Proceedings of the 2002 Winter Simulation Conference
E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, eds

SIMULATION-BASED ANALYSIS OF A COMPLEX PRINTED CIRCUIT BOARD TESTING PROCESS

Jeffrey S. Smith
Yali Li

Industrial & Systems Engineering Department
Auburn University
Auburn University, AL 36849, U.S.A.

Jason Gjesvold

Soldering Technology International, Inc.
102 Tribble Drive
Madison, AL 35758, U.S.A.

ABSTRACT

This paper describes a simulation-based analysis of a printed circuit board (PCB) testing process. The PCBs are used in a defense application and the testing process is fairly complex. Boards are mounted on a test unit in batches and go through three thermal test cycles. As boards fail testing during the thermal cycling, operators can either replace the failed boards at fixed points during the cycling or can allow the test unit to complete the testing cycle before removing failed boards. The primary objective of the simulation study is to select an operating strategy for a given set of operating parameters. A secondary objective is to identify the operating factors to which the strategy selection is sensitive. Initial testing indicated that failed boards should be replaced as soon as possible under the current operating configuration of the sponsor's facility. Secondary testing is also described.

1 INTRODUCTION

This paper describes a simulation-based analysis of a thermal testing process for printed circuit boards (PCB). The testing described here is the final stage of an assembly process for a defense-oriented product. This analysis was sponsored by the product manufacturer (a defense contractor) who was interested in the specific operating policy for the test process. In particular, the sponsoring company was interested in whether failed boards should be removed from the testing cycle as soon as possible after they fail or whether they should only be removed upon completion of the entire testing cycle. Simulation-based analysis showed that under current operating conditions, boards should be replaced as soon as possible after they fail. Additional testing indicated that this policy is preferable under a wide range of operating conditions around the current configuration. Finally two contrived cases where the replacement policy is not preferable are described.

This paper is organized as follows. Section 2 describes the board testing procedure and the alternative operating policies under consideration. Section 3 describes the simulation models developed for the project. Section 4 describes the initial testing and presents the results of the analysis. Section 5 describes the additional testing done to identify configurations in which the non-replace strategy is preferable. Finally, Section 6 presents the conclusions.

2 BOARD TESTING PROCEDURE

Individual boards go through seven individual tests spread over three thermal cycles as illustrated in Figure 1. between tests, the boards are iteratively ramped up to +40°*C* and down to -40°*C*. In order to prevent thermal shock to the boards, the ramp time is carefully controlled and can take a significant amount of time. The order of the thermal cycles that a board goes through is not important. That is, while boards must go through all three cycles, it does not matter whether they start in Cycle 1, Cycle 2, or Cycle 3. Considering continuous testing, boards can start at r_0, r_1, or r_2 and are finished after completing all three cycles. The only requirements are that test t_0 be done before starting the thermal cycling and that test t_7 be done when the final cycle is completed.

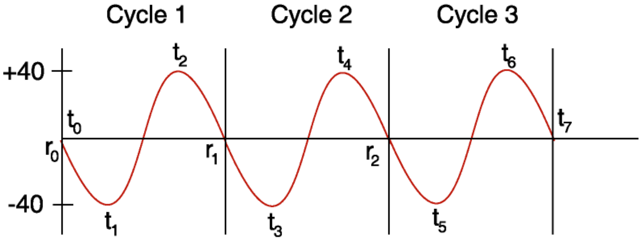

Figure 1: PCB Testing Procedure

Each test has an associated duration and a probability of failure (the current configuration values are shown in Table 1). Boards that fail any test are sent to a rework station and must repeat all testing regardless of where the failure occurs in the testing process (from a modeling perspective, reworked boards are treated as new arrivals). All of the tests are computer-controlled with deterministic processing times. However, while the tests themselves are automated, an operator must manually start each test and accept the results upon completion of each test. If an operator is not immediately available to start a test or accept the test results, the board waits at its current temperature until an operator is available. The operators have additional responsibilities in other areas of the system (not included in the model), so a stochastic delay is included before the operator is available to account for the operators' travel time.

