FabTime Cycle Time Management Newsletter

Volume 2, No 1, January 2001

Information

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

Publisher: FabTime Inc.

Editor: Jennifer Robinson

Table of Contents

Welcome

Welcome to Volume 2, Issue 1 of the FabTime Cycle Time Management Newsletter. In this issue, we have no responses to last month's newsletter topic (it's not a good time of year for things like that), but we do have a bunch of recommendations and resources.

New on FabTime's website is a directory of conferences and trade shows that we think will be of interest to people who work in wafer fabs (www.fabtime.com/confs.htm). We've included conference dates and locations, as well as paper deadlines (some of you know that we frequently encourage our friends to write more papers). If you know of a conference or trade show that you think should be added, please email your suggestion to Jennifer.Robinson@FabTime.com.

The new topic for this issue is the impact of batch size decision rules on cycle time. Batch tools are frequently subject to conflicting interests. On the one hand, we want to run them as full as possible, to minimize wasted capacity from half-empty batches. On the other hand, we want to minimize the contribution of the batch tools to cycle time. And as you'll probably expect if you've been reading previous newsletter issues, these two goals often conflict. In this issue we'll review why that is, and present some guidelines for balancing objectives at batch equipment.

Thanks for reading! -- Jennifer

FabTime

325M Sharon Park Dr. #219
Menlo Park CA 94025
Tel: 408 549 9932
Fax: 408 549 9941
www.FabTime.com

Impact of Batch Size Decisions on Cycle Time

Background

Batch tools are tools in which more than one lot may be processed at one time. They are generally used for long operations, such as furnace bake operations. For example, a typical batch furnace might be able to process up to eight lots at one time, and have a process time of up to twelve hours. Processing time is usually independent of the number of lots in a batch, and once a batch process begins, it cannot be interrupted to allow other lots to join.

From a local perspective, when a furnace is available and full loads are waiting, the decision to process a batch is obvious, since no advantage can be gained at that work area by waiting (although a decision may still be needed concerning which product type to process). However, when there is a furnace available and only partial loads of products are waiting, a decision must be made to either start a (partial) batch or wait for more products to arrive.

There are two problems with running a partial batch. One is that the unused capacity of the furnace will be "wasted." The other problem is that lots that arrive immediately after the batch starts cannot be added to the batch, and might have to wait many hours until another furnace is available.

There are also problems that stem from waiting to form a full batch. The lots that are waiting to be processed incur extra queue time while waiting for other lots to arrive. The furnace is held idle, driving down its efficiency. And full batches contribute more to variability after the furnace operation.

"Look-ahead policies are naturally dependent on the accuracy of the information concerning future arrivals."

Batch Size Decision Policies

There are two basic types of batch size decision policies. The first type are known as Minimum Batch Size (MBS) decision rules, or threshold policies. An MBS rule simply states that, whenever there are N lots in queue, ready to form a batch, and a furnace is available, the operator should immediately start processing those N lots. Here N could be any value from one up to the maximum load size for the furnace. An MBS rule with a load size of one is sometimes referred to as a "greedy" policy, while one with the maximum load size is called a "force-full" policy (since the furnace is only run when it is as full as possible). The latter policy is also sometimes referred to as a "near-full" policy, since it's not always possible to completely fill a batch (due to yield loss or lot/batch size mis-match), and in practice the batch is filled as near to full as practical.

The other category of batch size decision rules are known as "look-ahead" rules. With a look-ahead rule, the furnace operator looks ahead in some way to see which lots are expected to arrive soon, and sometimes waits to form the batch until additional lots arrive. Different methodologies are used to decide when to wait, but the general idea is to minimize the sum of the expected waiting time for lots already in queue and lots expected to arrive within some time window. Look-ahead policies are naturally dependent on the accuracy of the information concerning future arrivals, and require the presence of some sort of predictive control system. For the remainder of this article, we will focus mainly on threshold policies, rather than look-ahead policies.

