FabTime Newsletter

Volume 22, No. 4 October 2021

Information

Publisher: FabTime Inc. FabTime sells cycle time management software for wafer fab managers. FabTime's mission is to help the people who run fabs improve performance by 1) letting them configure their own charts, so that they don't need assistance from IT for each new data request; and 2) including them in a community of people around the world working to improve fab operations.

Editor: Jennifer Robinson

Date: Tuesday, October 12, 2021 – Vol. 22, No. 4

Keywords: Queueing Models, Variability, Single Path Tools, Utilization, Tool Qualification, Dispatching

Table of Contents

Welcome

Welcome to Volume 22, Number 4 of the FabTime Cycle Time Management Newsletter. We hope that life is in a relatively normal state where you are in terms of Covid restrictions, and that we'll all start having the chance to see one another more in person in the coming year.

In this issue we have an announcement about a promotion for Jean Paul Tu, now Principal Engineer for FabTime and another about the Fab Owners Alliance. We also share a few recent news stories from Jennifer's LinkedIn feed. Our FabTime software tip of the month is about customizing data tables using our relatively new data table engine, Ag-grid. We also introduce a new subscriber discussion topic about the application of critical ratio dispatching to wafer fabs.

In response to the continuing chip shortage, we've seen interest in understanding and improving fab performance from many quarters. In our main article we return to basics, discussing the three fundamental drivers of wafer fab cycle time, with added detail from our cycle time management course. We hope you'll consider sharing this article with others.

Thanks for reading! – Jennifer, Frank, Lara, and the FabTime Team

Community News/Announcements

Promotion to Principal Engineer for Jean Paul Tu

FabTime is pleased to announce the promotion of **Jean Paul Tu** from Senior Applications Engineer to Principal Engineer. This promotion recognizes JP's extensive practical experience, and his growing responsibility for FabTime's core software. In this new role, JP will provide leadership and technical direction for the software development team. In addition to direct engagement with customers, JP will be responsible for identifying, evaluating, and prioritizing opportunities for major software enhancements. We believe JP has the knowledge and skillset necessary to deliver these enhancements, and in doing so, to provide our customers with measurable improvements in productivity.

FOA Back in Person in December

In a promising nod towards normalcy, the Fab Owners Alliance is scheduled to be back in-person in December, in conjunction with Semicon West 2021. Although the SEMI FOA team has done a fabulous job running virtual meetings for the past 18 months, Jennifer (who represents FabTime at the FOA meetings) is looking forward to seeing long-time FOA friends in person. Fingers crossed!

A Few Highlights from Jennifer's LinkedIn

Jennifer continues to share articles about business management, the semiconductor industry, and productivity improvement on her LinkedIn feed. Recent links have included:

- Neat to see Polar Semiconductor highlighted in this TV news story. One more data point in #waferfabs becoming better known in the mainstream press. Thanks to **Surya Iyer from Polar**, who shared the link at the September Fab Owners Alliance meeting.
- This is a quite detailed interview in The Verge about the chip shortage, starting from fundamentals about how chips are designed and manufactured.
- Here's a story about an interesting U.S. wafer fab capacity expansion project in Detroit. "The purchase and renovation of the 140,000 square-foot site will allow for the production of silicon carbide wafers, a type of wafer the company's parent does not make." Thanks to **Nick Dowd from Mitsubishi Chemical** for sharing the link on LinkedIn.

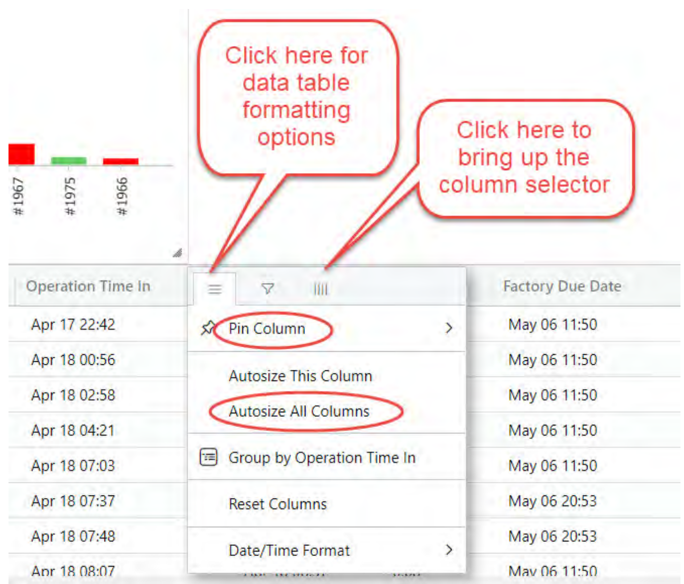
For more industry news, connect with Jennifer on LinkedIn: http://www.linkedin.com/in/jenniferrobinsonfabtime

FabTime welcomes the opportunity to publish community announcements, including calls for papers. Send them to newsletter@FabTime.com.