Table 1: Test Durations and Failure Probabilities

Test (i)	Duration (d_i) (minutes)	Failure prob. (F_i)
0	20	0.05
1	20	0.07
2	20	0.03
3	5	0.07
4	5	0.05
5	15	0.08
6	5	0.01
7	20	0.05

Individual boards must be mounted on *test units* to go through the thermal cycling and testing. In the current configuration, each test unit has a capacity of three (3) boards. An operator is required to load boards onto the test unit prior to the start of testing and to unload boards from the test units upon completion of the testing (either after three complete cycles or when a board fails one of the tests and is going to be replaced). Considering the thermal cycling illustrated in Figure 1, it is actually the test unit as a whole that goes through the cycling. Individual boards can be mounted and removed only at the end of a cycle (r_0, r_1, and r_2 in Figure 1). Unless there is a board to be removed or added, the test unit does not have to stop at ambient temperature while ramping down.

The use of multi-board test units is one of the major complications in the analysis of this system. More specifically, the sponsoring company is interested in determining whether and when failed boards should be *replaced* during the thermal cycling/testing process. That is, when an individual board on a test unit fails a test, replacing that board will delay the other two boards on the test unit from completing the next test cycle. Conceptually, the tradeoff is fairly straightforward. Allowing failed boards to remain on the test unit wastes test unit and thermal chamber capacity whereas removing a failed board delays the completion of the other boards mounted on the test unit. More specifically, to remove a failed board the test unit must be stopped at ambient temperature while coming down from +40°*C* before the operator can remove the failed board and replace it with a new board. In addition, the new board must go through test t_0 once it has been mounted. The delay time for the test unit (and the other mounted boards) clearly depends on the operator availability, the board loading and unloading time, and the duration and results of test t_0.

The primary objective of the project was to evaluate the current system configuration under the following two operational policies and to determine which policy is preferable:

Policy 1: *Replacement* – Operators replace failed boards as soon as possible after they fail any test.

Policy 2: *No replacement* – Failed board are not replaced until the end of the third thermal cycle.

The primary performance metric of interest is the completion time for a batch of boards of a given size (i.e., the makespan for the batch).

The secondary objective involved testing the sensitivity of the policy selection to three configuration parameters. The results (described in Section 4) indicate that the replacement policy (Policy 1) is generally better under the tested conditions. Based on these initial findings, the final objective was to identify a set of configurations under which the no-replacement policy (Policy 2) was the better alternative.

3 SIMULATION MODEL

The initial process-oriented model for this project was developed using the SIMAN simulation language (Pegden *et al.*, 1995). In this model, the testing process component was actually one part of a model of a larger assembly operation. While the no replace policy model was fairly straightforward, the replace policy was significantly more complicated. In the assembly operation model, boards were treated individually and, hence the thermal testing part of the code required fairly complicated grouping logic and significant conditional branching logic in order to model the use of test units in conjunction with the replacement policy. In addition to the model complexity, the original model was fairly inflexible. This SIMAN model was validated, used for the analysis of the current configuration, and delivered to the sponsoring company.

A second model was developed to perform the secondary testing of just the thermal testing process. In order to maximize the model flexibility, the second model was developed as an event-oriented model in C++. The event-oriented model is loosely based on the C++ code presented by Banks *et al.* (2001). The general results from the initial SIMAN model agreed with the results from the C++ model. All of the results presented here are based on the event-oriented C++ model.

The event-oriented model is comprised of ten events and the associated event-handling logic along with the initialization and reporting routines. The events and event logic include:

1. Arrival of the batch of boards – Available test units are each assigned boards and events corresponding to operators completing the initial test setup are scheduled.
2. Operator completes a test setup – Released operator checks the board queue for boards waiting to be unloaded (either completed boards or failed boards) and/or for tests ready to be started. If boards are waiting to be unloaded or tests are waiting to be started, the corresponding completion event is generated and scheduled. The test completion event t_0-t_7 is also generated and scheduled.
3. Test unit completes test t_0 for a board – For Policy 1, check the test status and if a board failed, request an operator to replace the board and perform the computer setup for test t_0. If an operator is not available, get in the queue. Otherwise schedule the event of the test unit reaching -40°*C*.
4. Test unit reaches -40°*C* – If an operator is available, schedule the completion event for test setup. If an operator is not available, place the unit in the operator queue.
5. Test unit completes either test t_1, t_3, or t_5 (the tests at -40°*C*) – Update the states of the boards and generate the event of the test unit reaching +40°*C*.
6. Test unit reaches +40°*C* – If an operator is available, schedule the completion event for test setup. If an operator is not available, place the unit in the operator queue.
7. Test unit completes either t_2, t_4, or t_6 (the tests at +40°*C*) – Update the states of the boards and generate the event of the test unit reaching ambient temperature.
8. Test unit ends at ambient temperature down from +40°*C* – For Policy 1 – If any boards failed and if the operator is available, replace any failed boards (schedule the completion(s) of the test setup(s)). If the operator is not available, put the unit in the operator queue. For Policy 1 or Policy 2, if any boards are completing their final cycle, schedule the completion of the final test computer setup if an operator is available. Otherwise, put the unit in the operator queue. If no boards are completing their final cycle, schedule the arrival of the test unit at -40°*C*.
9. Boards requiring the final test (t_7) complete testing – If an operator is available, remove the completed board(s) as well as any failed boards (under Policy 1 only) and schedule the test setup completion event for the newly loaded board(s). If no operator is available, put the unit in the operator queue.
10. End of the simulation – Write the output report.

The C++ model is completely parameterized to simplify the testing of alternative system configurations. Table 2 lists and describes the model parameters that can be set at runtime of the model.

Table 2: Simulation Model Parameters

Parameter	*Description*
Test setup time	Time required for the operator to initialize a test
Board load time	Time required for the operator to load a new board on the test unit
Board unload time	Time required for the operator to unload a completed or failed board from the test unit
Test unit capacity	The number of boards that can be mounted on the test unit for simultaneous testing
Test durations	Durations of the individual tests (deterministic)
Failure probabilities	Failure probabilities for the individual tests. Note that failed boards must be repaired at an external repair station.
Board batch size	The number of boards in the batch
Number of test units	The number of test units available to process the batch
Number of operators	The number of operators available during the testing
Process ramp time	Time required for a test unit to ramp from -40°C to +40°C (and vice versa).

Table 3 lists the model outputs. Of the model outputs, the batch makespan is viewed as the most important by the sponsoring company. The batch size represents monthly demand and the makespan can be used to compute at which point during the month the required production will be completed. The sponsor was also interested in the overall capacity of the system under the alternative policies, but this performance metric is not considered in this paper.

4 TESTING AND RESULTS

The initial model was developed simply to answer the question as to whether or not boards should be replaced when they fail given the current operating configuration of the sponsoring company. Table 4 gives the current operating configuration for the sponsoring company (in terms of the simulation parameters defined in Table 2) and Table 5 gives the simulation results for this configuration. The results are based on 200 replications of the model. Note that, although they are not shown in the table, the 95% confidence intervals for the makespans and utilizations indicate that the means are statistically different.

Table 3: Simulation Model Outputs

Output	*Description*
Makespan	Clock-time required to process the given batch of boards
Operator utilization	Utilization of the operator during the makespan
% of test unit idle time	Percent of test unit idle time during the makespan
% of test unit waiting time	Percent of the test unit time spent waiting for operators during the makespan

Table 4: Current Operating Configuration

Parameter	*Current Value (times are in minutes)*
Test setup time	Normal (1, 0.25)
Board load time	Triangular (3, 4, 5)
Board unload time	Normal (1, 0.25)
Test unit capacity	3 boards
Test durations	Given in Table 1
Failure probabilities	Given in Table 1
Board batch size	300 boards
Number of test units	6 test units
Number of operators	3 operators
Process ramp time	Triangular (60, 75, 90)

Table 5: Current Configuration Results

	Policy 1 – Replace		Policy 2 - Don't Replace	
Performance Measure	Mean	Std. Dev.	Mean	Std. Dev.
Makespan	11790.1	192.59	12732.6	112.36
Test unit utilization	0.9840	0.0053	0.9745	0.0055
Operator utilization	0.1867	0.0024	0.1609	0.0013
% idle time	0.0149	0.0053	0.0246	0.0055
% waiting time	0.0011	0.0003	0.0009	0.0002

The results for the current system configuration clearly show that the replacement policy (Policy 1) is preferable in terms of the batch makespan. Test unit utilization for Policy 1 is slightly higher reflecting the shorter makespan and the fact that test unit capacity is wasted by not replacing failed units in Policy 2. The percent idle and waiting times are not significant for either policy.

