lots that are closest to completion first. However, we've seen very little published about the application of such a dispatch rule, particularly in a wafer fab. It does seem to me that it could unbalance your line. In the short-term, you would process lots closer to the end of the process flow, and would end up leaving the later portion of the line empty. You would probably see a short-term improvement in shipped lot cycle time when you first started using this rule. However, this would be at the expense of the cycle time of the other lots still in process. In most cases, that's what a dispatch rule does - improves cycle time of one set of lots by shifting queue time over to some other set of lots. There are some exceptions, of course, particularly where setups or batching are concerned. However, if anything it seems like a closest to completion time rule would impair performance in these areas (e.g. if you followed it strictly, you would probably have more setups). In practice you would probably modify the rule to cap lot queue time. For example, process closest to completion unless there is a lot with queue time > X hours, in which case process the oldest lot. So, without having studied this in detail, we would tend to agree with your impression – that in a fab with higher utilizations, a strict closest to completion dispatch rule is more likely to hurt you than to help you in the long term. For our academic readers – do you know of research that has been published on this topic? Are there publicly available results from high utilization fabs running closest-to-completion dispatching?

Standby No Operator Time

Last month FabTime raised this discussion topic. "One of the SEMI E-10 tool states is "standby time". This is time when the tool is available to manufacturing, but is not processing WIP. The question is: do you break this standby time into more detailed categories? In practice, some of this standby time occurs because there is no WIP available. However, sometimes the tool is in a standby state even when there is WIP available that could be processed ... We would be interested to know whether or not you break out this time in your tool status reports, and if so, how accurate you have found it to be."

The responses that we received are included below. We have not included direct FabTime responses to these comments. However, our main article this month relies heavily on breaking down standby time into more detailed buckets according to whether or not WIP is waiting. This is something that we do implicitly in our software (by linking the WIP transaction history to the tool state data), so that people do not have to rely on additional operator logging.

Jimmy Giles (STMicroelectronics) wrote: "In regard to your question about subdividing stand-by, in Carrollton we don't subdivide the types of stand-by time; however, there is a great emphasis to automate tools to autolog to stand-by upon completion of a production cassette (thereby capturing the time when an Operator is not available to log the unload in Workstream)."

Peter Gaboury (STMicroelectronics) wrote: "Thought I would sound in on standby time. Personally speaking, I think you didn't ask a detailed enough question; I think the first question you should have asked was how do you accurately measure standby time. After you understand if you can accurately measure standby time, then you can reply to the question whether or not you break it into other categories.

In response to "my question":

To accurately measure standby time, you can do 2 things:

1. When automation receives a LOT END from the equipment, it uses a remote MES transaction (WorkStream in our case) to change the status of the equipment to IDLE. In the same manner, when you start a lot, automation uses a remote transaction to put the equipment in processing.

2. You can merge the MES data with automation data and create a consolidated state of the equipment. There was lots of work done on this subject by SEMATECH with TP2.

For the above 2 solutions, There are some modeling issues to tackle in front of this like cluster tools, photo clusters (track + stepper) and tools operating in cascading mode. You just need to define some rules and work a bit.

In response to your question, once you have implemented a system to accurately measure standby, to break it down into more meaningful categories you can do several things:

1) You can compare standby time and WIP levels. Put a tolerance on WIP level, and do queries like Machine in IDLE + WIP < Key value = no WIP, > Key value = No operator. Tools like the APF reporter from Brooks let you do this pretty easily.

2) Instead of putting the tool into IDLE, you put the tool into NO OPERATOR. This is what I call the "Hot Potato" Game: putting the "Hot Potato" into the hands of the operator. The object of Hot Potato is of course to get rid of the potato, so, the operator is motivated to enter the real status of the equipment in replacement of the NO OPERATOR status. If the operator is not there to enter the real status, ... Guess what, it is no operator. There are several variations on this to simplify and gain time changing the equipment state. This is a quite "controversial" solution and requires a mature and responsible manufacturing organization ...

3) We are currently exploring a way to use Brooks RTD in conjunction with automation. This should give us a real measure of no operator / no WIP / no qualified WIP. The advantage of this type of solution is RTD knows very well how many wafers are qualified and sitting in front of a piece of equipment.