We would like to note here a couple of specific situations in which look-ahead policies may be necessary. First, if you have a batch tool that has many different recipes that can't be batched together (e.g. all step-specific batching), it can be difficult to fill batches at all, because the likelihood of getting a full batch into the queue at once is very small. Here a full-batch policy is terrible, because there is so much waiting, but a greedy policy may lead to unacceptably small batches. This situation can be compounded in fabs that attempt to improve cycle times by running small lot sizes. If the maximum batch size is much greater than the average lot size (e.g. you run 4-wafer lots, and have a 200-wafer capacity batch tool), you may need to wait to group some number of lots together, even if not a full batch. Otherwise, the tool will always run nearly empty, and its practical capacity will be very low. In these situations, a look-ahead policy of some sort may be necessary.

Minimum Batch Size Rules

MBS rules are easier to implement than look-ahead rules. We simply select a threshold, N, and form a batch whenever N or more lots (of the same type) are ready to be processed. If more lots are available than the capacity of the furnace, a first-in-first-out rule is usually used to select between then. The difficulty with MBS rules lies in selecting the threshold, N. Do we set N high, to minimize the amount of unused space in process batches? Or do we set N low, to minimize the queue time of lots that are already waiting? It turns out that the answer depends on how highly loaded the furnace is. If we have a furnace with a very low utilization, and we always wait to process full batches, we will artificially inflate lot cycle times. Here's a simple example:

Full-Batch Policy

Suppose we have a single furnace, that can process up to eight lots at one time, and has an eight-hour process time (constant). If a single lot arrives every two hours (constant time between lot arrivals), then on average the furnace will be loaded to 50% of its capacity (since it can process eight lots every eight hours, but only four lots arrive every eight hours). Suppose that the furnace has just started processing a batch, and call this time zero. Let's look at what happens when we wait to form full batches. We won't be able to start another batch until eight lots have arrived, 16 hours from now. We get the following pattern of arrival, start process, and queue times, where queue time is simply Start Time - Arrival Time:

Lot	Arrival Time	Start Time	Queue Time
#1	2	16	14
#2	4	16	12
#3	6	16	10
#4	8	16	8
#5	10	16	6
#6	12	16	4
#7	14	16	2
#8	16	16	0

The average queue time is (14+12+10+8+6+4+2)/8 = 56/8 = 7 hours. Because everything is constant in this example, the entire pattern just repeats, thus the average queue time across all lots for the full batch scenario is eight hours, nearly equal to the raw process time of eight hours.

Greedy-Policy

Now suppose that instead of forming full batches, we use a greedy policy and start processing a batch whenever the furnace is free, and lots are available. In this case, we'll start a new batch every eight hours (every time the furnace becomes free). Starting with the same starting point as previously, we have the following pattern:

Lot	Arrival Time	Start Time	Queue Time
#1	2	8	6
#2	4	8	4
#3	6	8	2
#4	8	8	0
#5	10	16	6
#6	12	16	4
#7	14	16	2
#8	16	16	0

Now the average queue time is (6+4+2+0+6+4+2+0)/8 = 24/8 = 3 hours. We've eliminated four hours of queue time, on average, for all lots, by not forcing a low-utilization furnace to be loaded to full all the time.