Based on the initial results, a secondary set of tests was developed to evaluate the policy preference under slightly different configurations. The conditions were selected based on what might reasonably be expected in the sponsor's facility. As such, the following three factors were considered.

1. Number of boards (200, 300, 400)
2. Number of test units (5, 6, 7)
3. Number of operators (1, 3)

The replacement and non-replacement policies where tested using the 18 possible configurations based on the above factors. The makespan results for these tests are given in Table 6. Note that the test results are based on 200 independent replications of each configuration and that the makespans for each configuration are statistically different at the 95% confidence level.

As with the current configuration, the results for the other 17 considered configurations indicate that the replacement policy (Policy 1) is preferable to the non-replacement policy (Policy 2). These results were delivered to the sponsor with the clear message that, unless the configuration changes significantly, failed boards should be replaced as soon as possible.

5 ADDITIONAL TESTING

Intuitively, it would seem that Policy 2 (non-replace) would be preferable in certain configurations. As described earlier, there appears to be a clear tradeoff between wasting test unit capacity (under Policy 2) and unnecessarily delaying test units and the other boards during replacement (Policy 1). As such, configurations in which the total cycle time for the test units is shorter would seem to tip the scale in favor of Policy 2. Additional testing based on this intuition was performed.

Initially, the additional testing considered reductions to the process ramp time (the time required for the test unit to go from one temperature extreme to the other), thereby reducing the effects of the capacity loss when using Policy 2. Although reducing the process ramp time alone did not create a configuration in which Policy 2 performed better, this goal was achieved by simultaneously reducing both the process ramp time and the test times. In particular, reducing each of the test times to 10% of their original values and reducing the ramp time to triangular (.5, 1, 2) minutes resulted in a very small advantage in the makespan for Policy 2. The numerical results for this configuration are given in Table 7. In this configuration, Policy 2 does have a slightly lower makespan, but the difference is quite small.

The final configuration included in this paper also reduces the process ramp time (but to a lesser degree) and increases the test setup time (the time that the operator is required to start each test). The process ramp time for this

Table 6: Makespan Results for the Secondary Testing

			Policy 1 - Replace		Policy 2 - Don’t Replace	
# Operators	BatchSize	Test Units	Mean	Std. Dev	Mean	Std. Dev
1	200	5	9803.0	193.84	10615.6	151.54
		6	8338.5	174.55	9011.6	156.00
		7	7362.2	154.97	7955.4	128.75
	300	5	14574.8	244.56	15716.5	164.14
		6	12383.5	199.81	13368.9	148.03
		7	10839.8	196.06	11733.2	168.64
	400	5	19334.9	286.56	20850.9	177.09
		6	16380.8	245.99	17647.5	167.55
		7	14338.3	224.08	15409.1	152.96
3	200	5	9435.5	231.67	10265.1	145.98
		6	7932.0	170.88	8601.3	160.98
		7	6872.3	136.72	7445.8	83.75
	300	5	14083.4	245.32	15154.8	134.56
		6	11790.1	192.59	12732.6	112.36
		7	10152.3	152.19	11049.9	142.96
	400	5	18671.5	295.80	20144.1	159.50
		6	15577.8	226.00	16878.8	145.56
		7	13424.7	205.41	14477.4	127.31

Table 7: Additional Testing Results With Reduced Process Ramp and Test Times

Performance Measure	Policy 1 – Replace		Policy 2 - Don’t Replace	
	Mean	Std. Dev.	Mean	Std. Dev.
Makespan	2214.71	34.006	2124.88	12.078
Test unit utilization	0.7446	0.004	0.7568	0.0047
Operator utilization	0.9805	0.0037	.9641	0.0052
% idle time	0.0085	0.0034	0.0171	0.0049
% waiting time	0.2469	0.0037	0.2261	0.0036

configuration was set as triangular (9, 10, 11) minutes and the test setup times were set as normal (10, .25) minutes. Table 8 presents the numerical results for this final configuration.