FabTime® User Tip of the Month

Customize Ag-grid Data Tables

Users of more recent versions of FabTime will have noticed changes to the data table layout and functionality. FabTime is now using a third-party product called Ag-grid for generating data tables. In this tip, we share a few suggestions for using and customizing Ag-grid data tables.



Sort the Data Table by a Column: Click on any column once to sort the data table in ascending order by that column. Click again to sort in descending order. Hold down the Shift key and click a second column to apply a secondary sort, repeating as needed for additional sort levels.

Remove Extra White Space: If you would like to tighten up the columns in a data table to reduce white space, and fit more on screen, put your mouse over any column header and click on the icon of three horizontal lines that appears. Select "Autosize All Columns".

Pin Columns: From the same menu, mouse over the "Pin Columns" row at the top and select "Pin Left" or "Pin Right". You can click on the header and drag to reorder the pinned (or unpinned) columns.

Remove Many Columns Quickly: Sometimes data tables have columns included that are not relevant for you. You can remove individual columns from the data table by dragging them up and off the data table. If you have many columns to remove, however, there is a quicker way. Click on "Columns" all the way to the right-hand side of the data table. Click twice in the checkbox at the top of the resulting dialog (next to the Search box). The first click selects all columns and the second deselects them all. Then scroll down to re-select only the individual columns that you need. You can alternatively get to the column selector by clicking on the three horizontal lines icon in any column, and then clicking on the icon that has four vertical lines.

Save Your Data Table Configuration (Patch 114 only): When you save a chart with a customized data table to your home page, FabTime saves your table configuration for that chart. In our newest version, Patch 114, you can also save the data table configuration as your default for this type of chart. On the right-hand side of the data table, at the bottom (you may need to drag the bottom of the data table down to widen the table to see it) there is a "Layout" option. Click on that and then click on "Save State as Default". There is also an option to reset the data table state.

We hope you find this tip (and the Ag-grid data tables in general) useful.

Subscribe to the separate Tip of the Month email list (with additional discussion for customers only) here: http://www.fabtime.com/tip-of-the-month.php. Thanks!

Subscriber Discussion Forum

Critical Ratio Values by Zone (Linear Stage of the Flow) in the Fab

We have been having a discussion with several members of our User Group about the use of the critical ratio dispatch rule in wafer fabs. A couple of our members have observed higher variance in critical ratio values for lots that are further along in the line. In one case, at least, this higher variance causes many of the lots late in the flow to not be prioritized relative to the lots early in the flow. We have speculated that these results may be an artifact of applying critical ratio in a reentrant environment with long process flows.

If we were using critical ratio (CR) in a non-reentrant flow, we would be comparing lots against each other at the same rough stage of the process (lot vs lot at first step in the flow, lot vs lot at second step in the flow, etc). If the behavior of CR changes across the line, it doesn't matter as much because we are still comparing apples (lots at step 1) against apples (other lots at step 1).

But in wafer fabs, we are comparing lots that are at radically different stages in their life, e.g., comparing lots at the first step in the flow, vs lots at the 227th step in the flow. If the behavior of CR changes between the 1st and 227th steps, then we are comparing apples and oranges.

We did a quick search and didn't find any research on critical ratio by zone. There wasn't much about CR in re-entrant environments in general, beyond one article by Oliver Rose from the 2002 Winter Simulation Conference about the application of CR to wafer fabs that discusses the importance of setting good due dates for CR. We thought we would ask you, our subscriber community, whether you looked at CR by zone in the fab or have any research specific to CR in re-entrant environments. Your responses may provide input for a future article on this topic.

FabTime welcomes the opportunity to publish subscriber discussion questions and responses. Simply send your contributions to Jennifer.Robinson@FabTime.com.

Fundamentals of Wafer Fab Cycle Time

Introduction

As we've mentioned previously in the newsletter, the chip shortage is putting considerable pressure on wafer fab cycle times at all technology nodes. In Issue 22.02 we shared our top ten recommendations for fab cycle time improvement. We do hope that people have found those helpful. We thought it would be worthwhile in this issue for us to take a step back and look at the fundamental drivers of wafer fab cycle time. We first wrote about these in Issue 6.05, but we've been discussing them most recently in our remote cycle time management course. For those who haven't had access to the course, this issue will give you the highlights.