I also know there were several papers published by IBM about what they call their "CACTUS" system. I am sure someone from IBM will share with you about this. Hope you got some value from this feedback."

V.A. Ames (Productivity System Innovations (PSi)) wrote: "The stratification of data is always useful if you have a specific purpose. Although different sites may have their reasons for collecting data, there are two primary purposes for all sites to break out equipment Standby Time into No WIP and No Operator that I feel are very useful and important. These purposes are:

1. No WIP (all tools) – As SEMI E79 describes, Production OEE can be calculated by subtracting the No Wip time from the Total Time in the denominator of the OEE calculation (THT X Good Wafers) / (Total Time - No WIP time). Production OEE is extremely useful because it is a true comparative measurement of any tool's performance in the fab, whereas OEE is typically most useful only on bottleneck tools because there should be losses present for No WIP time.

2. No Operator – This is most useful on bottleneck and constraint tools. They should always have WIP and not be idle at any time. Tracking No Operator can give you insight into manpower issues, but it makes sense to me to have some no operator time on non-bottleneck tools because these resources can be diverted to bottleneck tools if needed.

WIP Utilization Percentage

By Frank Chance (FabTime), Jimmy Martin (Analog Devices), Jennifer Robinson (FabTime)

Introduction

When we talk with people who work in fabs, they frequently ask us to recommend shift-level metrics for operators that, if improved, will drive cycle time improvement (or at least not hurt cycle time). One value that tracks very closely with operation-level cycle time at the tool level is tool utilization (which FabTime defines as Productive Time / (Productive Time + Standby Time)). However, tool utilization is a problematic metric in this context, for several reasons that we will outline below. In this article, we propose a modified version of utilization called WIP Utilization%. WIP Utilization% overcomes several of the shortcomings of the standard utilization metric, and in particular rewards operators for running WIP as soon as possible during a shift.

Shortcomings of Tool Utilization as a Shift-Level Metric for Operator Performance

When we look at tool utilization values at the shift level, we have an inherent conflict. At least for bottlenecks, we want to run at high utilizations to maximize the value that we get out of these expensive tools. However, the higher the utilization, the higher the per-visit cycle times that we expect to see through the tool. So, for capacity purposes, we want to increase utilization, but for cycle time purposes we want to decrease it. We can decrease utilization by improving availability, and minimizing other capacity losses on the tool. However, this is outside the scope of what the operator can do during a shift. We would normally want the operator to drive to the highest possible utilization value during each shift. However, what this utilization value might be for different tools will be different depending on the respective tool loading conditions.

Another problem with utilization as a shift-level metric, especially for non-bottlenecks, is that it is independent of when WIP is run during the shift. Suppose we have two lots waiting to be processed on a tool at the start of an eight-hour shift, each requiring two hours of process time, and suppose that no other lots arrive during the shift. If the tool is available for the entire shift, then as long as we complete the processing of both lots during the shift, the tool utilization for the shift will be Productive / (Productive + Standby) = (2 lots * 2 hours/lot) / 8 hours = 4 hours / 8 hours = 50%. However, for cycle time, it would be much better for the operator to run the two lots as soon as possible at the start of the shift. This has two primary benefits. It moves the lots along to later operations (reducing their queue time at this operation). It also gets the tool ready to process other WIP, should something else happen to arrive during the shift.

Tool utilization, as defined above, treats all standby time as though it were the same. However, as discussed in the subscriber forum above, Standby No WIP time and Standby WIP Waiting time (or, Standby No Operator time) are not the same thing. For a non-bottleneck tool, we expect to have some standby time in which no WIP is available to be processed on the tool. This is not something for which we should penalize the operator – it's a planned condition that arises from the fact that we don't plan non-bottleneck tools to run at 100% utilization. However, when we do have WIP waiting in front of a tool, and the tool is idle, we're losing an opportunity. Therefore, we would like to have a tool utilization metric that treats standby time differently depending on whether or not WIP is available to be processed.

WIP Utilization% as an Alternative to Tool Utilization

In response to the shortcomings outlined above, we propose here a new shift-level metric for tool performance called WIP Utilization%.

