FabTime Cycle Time Management Newsletter

Volume 6, No. 9 November 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 site-customizable alert formatting and enhanced FabTime server performance reporting.

Editor: Jennifer Robinson

Contributors: Dr. Jairo R. Montoya-Torres (Ecole des Mines de Saint-Etienne); Ulrich Dierks (AMD)

Table of Contents

Welcome

Welcome to Volume 6, Number 9 of the FabTime Cycle Time Management Newsletter! This month we have an announcement and call for papers for a conference to be held in France in the spring. I've also provided a brief review of the ISMI conference held in Austin last month. For me, one of the best things about this conference was the chance to meet several newsletter subscribers in person. My thanks to those of you who went out of your way to introduce yourselves! I am also repeating an offer to give a free one-hour talk on fundamentals of cycle time management for fabs in the U.S. This month's FabTime user tip of the month is about clearing Internet Explorer's temporary file folder to improve FabTime performance. Our subscriber discussion forum is brief this month, but contains an excellent extension to our WIP States proposal, submitted by Ulrich Dierks.

In our main article this month, we discuss uses for both actual and planned operation-level cycle time data. Actual values can be used to flag operations for which the ratio of cycle time to process time (x-factor) is higher than expected. Actual values can be compared with planned values, to identify short-term problems. Actual operation cycle times are also helpful in calculating planned cycle times. Planned cycle time values are in turn useful for estimating lot completion dates and facilitating dispatching decisions. In this article, we also introduce a new sidebar feature: a brief exercise for FabTime software users. Our goal is to make the newsletter articles more hands-on and relevant to our customers, the ones who make this newsletter possible.

Thanks for reading!—Jennifer

FabTime

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

Community News/Announcements

Conference Announcement and Call for Papers: INCOM-2006

Dr. Jairo R. Montoya-Torres of Ecole des Mines de Saint-Etienne submitted the following conference announcement. "The INCOM-2006 Conference (Information Control Problems in Manufacturing) which is sponsored by IFAC, IEEE, EURO, IMS, will be held in Saint-Etienne, France in May 17-19, 2006. This will be an Industrial Engineering conference mainly focused on solving manufacturing control, technology and information problems.

A special track on "Novel models and approaches in semiconductor manufacturing" is being organized. The track topics include (but are not limited to): Wafer production planning and scheduling, parallel and batch scheduling, lot sizing, resource qualification, capacity planning, design of semiconductor fabs, simulation for semiconductor manufacturing, lot transportation strategies, automated transportation, semiconductor Supply Chain Management (SCM), demand management, yield management, information systems for semiconductor manufacturing, process modeling, wafer quality management, Statistical Process Control (SPC), Advanced Process Control (APC). Two special sessions are proposed on "Automated transport" (organisers: Dr. J.R. Montoya-Torres and R. Sturm) and "production planning and scheduling" (organiser: Dr. Y. Mati). Complementary sessions will be organized in the final program.

The submission deadline is 15 November 2005. Submitted papers will be reviewed by at least two referees. Both academic and industrial oriented communications will be considered. Accepted contributions will be published in INCOM Proceedings by Elsevier. Further submission instructions and a detailed call for papers are available on the symposium website (http://www.emse.fr/incom2006)."

ISMI Manufacturing Productivity Conference

I (Jennifer) attended the ISMI manufacturing productivity conference last month in Austin, TX. The conference was very successful. There were about 300 attendees – a slight increase over last year's conference. What I personally like best about the conference is that the organizers keep the focus squarely on manufacturing-related issues. Most of the people who attend work for wafer fabs, and either directly or indirectly strive to improve manufacturing performance. I heard positive feedback about the conference from many attendees, and I hope that the ISMI is able to continue to host this conference next year. If you would like a copy of my conference presentation, just email me your request.