The results of the additional testing confirm the intuition that Policy 2 will produce better results as the amount of time spend loading/unloading failed boards (in Policy 1) outweighs the capacity wasted by not replacing failed board (in Policy 2).

Table 8:Additional Testing Results with Reduced Process Ramp Times and Increased Operator Setup Times

Performance Measure	Policy 1 – Replace		Policy 2 - Don’t Replace	
	Mean	Std. Dev.	Mean	Std. Dev.
Makespan	4841.23	91.437	4553.13	27.576
Test unit utilization	0.7676	0.0052	0.8072	0.0034
Operator utilization	0.9843	0.0045	0.9726	0.0038
% idle time	0.0104	0.0044	0.0166	0.004
% waiting time	0.222	0.0049	0.1762	0.0034

6 CONCLUSIONS

This paper describes a simulation analysis of a complex printed circuit board testing process. Boards are grouped on a test unit to go through three thermal cycles at seven individual tests. The objective of the analysis was to determine whether failed boards should be replaced as soon as possible after they fail or whether they should be left on the test unit until all boards complete testing.

Testing indicated that replacing boards as soon as possible when they fail is the preferred policy in all tested configurations close to the current configuration in the sponsor's facility. Additional testing was performed and several configurations for which the no replacement policy is preferred were identified, but these contrived configurations were significantly different from the actual system.

Based on the results of this analysis, the initial recommendation to the sponsor is to operate the system under the replacement policy. Further analysis considering "hybrid" policies is currently being done. Hybrid policies involve replacing *some* boards when they fail. Under these policies, the replacement decision is made based on when the failure occurs and on the availability of the operators.

REFERENCES

Banks, J., J. S. Carson, B. L. Nelson, and D. M. Nicol, 2001, *Discrete-Event System Simulation*, 3rd Edition, Prentice Hall, Upper Saddle Rirver, NJ.

Pegden, C. D., R. E. Shannon, and R. P. Sadowski, 1995, *Introduction to Simulation Using SIMAN*, 2nd Edition, McGraw-Hill, Inc., New York, NY.

AUTHOR BIOGRAPHIES

JEFFREY S. SMITH joined Auburn University as an associate professor in the Industrial & Systems Engineering department in September of 1999. Prior to joining Auburn, Dr. Smith was an associate professor in the Industrial Engineering Department at Texas A&M University. He received the B.S. in Industrial Engineering from Auburn University in 1986 and the M.S. and Ph.D. degrees in Industrial Engineering from Penn State University in 1990 and 1992, respectively. In addition to his academic positions, Dr. Smith has held industrial positions at Electronic Data Systems and Philip Morris U.S.A. Dr. Smith is an active member of IIE, SME, and INFORMS. His email and web addresses are <jsmith@eng.auburn.edu> and <www.eng.auburn.edu/~jsmith>.

YALI LI is an M.S. student in the Department of Industrial & Systems Engineering at Auburn University. She received her B.S. degree in Process Control from Zhejiang University, China, in 1996. Her current research interests are in discrete event simulation and modeling for complex systems. Her email address is <yalili@eng.auburn.edu>.

JASON GJESVOLD joined Soldering Technology International as a project engineer in the Engineering Services Division in August of 2000. Prior to joining Soldering Technology, Mr. Gjesvold was a manufacturing engineering co-op at Schlumberger Industries, Resource Management Division. He received a B.S. in Mechanical Engineering from Auburn University in 2000, and is currently working towards a Master of Engineering Degree in Mechanical Engineering at North Carolina State University. Mr. Gjesvold is the Analytical Lab Manager at Soldering Technology and is responsible for directing failure analysis / root cause analysis efforts, as well as simulation development and performance metric development / implementation. Mr. Gjesvold is an active member of ASME, ASM, and SMTA. His email address is <jgjesvold@solderingtech.com>.