The three fundamental drivers of cycle time in a manufacturing facility are tool utilization, variability, and number of qualified tools per tool group. At the tool group level, these are the primary factors that affect cycle time. Other secondary factors such as downtime and product mix affect cycle time through their impact on these three primary factors. Below, we will introduce each of the three factors, and suggest ways to influence them for performance improvement in wafer fabs.

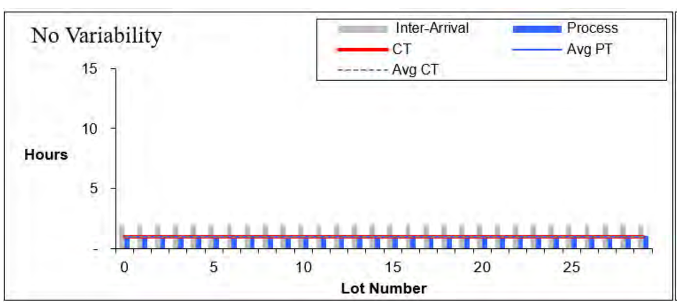
Tool Utilization

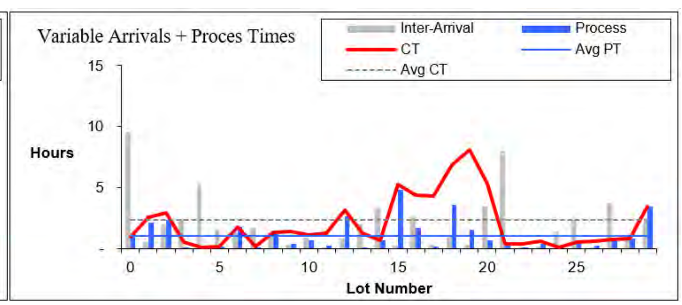
Utilization has a direct and non-linear impact on cycle times through a tool. Here we define utilization for a tool (as in previous newsletters) as Productive Time / (Productive Time + Standby Time). Productive Time is time that the tool is busy processing wafers. Standby Time is time that the tool is available, and hence could be processing wafers, but is not. Utilization under this definition is what drives cycle time.

In the presence of any variability, when standby time gets small relative to productive time, cycle time increases. As standby time approaches zero, cycle time gets very large. Intuitively, what happens is that we need the standby time to recover from variability. When there isn't much standby time, it takes longer to recover from variability, and cycle times suffer.

Consider a tool where lots arrive at a regular interval (the gray bars in the graphs below), with no variability and where process time (the blue bars) is always one hour, again with no variability. Whether the utilization is 50% (as shown) or 80% (or anything < 100%), the cycle time (the red line in the left-hand graph below) will always equal the process time.

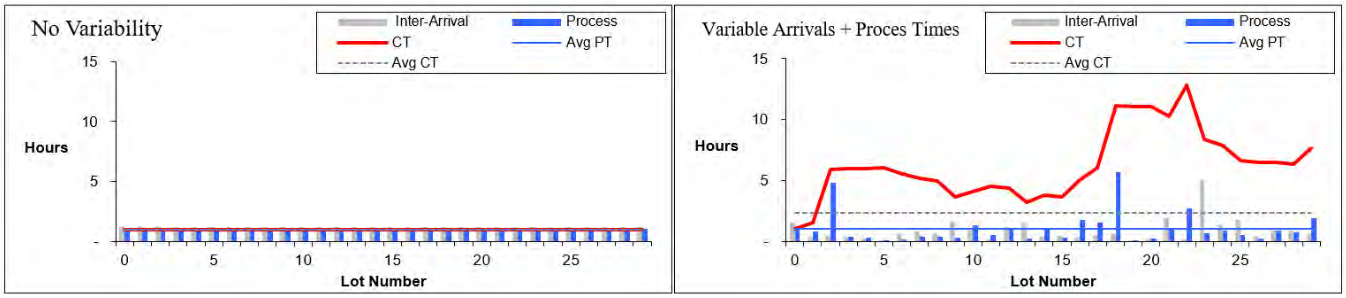
But now suppose the average process time is still an hour, but the process times and times between arrivals both have medium variability (exponential distributions). Now, even at 50% utilization, we see some queueing. In the chart shown on the right-hand side below, Lots 15, 16, 18, and 19 all have longer than average process times (the straight blue line is the average). Lots 17 and 19 have shorter than average times between arrivals (the dashed gray line). This creates a WIP bubble (the rising red line). Because the utilization is only 50%, there is plenty of standby time to recover from the WIP bubble before the end of the interval. In this case, the long time before the arrival of Lot 21 allows the system to recover.