Free One-Hour Talk on Cycle Time Management at Your Fab

Are you kicking off a new cycle time improvement effort? Do you want to ensure that your production personnel are thinking about cycle time as they make operational decisions? Do you use the best metrics for improving cycle time? FabTime is currently offering to have Jennifer Robinson visit your site to give a one-hour talk on the factors that influence cycle time in wafer fabs, and the best metrics for cycle time improvement. This talk is a sub-set of our one-to-two day cycle time management course, and is being offered at no charge to fabs in the United States. The talk will be paired with a one-hour demonstration of FabTime's web-based digital dashboard software, with emphasis on ways that the software supports cycle time improvement efforts. If you are interested in scheduling a visit, please contact Jennifer.Robinson@FabTime.com.

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

FabTime User Tip of the Month

Delete IE's Temporary Internet Files to Improve Performance

When you use FabTime, you frequently move back and forth between different chart pages. If you use your browser's "Back" button to view a previously-displayed chart, the chart usually appears very quickly. This is because using the "Back" button tells Internet Explorer to display a page that is already stored in the page cache (temporary internet files) on your computer. IE doesn't have to send a request to FabTime, and FabTime doesn't have to rebuild the chart. However, we have worked with users for whom use of the "Back" button is slower, because instead of using the cache, their computer DOES send a request to FabTime. This can happen when you have too many temporary internet files stored on your computer. When this happens, IE stops saving new pages to the local cache, and must always build pages by sending a request to the FabTime server.

Fortunately, this is easy to fix. From inside Internet Explorer, select "Tools | Internet Options". In the middle of the dialog that displays, there is a section labeled "Temporary Internet Files." Click on the "Delete Files" button in this section. Depending on your IE version, you will probably get a dialog box that says "Delete all files in the temporary internet files". Leave the box labeled "Delete all offline content" unchecked, and click "OK". Wait until the hourglass goes away, and your cursor returns. This may take several minutes. Click the "OK" button to exit the internet options dialog.

This cleanup of temporary internet files is something that can help you with both FabTime performance and with general browsing. We recommend that even if you aren't having problems, you consider cleaning up your temporary internet files right now.

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

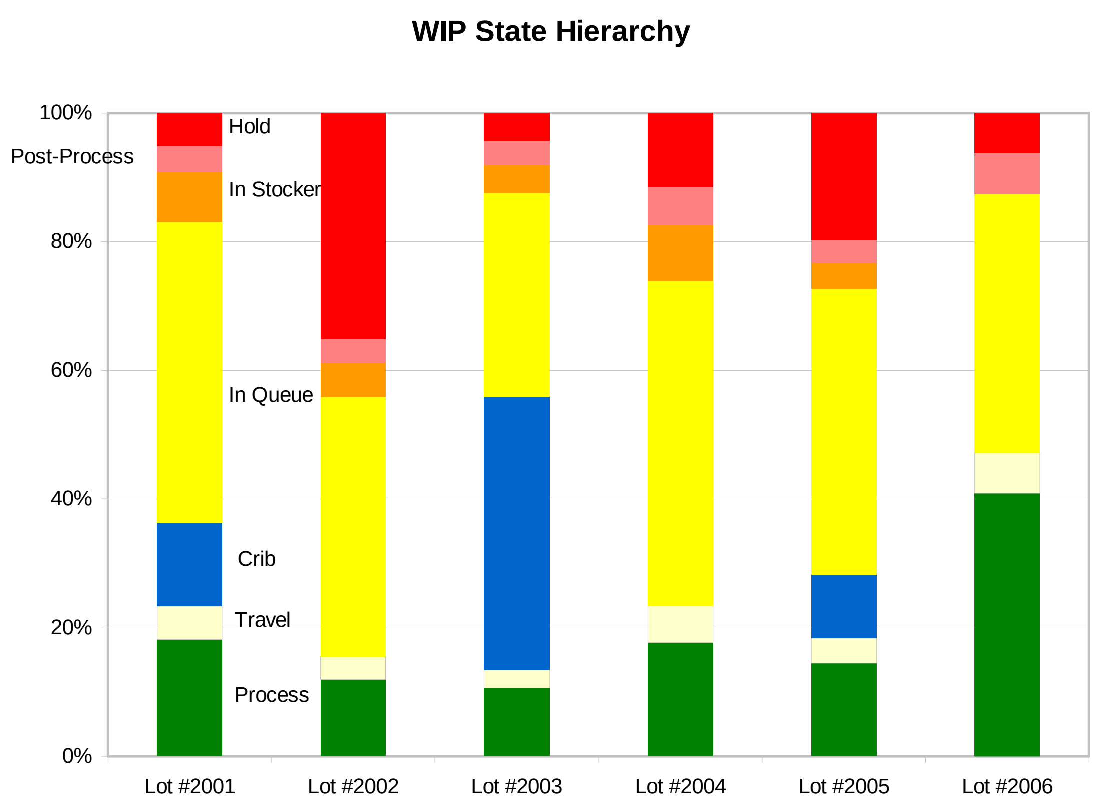
WIP States: A Proposed Hierarchy of the States