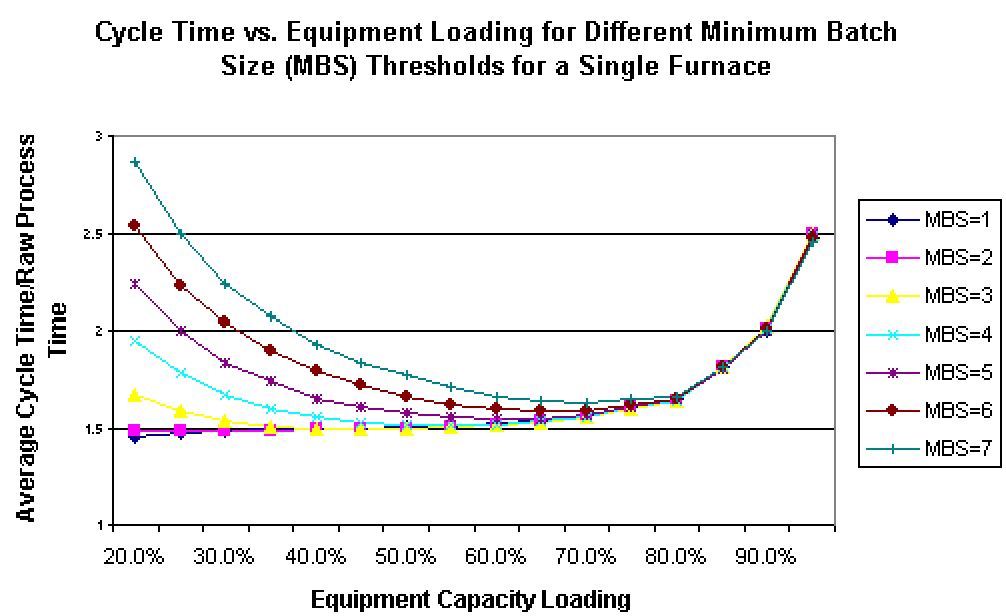
Single-Tool Simulation Results

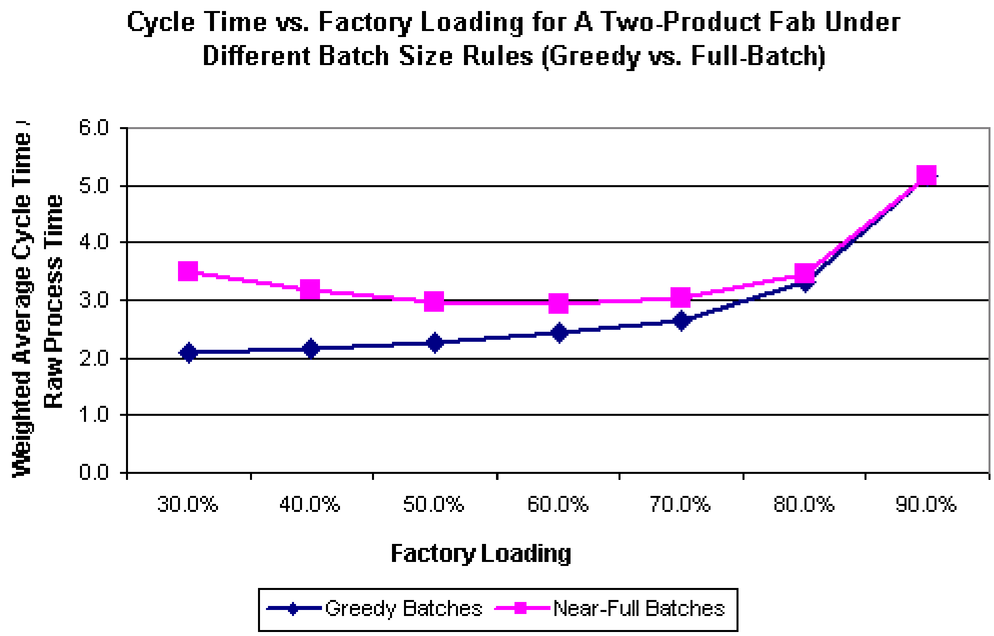
Obviously, the above example is unrealistic -- with constant process and interarrival times. We ran a series of simulation models of this system, with highly variable times between arrivals (exponential). We varied the interarrival time, to look at the interaction between furnace utilization and minimum batch size, and also varied the threshold value. The results for minimum batch sizes of 1 (greedy) to 7 (nearly full) are shown in the chart below. We ran each simulation for 5 years, and only include results from a single replication for illustration. The numbers plotted are system cycle time divided by raw process time.

Here we see that until the furnace is loaded to about 90%, a greedy (minimum batch size of one) policy results in lower cycle times than a full-batch policy. At high utilizations there is a very slight improvement from using a near full-batch policy over a greedy policy. This is consistent with other research in this area.