WIP Utilization% = (Productive Time) / (Productive Time + Standby WIP Waiting Time)

Note that this is almost the same as FabTime's regular definition for utilization (Productive/(Productive + Standby)). However, we only include standby time when WIP is waiting in the denominator. This accomplishes several things:

1. WIP Utilization% will be 100% as long as we have no standby time during which WIP is waiting. This means that we can drive WIP Utilization% to 100% without hurting cycle time, because the tool can still have some standby time during which no WIP is available.

2. WIP Utilization% rewards operators for running available WIP as soon as possible during the shift. If, as in the example above, they leave WIP to run later in the shift, they will have a higher value for Standby WIP Waiting Time, and hence a lower value for WIP Utilization%. A detailed example is included below.

3. Because WIP Utilization% is independent of tool loading, fabs can set a similar goal for all tools (e.g. 90%, or slightly higher for tools with automated input queues, and slightly lower for all other tools). This says, for all tools, if you have WIP waiting and a qualified tool is available, you should try to get the WIP moving. Setting the goal to less than 100% reflects the reality that operators cannot be in more than one place at one time, and will sometimes have to let tools sit idle with WIP waiting.

Note: In the boundary case where Productive Time and Standby WIP Waiting Time are both zero, we define WIP Utilization% to be 100%.

Example

Suppose that we have a single-path, non-batch tool. At the start of a particular 12-hour shift there are four lots ready to be processed, each requiring one hour of process time. The tool remains available during the entire shift (no downtime). Eight hours into the shift, two additional lots arrive, each also requiring one hour of process time. Let's look at four possible cases, each assuming first-in-first-out dispatching.

Case 1: The Operator Processes all WIP as Soon as Possible

The first four lots are processed in sequence, at the start of the shift, requiring four hours of process time. The first lot incurs no queue delay during the shift (because it is processed right away). The other three lots incur one, two, and three hours of queue delay, respectively. When the two additional lots arrive, the first four lots are gone, and so the two additional lots are processed right away, in sequence, requiring two additional hours of process time. The first of these two additional lots incurs no queue delay, and the second incurs one hour of queue delay (while the first is being processed). Total Productive Time during the shift is six hours (since all six lots are processed during the shift, and each has a one-hour process time). Total Standby Time is six hours. Standby WIP Waiting Time is zero (whenever there is WIP waiting, the tool is working).

Regular Utilization% = Productive / (Productive + Standby) = 6/(6+6) = 6/12 = 50%

WIP Utilization % = Productive / (Productive + Standby WIP Waiting) = 6/(6+0) = 6/6 = 100%

Average Queue Time per Lot During the Shift = Sum(Individual Lot Queue Times)/6 Lots = (0+1+2+3+0+1)/6 = 7/6 = 1.167 hours/lot

Case 2: The Operator Begins Processing at the Four Hour Mark

The operator waits four hours to start work on this tool (presumably, is busy on other tools). At that point, the four waiting lots are processed in sequence, requiring four hours of process time. Each of these lots has four more hours of queue delay, respectively, than in Case 1. The two additional lots arrive immediately after that, and are processed right away, in sequence (and so have the same zero and one hours of queue delay). Total Productive Time during the shift is still six hours. Total Standby Time is also still six hours. However, Standby WIP Waiting Time is now four hours. Only the two hours at the end of the shift represent standby time in which no WIP is waiting.

Regular Utilization% = Productive / (Productive + Standby) = 6/(6+6) = 6/12 = 50%

WIP Utilization % = Productive / (Productive + Standby WIP Waiting) = 6/(6+4) = 6/10 = 60%

Average Queue Time per Lot During the Shift = Sum (Individual Lot Queue Times)/6 = (4+5+6+7+0+1)/6 = 23/6 = 3.833 hours/lot

Case 3: The Operator Begins Processing at the Six Hour Mark

The operator waits six hours to start work on this tool (presumably, is busy on other tools). At that point, the four waiting lots are processed in sequence, requiring four hours of process time. The first lot incurs six hours of queue delay, with seven, eight, and nine hours, respectively, for the other three lots. The two additional lots arrive during that time, and are processed immediately after the first four lots are completed, in sequence. The first of these additional lots waits two hours to be processed (while the third and fourth lot of the previous arrivals are being processed), and the second lot waits three hours. Total Productive Time during the shift is still six hours (the operator is just able to complete processing on all six lots, wrapping up right at the end of the shift). Total Standby Time is also still six hours. However, Standby WIP Waiting Time is now six hours. There is no standby time in which no WIP is waiting.