As mentioned above, Jennifer presented a talk at the October ISMI Symposium about FabTime's proposal for WIP States. WIP States were introduced in the newsletter in Issue 6.02. The idea behind WIP States is to break up the time that lots spend in the fab into states, and use this information for targeting improvement activities. FabTime proposed 6 initial states: Processing, Post-Processing, In Queue, Traveling, On Hold, and in Crib (extended hold for product mix purposes). We further suggested that only process time (actually, non-rework process time, when lots are being processed at the best theoretical processing rate) should be considered value-added time. After the talk, **Ulrich Dierks of AMD-Dresden** suggested that the WIP States illustrated during the talk should be re-ordered, to indicate a hierarchy. His point was that although the only truly value-added state for WIP is non-rework theoretical process time, some of the remaining WIP States are worse than others. He also suggested that for more automated fabs, it makes sense to break out time in stockers from time in racks. He wrote up the following proposal, and also extended our analogy to the SEMI E10 states for equipment performance tracking.

WIP State	Color	E10 Parallel
Processing	Green	Productive
Travel	White	Standby No WIP
In Queue	Light Yellow	Standby-Other
In Stocker	Dark Yellow	Standby-Other
Post-Process	Light Red	Scheduled Down
On Hold	Dark Red	Unscheduled Down

The idea here is that travel time is necessary to complete processing on a wafer, and so should not be treated as very negative, although it can still be improved in some cases. Post-process time (when a lot has finished being processed at a tool, but has not yet been moved out of the tool, usually due to the lack of an operator) is worse than regular queue time, because both the lot and the tool are held up due to the lack of an operator. We think that Ulrich makes an excellent contribution with regards to this hierarchy, particularly by drawing specific parallels to the SEMI E10 tool states.

We would, in fact, take Ulrich's proposal one step further to add in crib time (or storage time – time that a lot is taken off of its process flow for marketing reasons, to be re-started later). Crib time is analogous to non-scheduled time for tool states. We would color it blue, and place it between travel time and time in queue in the hierarchy. The reason is that crib time is usually outside of the control of manufacturing personnel, and isn't something that they can be working on directly. However, it should not be ignored, because it can have a significant impact on overall lot cycle times, and does contribute to variability in the fab. A graphical example of this WIP State hierarchy is shown below.



Estimating and Using Operation Cycle Times

Introduction

Actual operation-level cycle time is the average time per visit for all lots passing through a particular operation during some time period. For example, the average cycle time for all of the lots that completed operation #1500 in the past 48 hours. This type of actual cycle time data is useful for identifying improvement opportunities and also for generating planned cycle times. Planned operation cycle times are in turn useful for comparing actual performance to planned, as well as for projecting forward to predict lot completion dates. In this article, we will begin by discussing measurement of actual operation cycle times. We will then move on to discuss uses for this type of actual cycle time data. Finally, we will discuss the uses of planned operation cycle time data in more detail.

Measurement of Operation Cycle Times

Most fabs that we work with measure per-visit operation-level cycle times from manufacturing execution system (MES) transactions. Usually they record the operation cycle time as the time from the "move out" of the previous operation until the "move out" of the current operation. If a fab has significant travel and/or stocker time, the travel time may be broken out separately. The latter is probably more common in highly automated fabs. However, at the simplest level, the travel time is treated as part of the operation-level cycle time of the downstream operation.