Full Fab Model Simulation Results

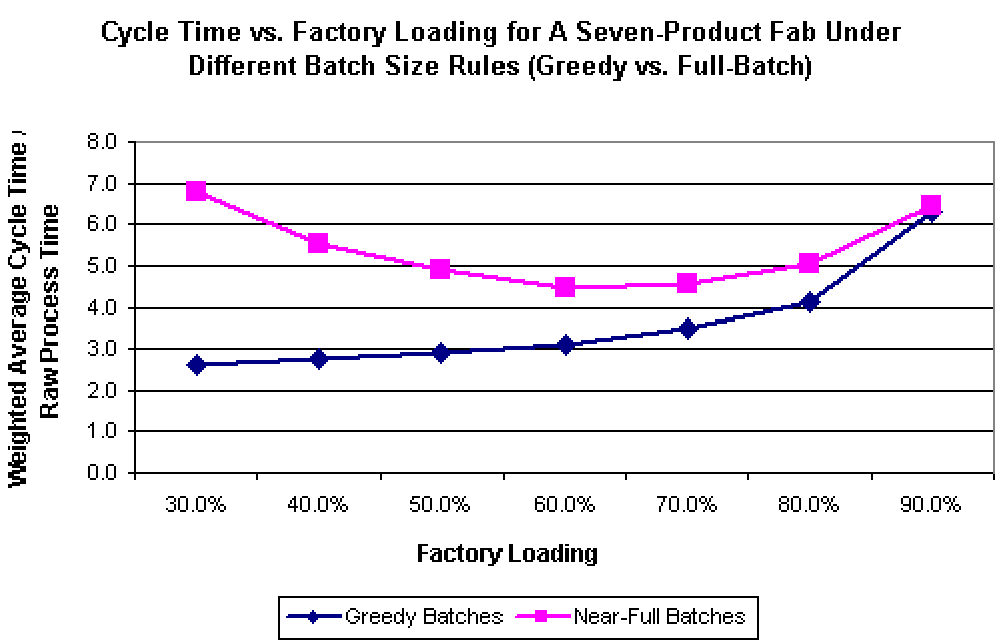
Now, you might wonder if this has any effect on the factory as a whole. After all, an extra few hours here or there on the furnaces could be lost in the noise relative to the overall cycle time. We therefore did another experiment using a simplified version of a full factory model. The model had two products, 115 steps per process flow, 22 tool groups, and 21 operator groups. We simulated this model for two years, varying the start rate to allow different levels of bottleneck utilization for each run. The results are on the following page.





In the full factory model, the average cycle time is almost 70% greater for the full-batch policy than for the greedy policy at very low utilizations. Up to 80% loading, the greedy batch policy yields lower cycle times. For very highly loaded fabs the full-batch policy yields essentially the same results as the greedy policy.

For a more extreme example of the impact of batching on this fab, we modified the factory to have a greater number of products. We held the total volume the same, but divided it among seven products instead of two. All products used the same process flow, but for certain batch tools in the model, lots of different product types could not be batched together. This change thus increased the volume of distinct batch IDs in the model. The change led to a slight degradation in performance under the greedy policy, and to a significant cycle time increase under a full-batch policy, as follows:



Clearly, batching policy makes a big difference in this high-product mix fab, because there are so many distinct batch IDs. Lots almost always wait a long time to form a batch under a full-batch policy, especially for very low utilizations. The increase in cycle time between this case and the previous case also illustrates how sensitive fab models can be to batching rules (in this case, decisions about which types of lots can be batched together).

Single-Sentence Summary

For batch tools that are not highly loaded, setting a high threshold for a minimum batch size decision rule (forcing full or near-full batches) can significantly increase local cycle times, as well as overall fab cycle times.

Further Explanation

Applying a full-batch policy to a tool that is not highly loaded can significantly degrade performance. Applying a greedy-policy to a highly loaded tool, on the other hand, has at worst a slight negative effect in most cases (with the exceptions noted above for very small lot sizes and/or abundant recipes). The reason for this is simple. Suppose you have a greedy rule, and you start a batch with only a single lot in it, even though the furnace is in general highly loaded. By the time the batch finishes, and the tool is ready to process another batch, there will probably be a whole bunch of lots waiting in queue, so that the next batch will be large. Remember that a greedy policy just means that the operator will start processing if only one lot is there, but if many lots are there ready to be processed, then the operator will take as many as can fit in the furnace.

One other point is that a full-batch policy contributes more variability to the fab than a greedy policy, and this is part of what drives up overall fab cycle times. For example, suppose you have the batch tool described in the example above (capacity of eight lots, eight hour process time), and the batch step is followed by an inspect step that takes place on one lot at a time. Now suppose that the process time on the inspect step is 15 minutes. The inspect tool can process four lots per hour, compared with the batch tool, which processes eight lots in eight hours, or one lot per hour. Clearly, the inspect step is not a bottleneck for the sequence. However, when you dump a batch of eight lots in front of the inspect step after finishing at the furnace, most of them will have to wait, because the inspect tool can only process one lot at a time. One of the lots waits 15 minutes, the next waits 30 minutes, etc., until the last lot waits for an hour and forty-five minutes, or seven times the raw process time of the inspect step. In practice, a fab will have multiple identical inspect tools, mitigating the effect of this situation, but you're still going to be better off in terms of overall cycle times with smaller, more frequently released batches.