We coded this into a spreadsheet tool using random numbers for the interarrival and process times. Running for a longer sequence of arrivals (230), and over a series of different replications of the random numbers (100), we achieved an average x-factor of 1.93 for the case with variability. The no variability case always has an x-factor of 1.

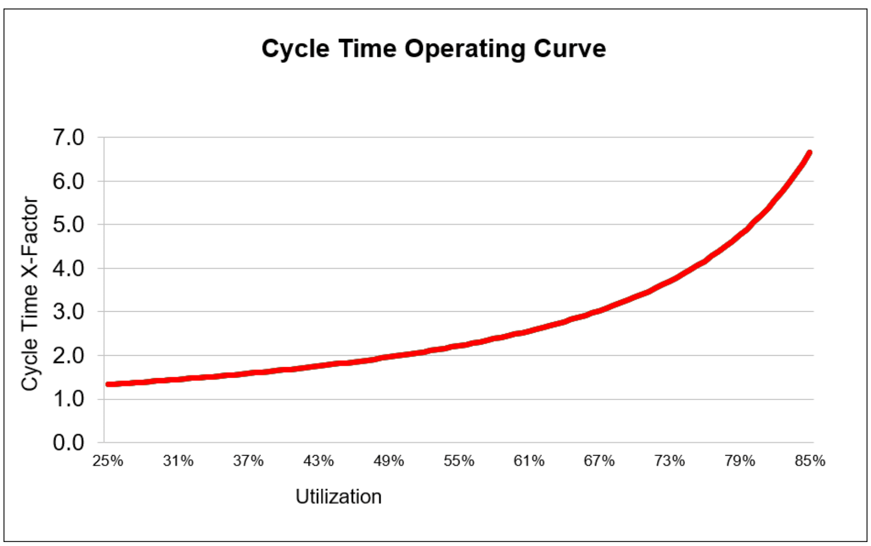
Suppose now that we increase the utilization of this tool to 80% (by reducing the average time between arrivals). At 80% utilization, we see more and longer WIP bubbles in the variability case. In the example below, the tool is unable to recover from an early long process time followed by a sequence of short arrival times. A second long process time for Lot 18 sends the WIP bubble even higher. In general, we have more WIP arriving and less standby time with which to recover. Again running this for a longer sequence of arrivals, and more replications, the average x-factor we observed at 80% for the variability case was 4.6. The x-factor for the no variability case, of course, remained at 1.



Running this simulation exercise at varying tool utilizations allows us to build an operating curve (a graph of cycle time x-factor vs. utilization) for this system. But there's an easier way to build the same operating curve using a simple queueing model. For a one-of-a-kind tool with no downtime under medium variability and independent arrival and process times (e.g., the tool does not get faster just because it is busier), the queueing formula is:

X-Factor ~= 1 / (1 – Utilization)

A graph generated using the queueing formula is shown below.



As standby time approaches zero, utilization approaches 100%, and the denominator of the above equation approaches zero. Then we have one divided by zero, which approaches infinity. In practice, x-factors in the fab don't get quite as high as those in this graph, because in the fab we don't have infinite WIP. In the simulated example above the system was capped at 230 lots. The x-factor observed at 80% utilization was 4.6, which is a bit lower than the 5.0 predicted by the 1 / (1 – Utilization) formula.

What we do see in practice in fabs is that as utilizations get above about 85%, cycle times start to become large. And because the operating curve is non-linear, even small increases in utilization lead to big cycle time increases at this point. This, we believe, is why so many fabs plan capacity such that most tools are loaded to no more than 85% (especially one-of-a-kind tools, which we'll get to below).

Fabs are under constant pressure to increase utilization, to make the most of the high capital cost of the equipment. Therefore, saying that you should reduce tool utilization in order to reduce cycle time does not immediately sound practical. However, remember the definition of utilization that we're using.

Utilization = Productive Time / (Productive Time + Standby Time)

The denominator is the time that the tool is available to manufacturing to process wafers. It's what you have left after you take out any downtime, engineering time, and non-scheduled time. Therefore, anything that we can do to reduce downtime, engineering time, and non-scheduled time will, if starts are not increased, directly increase standby time. This will reduce utilization, and hence improve cycle time.