To be useful for identifying improvement opportunities, the per-visit cycle times must be further broken down into queue time vs. process time. This generally requires the logging of a "begin run" or "start process" transaction. If the "begin run" event is logged, then the time from "begin run" to "move out" is recorded as the process time. The time from "move out" of the previous operation until the "begin run" event is recorded as the queue time.

The process time can be further broken down to further detail in fabs that log an "end run" transaction that is distinct from the "move out" transaction. For example, certain tools can automatically generate an "end run" transaction to indicate when a lot has finished processing. Time can elapse between this "end run" event and the subsequent "move out" transaction. This might occur, for example, because no operator is available to log the "move out". In this case, the time between "end run" and "move out" can be recorded separately as post-process time. (See the subscriber discussion above on WIP States for further thoughts on this, as well as Volume 6, Issue 2 of the newsletter).

What about load and unload times?

The simplest approach to handling load and unload time is to include it as part of the process time. If an engineer is collecting detailed data about how a particular tool operates (e.g. for a staffing simulation model), breaking out the load and unload times separately may be necessary. However, for the purposes discussed in this article (identifying cycle time improvement opportunities, and predicting when lots will complete processing), breaking out load and unload times separately is probably not necessary.

What about setups?

Setups are different from ordinary load times because they are generally not required for every lot. Some percentage of the lots will be ready to process, and then have to wait for a setup. In one sense, this setup time should be treated as process time variability. That is, from the perspective of lot that is at the front of the queue, the setup time looks like part of the process time. However, in practice it is difficult to treat the setup time as part of the process time because the operator does

Sidebar: Exercise for FabTime Software Users

If you have FabTime's software, you can view operation cycle time data yourself.

1. From the Charts page, show "Operation Cycle Time Charts", and click "Go" next to "Operation Cycle Time Pareto" chart.

2. Change the slice control (near the bottom of the main set of filters) to select "Operation" and press "Go".

3. Edit the "From" date for the chart to move it back by a day or two, and note the impact on the "Lots" column in the data table. This tells you how many individual lot cycle times makes up the average value displayed on the chart for each operation. Look for a time window that gives you a reasonable number of lots passing through many of the operations.

4. Find an operation cycle time based on at least 20 lot moves, and click on "List" just below the number of lots. This shows you the queue time and process time for each individual lot move during your time period of interest. How consistent is the data for this operation?

not log the start process transaction until after the setup has been completed. As a result, the setup time will likely appear as part of the queue time for all lots affected instead. Because there are trade-offs between avoiding setups and minimizing cycle time (see newsletter issue 6.07), certain tools may require more in-depth analysis regarding setups.

What about tool qualification time?

Qualification time is generally treated as part of scheduled downtime for a tool. As such, it is not directly part of a lot's process time. It will implicitly show up as part of queue time for any lots that are in queue when a tool qualification occurs. However, it is probably better examined through looking at tool state data, instead of through operation cycle times.

Use of Actual Operation Cycle Time Data

In this section, we will discuss three potential uses of actual operation cycle time data.

Comparing Actual Operation Cycle Times to Planned Cycle Times:

Sorting actual operation cycle time values for some time period in descending order by operation gives an indication of which operations, right now, are contributing the most to cycle time. This data can also be aggregated by tool group, to show which tool groups are contributing the most to cycle time. If we then compare this to the planned cycle time for each operation or tool group, we can highlight places where actual performance is not meeting expectations. Over time, if too many operations do not meet their planned cycle time targets, the fab will start to experience problems with due date performance.

Comparing Actual Operation Cycle Times to Process Times (X-Factors):

If we take actual average operation cycle time and divide by average process time, we get an estimate of operation-level x-factor. This gives us an indication of how we're doing relative to the best that we could be doing for this operation. If the x-factor for a particular operation is much higher than the overall x-factor target for the fab, this operation may be a target for improvement. This data is probably best aggregated by tool group, so that the particular aspects of the tool group can also be taken into account. For example, lots passing through the bottleneck toolgroup may be expected to have a higher operation-level x-factor than lots passing through other groups. Similarly, lots passing through one-of-a-kind tool groups will most likely have higher x-factors than lots passing through tool groups that have more redundancy. Because batch tools have such long process times, they are often expected to run at lower x-factors than other tools. Despite these exceptions, most fabs have a general expectation of operation-level x-factor performance, and can benefit from analyzing operations or tool groups that do not meet this target.