Regular Utilization% = Productive / (Productive + Standby) = 6/(6+6) = 6/12 = 50%

WIP Utilization % = Productive / (Productive + Standby WIP Waiting) = 6/(6+6) = 6/12 = 50%

Average Queue Time per Lot During the Shift = Sum (Individual Lot Queue Times)/6 = (6+7+8+9+2+3)/6 = 35/6 = 5.833 hours/lot

Case 4: The Operator Begins Processing at the Eight Hour Mark

The operator waits eight hours to start work on this tool (presumably, is busy on other tools, and decides to leave the four lots to be processed at the end of the shift). At that point, the four waiting lots are processed in sequence, requiring four hours of process time, and incurring eight or more hours each of queue time. The two additional lots arrive during that time, but cannot be processed during the shift, because there is no time left by the time the first four are completed. Each of these two lots thus incurs four hours of queue time (from arrival at the eight hour mark until the end of the 12-hour shift). Total Productive Time during the shift is now only four hours. Total Standby Time is now eight hours. Standby WIP Waiting Time is also eight hours (because the first four lots were waiting during the whole eight hours).

Regular Utilization% = Productive / (Productive + Standby) = 4/(4+8) = 4/12 = 33%

WIP Utilization % = Productive / (Productive + Standby WIP Waiting) = 4/(4+8) = 4/12 = 33%

Average Queue Time per Lot During the Shift = Sum (Individual Lot Queue Times)/6 = (8+9+10+11+4+4)/6 = 46/6 = 7.667 hours/lot. Also note that two of the lots are still queue at the end of the shift, and at least one of them will incur additional queue time during the next shift.

Summary

1. (Process as soon as possible). Util. = 50%, WIP Utilization = 100%, Avg Queue = 1.2 hrs

2. (Wait 4 hours to start). Util. = 50%, WIP Utilization = 60%, Avg Queue = 3.8 hrs

3. (Wait 6 hours to start). Util. = 50%, WIP Utilization = 50%, Avg Queue = 5.8 hrs

4. (Wait 8 hours to start). Util. = 33%, WIP Utilization = 33%, Avg Queue = 7.7 hrs

So, what we observe from this example is that the WIP Utilization% is only 100% when the operator processes all available WIP as quickly as possible. This corresponds to the lowest average queue delay per lot of the four cases (and the best possible outcome for queue delay in this example). Despite being the best possible outcome in regards to queue delay and throughput, the regular utilization value is only 50%. If the operator delays processing for four or six hours, the regular utilization value remains constant at 50%, but the WIP Utilization% degrades sharply, in concert with the increase in average queue time per lot. If the operator delays processing too long (eight hours), then throughput for the shift suffers, and both regular utilization and WIP Utilization% degrade further. This is also clearly the worst case in regards to queue delays, especially since there are two lots left in queue at the end of the shift. If we were only comparing these cases based on the utilization value, we would consider the first three cases to be equivalent.

Treatment of Delays in Unloading

We need to be careful in our treatment of delays in unloading a tool. For example, if elapsed time from start-process to end-of-unload is all categorized as Productive, then one way to achieve 100% WIP utilization is to start processing a lot on a tool at the beginning of the shift, and leave the unload until the last possible moment in the shift. Obviously the moves performance for this tool will suffer, and WIP utilization will quickly be discarded as a metric.

If your automation system delivers a signal when a lot is finished processing and ready for unload, then you can address this issue by changing the tool state from Productive to a non-productive state at this time (perhaps standby-WIP-waiting or standby-no-WIP depending on lots in queue) – see Jimmy Giles' and Peter Gaboury's comments above.

If you do not receive an automation signal when lots finish processing, then you will need to use a metric such as Production OEE (see V.A. Ames' comments above) or moves per productive hour (UPH) in addition to WIP utilization. Both of these metrics should be sensitive to significant delays in unloading.