Conclusion

There are sometimes process or cost reasons to attempt to run full batches in a wafer fab. However, artificially imposing mimimum high batch size policies for tools that are not heavily loaded can significantly increase overall fab cycle times. For most tools, a greedy batching policy works well at lower utilizations, and still performs acceptably at higher loadings (when the batches become full by default). This rule is easy to apply, and fairly robust to changes in product mix. In a few situations, however, such as when there are many distinct batch IDs, or there is a mismatch of some sort between lot size and maximum batch size, something smarter than a simple threshold policy is necessary. Here the application of look-ahead rules can improve factory performance.

Additional References

For a much more detailed discussion on batch size decision rules, see "A review of real-time control strategies for furnace batch sizing in semiconductor manufacturing" by J. K. Robinson, J. W. Fowler, and J. F. Bard, available from www.fabtime.com / abs_MfgRev.htm.

Other articles that discuss batch size decisions in wafer fabs include:

- R. Deb and R. F. Serfozo, "Optimal control of batch service queues," *Advances in Applied Probability*, vol. 5, no. 2, pp. 340-361, 1973.
- J. W. Fowler, D. T. Phillips, and G. L. Hogg, "Real time control of multiproduct bulk service semiconductor manufacturing processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 5, no. 2, pp. 158-163, 1992.
- C. R. Glassey and W. W. Weng, "Dynamic batching heuristic for simultaneous processing," *IEEE Transactions on Semiconductor Manufacturing*, vol. 14, no. 2, pp. 77-82, 1991.
- H. Gurnani, R. Anupindi, and R. Akella, "Control of batch processing systems in semiconductor wafer fabrication facilities," *IEEE Transactions on Semiconductor Manufacturing*, vol. 5, no. 4, pp. 319-328, 1992.
- M. F. Neuts and R. Nadarajan, "A multiserver queue with thresholds for the acceptance of customers into service," *Operations Research,* vol. 30, no. 5, pp. 948-960, 1982.
- W. B. Powell and P. Humblet, "The bulk service queue with a general control strategy: theoretical analysis and a new computational procedure," *Operations Research*, vol. 34, no. 2, pp. 267-275, 1986.
- J. K. Robinson, J. W. Fowler, and J. F. Bard, "The use of upstream and downstream information in scheduling semiconductor batch operations," *International Journal of Production Research*, vol. 33, no. 7, 1849-1870, 1995. (abstract at http://www.fabtime.com/abs_IJPR.htm).
- W. W. Weng and R. C. Leachman, "An improved methodology for real-time production decisions at batch-process work stations," *IEEE Transactions on Semiconductor Manufacturing*, 1993.

Community News / Announcements

Winter Simulation Conference

I attended the Winter Simulation Conference in Orlando last month. As in the past few years, I found lots of good papers on the use of simulation in semiconductor manufacturing. You can download these papers from www.informs-cs.org (www.informs-cs.org/wsc00papers/prog00.htm#SE to go directly to the semiconductor track for this year). I was only sorry that I had to cut my trip short, and didn't have time to spend with all of the people I would have liked to see. Maybe next year... the conference will be in Alexandria, Virginia, and contributed papers are due in April.

FabTime welcomes the opportunity to publish announcements for individuals or companies. Simply send them to Jennifer.Robinson@FabTime.com.

FabTime Recommendations

■ The 2000 Update for the International Technology Roadmap for Semiconductors (ITRS) was released to the public in December. You can download the associated PDF files from the ITRS website, at http://public.itrs.net/Home.htm. The roadmap is an assessment of the semiconductor industry's technology requirements, to ensure performance advances. This assessment, called roadmapping, is a cooperative effort of the global industry manufacturers and suppliers, government organizations, consortia, and universities, with communication organized by SEMATECH.

■ The high tech industry is filled with acronyms. If you come across an acronym that you don't recognize, we suggest trying the Acronym Finder website (www.acronymfinder.com). It includes multiple definitions for each acronym, with the most common definitions listed first. The site appears to be advertising sponsored, and thus has many annoying blinking ads, but it's quite useful.