Using Actual Cycle Time Data to Generate Planned Numbers:

Although planned operation cycle time data can come from other sources (e.g. process requirements), many fabs use actual cycle time data to estimate or update planned values. This can be accomplished using two primary methods. The first involves simply averaging (or taking the median of) past operation-level cycle times, perhaps decreasing them by some percentage improvement target, and using them as planned values going forward. The second method, and the one that we recommend, involves using actual process time data to estimate theoretical process times, and then inflating these by some target x-factor to get to the planned operation cycle times. Because this a more granular approach than simply averaging the overall operation cycle times, it allows for more finely targeted improvement efforts. For instance, different types of tools can have different planned x-factors, each specific to the characteristics of the tool type. Also, improvement efforts can be phased in, with aggressive targets for some sub-set of the tools put in place first, and then rolled out to other tools as needed.

There are a number of specific implementation issues to be resolved regarding the generation of planned values from actual process times.

- What time window do we use for tracking actual data and aggregating it to estimate theoretical/planned data?
- When we aggregate, do we use the average process time, or do we use the median?
- What do we do for operations for which we don't have enough data points to form a reasonable estimate? How many data points are "enough"? How do we know that we have a "reasonable" estimate? If we don't have enough points, do we aggregate across other operations run on the same tool? Or across similar steps on other process flows?
- What level of detail should we use to store this cycle time information? Do we have separate estimates for each route/operation combination? Or do we use an average for each operation across the different routes? Do we store tool-specific operation cycle time data, or store an average across all of the qualified tools for a step?
- How frequently should we update planned process time data? Weekly? Quarterly? Annually?

There are no absolute answers to any of these questions. The responses will vary according to how a company intends to use the data, how detailed their actual data is in the first place, and what resources and systems they have in place to automate the calculations. We welcome subscriber feedback regarding how individual fabs have chosen to address these issues. Your response can be attributed to you, or can be anonymous if you prefer.

Use of Planned Operation Cycle Time Data

In this section, we will discuss three potential uses for planned operation cycle time data.

Using Planned Operation Cycle Time Data to Estimate Lot Completion Dates:

Planned operation-level cycle time data can be used to project individual lots forward and predict an expected completion date for each lot. This allows us to identify lots that are at risk of missing their due dates. It also allows us to predict which lots will ship during a particular time frame. This can be valuable information for planning overall shipments, and for predicting workload in back-end manufacturing areas.

Using Planned Process Time Data for Dispatching:

Dispatching decisions sometimes require knowledge of planned process times. For example, if we are trying to avoid starvation of a downstream tool through dispatching decisions on an upstream tool, we need to be able to predict when the tool will finish processing its current WIP load. We also need to be able to predict what the process time will be for each lot at the current tool, to know how soon we can get lots through to the downstream tool. Similarly, we may require planned process time data to make batching decisions (should we wait for a lot due to arrive soon, or start processing now?) or decisions about setups or tool dedication.

Using Planned Operation Cycle Time Data to Compare with Actual Performance

As discussed above, planned cycle time data can be used to compare with actual operation cycle time data. The idea here is to identify tools or operations where the plan is not being met, and then investigate further to understand why. Has there been a process change that is not being reflected in the planned cycle time data? Was the planned data generated during some atypical situation, such as a particularly bad tool downtime, such that it is not really representative of the current performance?

Conclusions

Operation cycle time data is relatively straightforward to estimate, and has a variety of uses. Actual values can be used to flag operations for which the ratio of cycle time to process time (x-factor) is higher than expected. Actual values can be compared with planned values, to identify short-term problems. Actual operation cycle times are also helpful in calculating planned cycle times. Planned cycle time values are in turn useful for estimating lot completion dates and facilitating dispatching decisions. Because cycle time can be improved by reducing queue time anywhere in the line (not just through improvements at the bottleneck), analysis of operation-level cycle times is a powerful cycle time improvement technique.