Alternatives

By definition, WIP Utilization% can only reach 100% if Standby WIP Waiting Time is zero. So an alternative would be to measure and report Standby WIP Waiting Time – or Standby WIP Waiting Time as a percentage of available time – and seek to drive these metrics to zero. These are certainly valid alternatives, but our experience suggests that metrics where higher values are better than lower values are more intuitive than those where the scale is reversed. For example, consider line yield%. It's equivalent to measure scrap% = (100% - line yield%), but line yield appears to be the more appealing metric.

Conclusions

We know that not all tools in a fab are going to be operated at 100% utilization. This is a good thing, because if we were to plan for 100% utilization, we would end up with unacceptably high cycle times. But what we would like to see, regardless of a tool's planned utilization value, is that whenever there is WIP waiting in front of the tool and the tool is available, the tool is busy processing. The metric WIP Utilization% = Productive Time / (Productive Time + Standby WIP Waiting Time) drives toward this behavior. It motivates operators to run WIP as soon as possible during the shift, and does not penalize for standby time when no WIP is available. It is also independent of planned tool utilization values, and can be set to the same target across many different tools in the fab. This, we think, makes it a nice metric to consider for shift-level operator performance relative to the use of individual tools.

Closing Questions for FabTime Subscribers

Do you think that WIP Utilization% makes sense as a shift-level metric? Have you used anything like this in your fab? If so, has it been successful?

Subscriber List

Total number of subscribers: 1636, from 390 companies and universities. 25 consultants.

Top 10 subscribing companies:

- Analog Devices (79)
- Intel Corporation (76)
- Motorola Corporation (57)
- Infineon Technologies (50)
- STMicroelectronics (49)
- Philips (46)
- Seagate Technology (41)
- Micron Technology (40)
- Texas Instruments (39)
- AMD/Spansion (35)

Top 4 subscribing universities:

- Arizona State University (11)
- Virginia Tech (10)
- Technical University of Eindhoven (6)
- University of California – Berkeley (6)

New companies and universities this month:

- BENOITE Developments
- BOC Edwards
- Calient Optical Components
- Mills & Associates Consulting
- TOCGC
- TU Bicycle
- University of the Philippines

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 available for a small fee from FabTime's Amazon zShop, at www.amazon.com/shops/fabtime.

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.
325M Sharon Park Drive #219
Menlo Park, California 94025
Phone: +1 (408) 549-9932
Fax: +1 (408) 549-9941
Email: Sales@FabTime.com
Web: www.FabTime.com

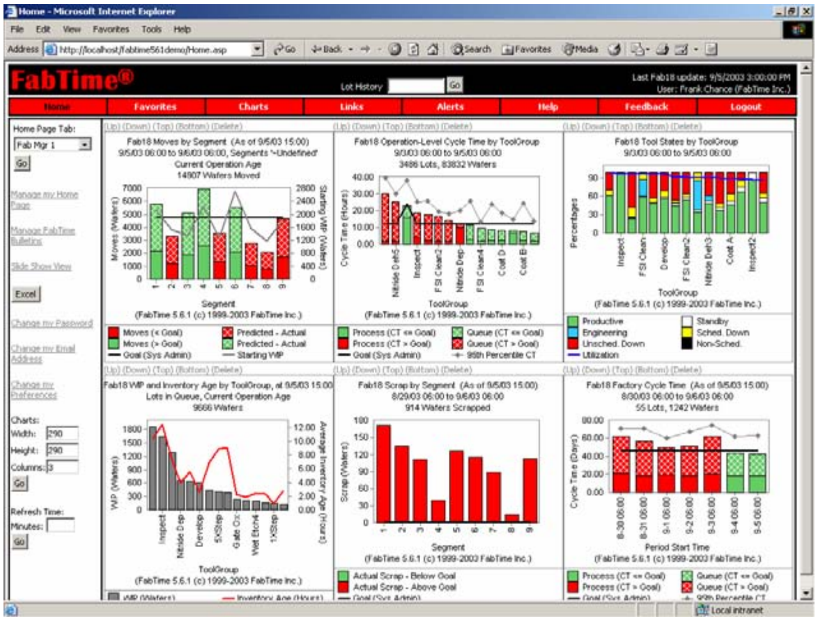
Do you have the best possible information?

- 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.

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%.