One further note is necessary regarding standby time. What improves cycle time is having standby time as catch-up time on the tool. This means that everywhere that we've discussed standby time in the above, we should be more specific, and refer to standby time during which no WIP is waiting for the tool. This is true standby time. The tool is available but is not running because there is no WIP waiting to be processed. Having a buffer of such true standby time is helpful for cycle time improvement because it allows room to recover from variability. If, however, we have time reported as standby time on the tool, during which WIP was qualified and waiting for the tool, this time should be treated as a loss for the tool. In our software, we call this Standby WIP Waiting time. Often it occurs because there is no operator available to load the tool. Reducing time spent in this state and replacing it with true standby time will improve cycle times.

Variability

Variability also affects cycle times adversely. Variability takes the operating curve of cycle time vs. utilization and moves it upward and to the left. This means that for the same utilization, lots passing through higher variability tools will have higher average cycle times. There are many sources of variability in a fab, both in process times and in times between arrivals to tools. Contributors to process time variability include:

- Different recipes on the same machine, with different process times.
- Setups
- Equipment failures and maintenance events
- Rework lots
- Yield loss (scrap)
- Operators

Contributors to variability in arrivals to tools include:

- All the above (because departures from one step become arrivals to the next step)
- Transfer batching and automated material handling
- Batch processing (running multiple lots in a machine at one time)

Earlier we said that for a one-of-a-kind tool, the operating curve is shaped like: X-Factor ~= 1 / (1 - Utilization). This was a simplification of a more general formula:

X-Factor ~= 1 + [Utilization/(1-Utilization)] * [Variability Factor]

When the variability factor equals one, this reduces to the previous equation (1 / (1 - Utilization)). When the variability factor equals zero, the entire second term drops off, and we get X-Factor ~= 1. That is, we only have cycle time equal to theoretical cycle time when there is no variability. Any variability leads to increased cycle time, particularly when utilization is relatively high. The more variability, the higher the cycle time.

The variability factor is a sum of arrival variability and process time variability. More specifically, we have:

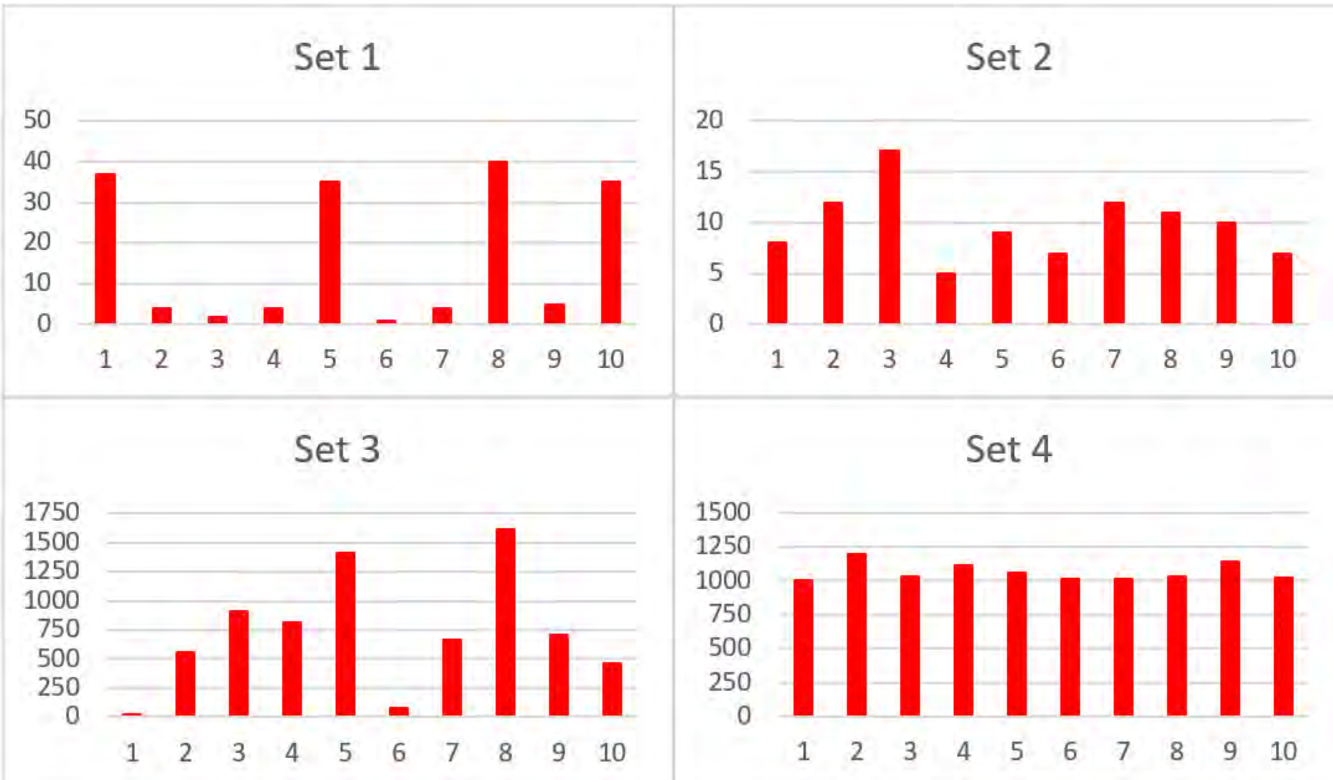
Variability Factor = $(CV_a^2 + CV_p^2)/2$