Closing Questions for FabTime Subscribers

Do you use actual operation cycle time data to identify cycle time problems, or to estimate planned numbers? How have you dealt with the implementation issues discussed above? If you use x-factors applied to process time values, do you use the same x-factor for all operations, or do you vary the x-factor according to tool-specific attributes?

Subscriber List

Total number of subscribers: 1925, from 431 companies and universities. 23 consultants.

Top 10 subscribing companies:

- Intel Corporation (102)
- Analog Devices (78)
- Infineon Technologies (64)
- STMicroelectronics (61)
- Atmel Corporation (59)
- Freescale Semiconductor (56)
- Micron Technology (52)
- Philips (47)
- Texas Instruments (44)
- TECH Semiconductor (40)

Top 3 subscribing universities:

- Virginia Tech (10)
- Arizona State University (8)
- University of California – Berkeley (7)

New companies and universities this month:

- Central Michigan University
- COVEGA
- Integrated Micro Technology (IMT)
- iSky Factory Automation
- Lumileds Lighting LLC
- LuminentOIC
- MIRC Electronics Ltd.
- Neuberger Berman
- Toyo Tanso USA Inc.
- University of North London

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® Software Capacity Planning Module

Installation

For a fixed price, FabTime will:

- Identify the source of any additional data needed for the planning module.
- Automate the process of importing the additional data into FabTime.
- Validate against client data.

Interested?

Contact FabTime for more information, or for a quote.

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

Do you need to answer questions like:

- Given a target product mix, do we need any new tools?
- Given the tools that we have, and the products that we are running, how many wafers can we expect to produce?
- Given our existing set of products and tools, what happens if the product mix changes? Where can we expect bottlenecks?

Are you tired of maintaining a standalone capacity planning spreadsheet?

FabTime's capacity planning module leverages the data already stored in the FabTime digital dashboard software, to make it easier to build capacity planning scenarios. The only required manual inputs are:

- Weekly ships per product.
- Product line yield percentages.

FabTime uses route information from the fab MES and calculates UPH data (tool speed) based on actual performance. FabTime also uses tool uptime performance to estimate availability (though this can be overridden). These inputs are used to generate predicted utilization percentages for each capacity type. Detailed intermediate calculations (UPH, tool productive time, tool rework percentage, etc.) are also available (an example for one tool is shown below). All outputs can be easily exported to Excel.

Capacity Planning Module Benefits

- Eliminate the need to maintain offline capacity planning models.
- Automatically update capacity planning data to reflect new conditions (process flows, tool uptime characteristics).
- Quickly run scenarios to anticipate (and avoid) bottlenecks caused by product mix changes.

C Type	Output	Value	Notes
1XStep	Rework Moves/Week	21	2004-09-06 10:00:00 to 2004-11-15 10:00:00
1XStep	Total Moves/Week	12310	2004-09-06 10:00:00 to 2004-11-15 10:00:00
1XStep	Rework Ratio	0	Rework Ratio = Rework Moves / Total Moves.
1XStep	Productive%	61	2004-09-06 10:00:00 to 2004-11-15 10:00:00
1XStep	Availability%	76.26	Availability = Productive% + Standby%.
1XStep	Historic Utilization%	79.99	Utilization (Mfg efficiency) = Productive% / Availability%.
1XStep	Productive(Rework)%	0.1	Productive(Rework)=Productive% * ReworkRatio.
1XStep	Net Availability%	76.15	Net availability% = Availability% - Productive(Rework)%.
1XStep	Arrivals (Units/Hour)	79.36	Based on total plan WGR=2025
1XStep	Tool Quantity	8	1XStep#1 ... 1XStep#8
1XStep	UPH	15.02	UPH = (TotalMoves/ToolQty) / (Productive% * 168)
1XStep	Required Hours/Day	126.84	Required hours = 24 * HourlyArrivalRate / UPH
1XStep	Predicted Utilization%	86.75	Util = 100 * ReqdHours / (24 * NetAvail * ToolQty / 100)
1XStep	Max WGR	2334.22	MaxWGR = PlanWGR / PredictedUtilization
1XStep	Historic WGR	2457.8	(Non Rework Moves) / (OperationCount / ProductCount).