■ FabTime's Book of the Month for January is "Faster: The Acceleration of Just About Everything," by James Gleick. You can find this review at www.fabtime.com/faster.htm. The book also has a website at www.fasterbook.com. The website includes excepts and links for each chapter, as well as links to the author's webpage, and to the US Naval Observatory's master clock time, tycho.usno.navy.mil/cgi-bin/timer.pl.

■ The January 7th issue of the Journal of Production Economics (Volume 69, No. 1) appears to be focused on measuring productive efficiency. I have abstracts, if anyone is interested, or you can view them at www.elsevier.nl/locate/issn/09255273.

Subscriber List

Total Subscribers: 281

Advanced Energy Industries (1)
Advanced Micro Devices (11)
Agere (formerly Lucent) (3)
Agilent Technologies (2)
Alpha Industries (1)
Amkor (2)
Analog Devices (2)
Applied Materials Corporation (6)
Arizona State University (3)
Artest Corporation (1)
Atlantic Technologies Ltd. (1)
AT & S India Limited (1)
BP Solarex (3)
Canon USA (1)
Carsem M Sdn Bhd (1)
Chartered Semiconductor Mfg (3)
Clarkson University (1)
Cofer Corporation (1)
Compugraphics International Ltd. (1)
Conexant Systems, Inc. (2)
Cronos Integrated Microsystems (1)
Dallas Semiconductor (4)
Dick Williams and Associates (1)
DomainLogix Corporation (1)
Durham ATS Group (2)
E-Tek Dynamics (2)
e-METS Co, Ltd (1)
FabTime (2)
Gintic Institute of Mfg. Technology (1)
Headway Technologies (5)
Hewlett-Packard Company (2)
Hyundai Semiconductor America (2)
IBM (4)

Infineon Technologies (19)
Intarsia Corporation (2)
Integrated Technologies Company (2)
Intel Corporation (16)
IRIS Technologies (1)
James Nagel Associates (1)
Kansas State University (1)
Ken Rich Associates (1)
Lexmark International, Inc. (1)
Lockheed Martin Fairchild Systems (1)
LSI Logic (6)
Lucent Technologies (1)
Macronix International Co. (1)
Maxim Integrated Products, Inc (1)
Metrology Perspectives Group (1)
Micrel Semiconductor (1)
MicroVision-Engineering GmbH (1)
Mitel Semiconductor (8)
Motorola Corporation (28)
MTE Associates (1)
Multimedia University (1)
Nanyang Technological University (1)
National Semiconductor (11)
NEC Electronics (6)
Nortel Networks (4)
Oklahoma State University (1)
ON Semiconductor (8)
Penn State University (1)
Philips Semiconductors (2)
Powerex, Inc. (2)
PRI Automation (1)
Productivity Partners Ltd (1)
Raytheon (1)
Read-Rite Corporation (2)
RTRON Corporation (2)
SAMES (1)
Samsung Semiconductor (2)
Seagate Technology (8)
SEMATECH (10)
Semiconductor Research Corp. (SRC) (1)
SEZ America, Inc. (1)
Solectron Corporation (1)
SSMC (1)
STMicroelectronics (8)
SVG (1)
Synergistic Applications, Inc. (1)
Synquest (3)
Takvorian Consulting (1)
TDK (1)
TECH Semiconductors (2)
Texas A&M University (1)
Texas Instruments (5)
United Microelectronics Corp (1)
University of Arkansas (1)
University of California - Berkeley (2)
University of Maryland (1)
University Porto (Portugal) (1)
University of Virginia (1)
University of Wuerzburg (Germany) (2)
Wacker Siltronic AG (2)
White Oak Semiconductor (2)
Unlisted Companies (3)

Independent Consultants:
Stuart Carr
Alison Cohen
Ted Forsman
Dan Theodore
Craig Volonoski

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. To protect the privacy of our subscribers, email addresses are not printed in the newsletter. If you wish contact the subscribers from a particular company directly, simply email your request to the editor at Jennifer.Robinson@FabTime.com. To subscribe to the newsletter, send email to the same address. We will not, under any circumstances, give your personal information to anyone outside of FabTime.