where $\mathbf{CV_a}$ is the coefficient of variation of interarrival times, and $\mathbf{CV_p}$ is the coefficient of variation of process times. Coefficient of variation (CV) is a statistical measure of how widely dispersed values are, where:

CV = Standard Deviation / Average

We observe high CVs when values are widely spread out. For example, arrivals to a tool immediately downstream from a large batch tool might have a high CV, because the sequence of times between arrivals looks like this: 0, 0, 0, 0, 0, 0, 0, 0, 12 hours. Here the zeros represent a batch arriving, with essentially no time between arrivals from lot to lot. CV is useful as a metric because it quantifies that variability into a number and makes it easier to compare.

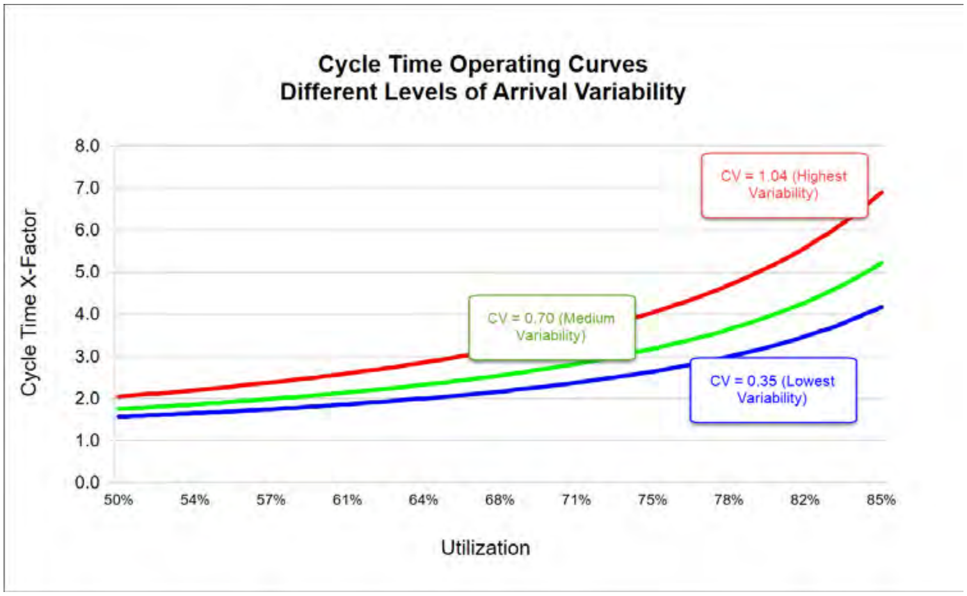
For example, suppose the four charts below represent four different sequences of interarrival times to a tool.



We can see immediately that Set 4 has the lowest variation between values. A bit more inspection reveals that Set 1 is has the most variation. But discerning between sets 2 and 3 is a bit more difficult. Enter the CV values:

- Set 1: 1.04
- Set 2: 0.35
- Set 3: 0.70
- Set 4: 0.06

In general, we can measure CV for times between arrivals or for process times and use the queueing formula above to view the impact of different levels of variation on the operating curve for a tool. FabTime has built a spreadsheet tool that uses a slightly more advanced version of that formula and allows users to enter up to three different scenarios. In this case (dropping Set 4), we can use the tool, **FabTime's Operating Curve Spreadsheet**, to see the following:

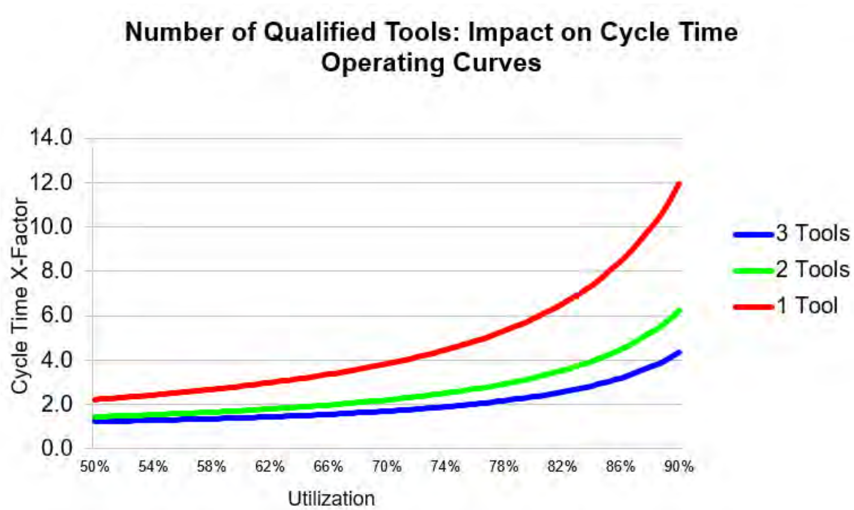


The difference in the curves would be the same if, instead of varying arrival variability, we varied process time variability, as those two are weighted equally in the formula. In either case, we can see that the impact of variability on cycle time through this one-of-a-kind tool is significant. The good news is that anything that we can do to reduce variability will tend to improve cycle times. Some concrete variability reduction suggestions (which have been discussed in more detail in other newsletter issues) include:

- Reduce transfer batch sizes.
- Eliminate minimum batch size constraints on batch tools that are not heavily loaded.
- Break up maintenance events, to avoid having tools unavailable for long, continuous stretches of time.
- Focus downtime improvement programs on reducing the duration of repair times.
- Spread out lot releases into the fab, instead of releasing lots in large groups.

Number of Qualified Tools per Tool Group

The third fundamental driver of cycle time at the tool level is tool qualification, or the number of tools available to process a lot at a particular recipe or operation. A recipe with only one qualified tool (often called a single-path tool, or a one-of-a-kind tool) will have higher average cycle times than a recipe with two qualified tools, even if the two tools each have the same utilization as the single tool. A recipe with two qualified tools will have higher cycle times than a recipe with three qualified tools (again, assuming the same utilization on all the tools), and so on, although the effect is most dramatic when going from one tool to two tools. We have observed that per-visit cycle times are often reduced by about 50% when going from single path to dual path, with a smaller reduction when going from two to three paths. An example generated using FabTime's Operating Curve Spreadsheet is shown to the left.



We've done a similar analysis to display the impact of qualifying from one tool up to ten tools at 90% utilization. This analysis also showed a nearly 50% reduction in going from single path to dual path, with smaller reductions going from two to three paths, and so on, gradually leveling off.

The intuitive explanation for this dramatic difference between single and dual path is that when there's only one tool, a lot is subject to all the variability associated with that tool (long process times, downtimes, etc.). When there are two tools, the probability of both tools being hit by a long delay at the same time is much smaller. The lot can usually be processed on the second tool. With three tools, the probability of all three being impacted at the same time is even smaller.

Real-world examples of this include:

- Shared line vs. dedicated lines at a store or an airport. The shared line is much faster, on average.
- Single lane road vs. multi-lane road. When you get caught behind a slow truck on a single lane road, you absorb all the variability from that truck. On a two-lane road, there is usually an opportunity to get by.

One other point about tool dedication is that "soft constraints" can also arise. Soft constraints are places where tools are dedicated in practice, even if there is no such official restriction. This can happen due to layout issues (where a tool group is broken up across the fab), setup minimization policies, or operator preferences for tools. Such soft dedication may make sense in many cases, but when it leads to single-path operations, it can be a hidden source of cycle time in the fab.

We wrote about the impact of tool qualification in more detail two years ago, in Issue 20.05. This article included the extension of the queueing formula for x-factor to account for more than one tool. We encourage those who missed that issue to download it from our website. The current password is "FabTimeCommunity" (no quotes). We have also, for the first time, made our Operating Curve Spreadsheet available to newsletter subscribers for download from the same location.

Conclusions

Most fabs these days are under pressure to produce more wafers AND get them to customers as quickly as possible. The fundamental factors that drive factory behavior, however, put these two goals into direct conflict. As tool utilization increases, particularly as it approaches 100%, cycle times become exponentially higher. People who run fabs must manage these competing pressures.

Two ways to do this are:

1. Extract standby time from non-value-added categories, including unscheduled downtime and time spent waiting for operators, and thereby increase time available for processing. Increasing standby time (equivalently, time available for processing), for the same amount of productive time, will decrease utilization, and allow more time for tools to recover from variability. This will improve cycle time without decreasing productive time (or throughput) or will allow for an increase in productive time (and an increase in throughput).
2. Reduce variability. Efforts to drive down variability help move a fab to a more favorable operating curve. This allows a fab to either get better cycle time at the same throughput rate or increase throughput at the same cycle time level.

The third fundamental driver of fab cycle time is number of qualified tools per tool group. Fabs that have many one-of-a-kind tools have a more difficult time achieving low cycle times. These fabs must trade off running the one-of-a-kind tools at a lower utilization rate or accepting higher cycle times. Variability reduction efforts will help these fabs, too, however, allowing them to push utilization of the one-of-a-kind tools higher.

Fabs that have single path operations due to process restrictions or soft dedication, on the other hand, can improve cycle time and/or increase utilization through efforts to qualify a second, and even a third, tool for each operation. Going from single path to dual path decreases cycle time through that operation by up to 50%. Identifying and removing single path constraints wherever possible is one of the most effective things a fab can do to improve manufacturing performance.

Fabs are complex, challenging environments. It's our view that understanding the fundamental principles that govern fab behavior can help cut through the complexity and focus improvement efforts. In summary:

- Cycle time increases nonlinearly with utilization (productive / (productive + standby)), particularly as standby time gets small. Anything that increases true standby time (not standby time when WIP is waiting) without reducing productive time will help the fab to run more efficiently.
- Anything that reduces variability in times between arrivals or process times will reduce cycle time at the tool level, giving the fab a choice to either improve cycle time or push upwards a bit on throughput.
- The cycle time cost of single path operations is high. Where single path is unavoidable (e.g., one-of-a-kind tools), decreasing utilization and/or variability can help to keep cycle times in check. Where qualification of a second tool is possible, making this a priority has significant benefit.

We hope you find this article useful.

Closing Questions for Newsletter Subscribers

Does your company's management push you to increase tool utilizations, to get more product out of the same tool set? Is there an understanding at your company of the impact on cycle time of driving utilization too high? Do the metrics that you use include variability metrics? Does your company have policies requiring a second (or third) qualified path before releasing a new flow to production? Do you have an easy way to highlight single path options?

Additional Resources

This article draws extensively from material in FabTime's half-day cycle time management course. The course is currently delivered remotely via Teams as two two-hour sessions for company sites. For more information, see FabTime's website.

See also FabTime newsletters 6.05 (The Three Fundamental Drivers of Fab Cycle Time) and 20.05 (The Impact of Tool Qualification on Cycle Time), both available from FabTime's newsletter archive. A more recent issue, 22.02 (10 Recommendations for Fab Cycle Time Improvement), includes a tip about how to identify "soft dedication."

The FabTime Operating Curve Spreadsheet is also available from the same archive (at the top of the page) for newsletter subscribers.

Subscriber List

Total number of subscribers: 2866

Top 20 subscribing companies:

- onsemi (204)
- Infineon Technologies (158)
- Analog Devices (includes Maxim) (131)
- Intel Corporation (122)
- Micron Technology, Inc. (122)
- GlobalFoundries (105)
- NXP Semiconductors (82)
- Carsem M Sdn Bhd (70)
- Microchip Technology (70)
- Skyworks Solutions, Inc. (69)
- STMicroelectronics (65)
- Western Digital Corporation Inc. (63)
- Seagate Technology (56)
- Texas Instruments (52)
- X-FAB Inc. (52)
- Wolfspeed, Inc. (43)
- Qualcomm (38)
- Tower Semiconductor (32)
- Hitachi Energy Ltd. (30)
- Honeywell (30)

Top 3 subscribing universities:

- Arizona State University (8)
- Ecole des Mines de Saint-Etienne (EMSE) (7)
- Virginia Tech (7)

New companies and universities this month:

- BI Norwegian Business School
- Cooper Vision Specialty Eye Care
- Fabmatics
- Memstronics
- Trumpf Photonic Components

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 to the newsletter. Past issues of the newsletter are now available in PDF for download by newsletter subscribers from FabTime's website. To request the current password, email your request to Jennifer.Robinson@FabTime.com, or use the Contact form.

To subscribe to the newsletter, send email to newsletter@FabTime.com, or use the form at www.FabTime.com/newsletter-subscribe.php. 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® Software: If you would like more information about our web-based dashboard for improving fab cycle times, please visit our website. A sample home page and a sample page from FabTime's new Charts menu are shown below.