FINAL REPORT

Summary: Experimental Validation of Real-Time Fault-Tolerant Systems

Supported by NASA Langley Research Center

under grant NASA NAG-1-602

R. K. Iyer and G. S. Choi

Center for Reliable and High-Performance Computing
Coordinated Science Laboratory
University of Illinois
URBANA IL 61801
(217) 333-3774
FAX: (217) 244-5686
iyer@crhc.uiuc.edu

Scientific Officer:
Celeste M. Belcastro
NASA Langley Research Center
Hampton, VA 23665-5225

October 1992

Summary: Experimental Validation of Real-Time Fault-Tolerant Systems

Research Summary

R. K. Iyer and G. S. Choi

August 28, 1992

Center for Reliable and High-Performance Computing
Coordinated Science Laboratory
University of Illinois
URBANA IL 61801

Introduction

This is the final report on the research performed under NASA grant NAG-1-602. The project has been concerned with the validation of real-time fault-tolerant systems with particular emphasis on upset analysis.

The aim of the research was to investigate fundamental techniques to aid in the design of high dependability systems. In particular, we addressed issues concerning the sensitivity of a design to single and multiple upsets under realistic application environments. Automated reliability analysis/design techniques were also developed for the systematic evaluation of VLSI chips. The methodology can locate areas or design features that are susceptible to common wear-out mechanisms.

Such techniques are essential if NASA's plan for developing complex, integrated aerospace systems of the future is to succeed. We believe that we have progressed substantially toward our objective. The developed approach is a combination of experimental and analytical techniques, and uses an actual system design to evaluate and illustrate the results of this research.

Summary

Testing and validation of real-time systems is always difficult to perform since neither the error generation process nor the fault propagation problem is easy to comprehend. There is no better substitute to results based on actual measurements and experimentation. Such results are essential for developing a rational basis for evaluation and validation of real-time systems. However, with physical experimentation, controllability and observability are limited to external instrumentation that can be *hooked-up* to the system under test. And this process is quite a difficult, if not impossible, task for a complex system. Also, to set up such experiments for measurements, physical hardware must exist. On the other hand, a simulation approach allows flexibility that is unequaled by any other existing method for system evaluation. We have successfully developed and implemented a simulation methodology for system evaluation and demonstrated the environment using existing real-time avionic systems.

Our research has been oriented toward evaluating the impact of permanent and transient faults in aircraft control computers. Results were obtained for the Bendix BDX 930 system and Hamilton Standard EEC131 jet engine controller. The studies showed that simulated fault injection is valuable, in the design stage, to evaluate the susceptibility of computing systems to different types of failures.

Experiences with Bendix BDX-930 flight control computer

The BDX-930 flight control computer, which we have used in the early phase of our research, is based on the AMD 2901 bit-slice processor. This chip was the focus in the error propagation study as it is the most complicated chip in the computer. The computer was fabricated and originally emulated by Bendix Corporation. A simulator for this computer was later developed at NASA-Langley. This simulator was the tool for our initial research in error propagation.

The motivation for this work was a lack of understanding of how errors propagate in a system. By using the simulator developed at NASA, we have gained insight into the fault behavior of a real system running real instructions. We also studied the classical assumptions in reliability/availability models. Specifically, the accuracy of the pin-level fault models was investigated. By simulating at the gate level, and following the error propagation to the pins, we found that the pin-level single-bit fault model does not accurately represent the true fault behavior. Also, we obtained the results on dependency of fault propagation on system activity through simulating the system running real instructions.

EEC131 jet-engine controller

Since then, we have proposed a range of hierarchical fault-injection methods with specific emphasis on real-time fault-tolerance in air- and space-borne computing. The proposed methods are particularly suited for quantifying the susceptibility of complex VLSI architectures to internal current and voltage transients. An automated environment, **FOCUS**, which allows for the run-time injection of transients and for their tracing, from the device and the pin level to the program flow-level, has been developed. The environment incorporates a graphics facility to enhance the user interface and to provide real-time propagation and characterization information. Graphical analysis of the results of the validation experiments is also displayed. The EEC131 jet-engine controller has been used to illustrate the methodology in a realistic setting.

Some key results have been obtained from the analysis of the EEC131 controller. Over 3000 fault injections were performed on the HS1602 microprocessor for the EEC131 controller and the results were collected and analyzed. We determined, experimentally, that at least 3 gate levels are needed for accurate electrical level analysis and performed our experiments accordingly. A charge threshold of 2 picoCoulombs, experimentally determined, was needed for transients to have any impact at the logic level. About 22% of the injected faults resulted in latch errors, and 12% resulted in pin errors. Once an error was latched, an average of eight latches were affected. The injected faults had a 42% chance of having no effect and a 41% chance of having a data upset. There was also a 17% chance of having a program control-flow deviation. About 81% of the software upsets were single point (in time and space) in nature, and about 19% of them were multiple. From the result of our experiments, we found that the ALU unit was the most sensitive to transients. The fault/error latency was monitored, and an error explosion/degeneration model was generated to show the dynamics of internal latched-error behavior. The impact of latch errors lasted at most 8 clock cycles from the time of fault occurrence. A state transition model was generated to depict the module-to-module in-chip error propagation. It indicated that the watchdog and control units were the most sensitive to fault propagation.

We have also investigated the behavior of the target system at the program flow-level. Program deviations due to the injected transients have been monitored and analyzed. Using test workloads, an evaluation of the resulting functional upsets has been made. The analysis showed that there is a significant chance that a single point transient can cause multiple upsets at the program-flow level. They suggest that current methods of validation that assume single upsets may be inadequate. For the HS1602, the microprocessor for the EEC131 controller, there is about a 15 percent chance of having multiple program upsets due to single transients. External fault conditions, such as disturbances on lines, have also been investigated. Our experiments include the injection of oscillating electrical waveforms on the power supply lines. Again, program deviations (both single and multiple) were monitored. For the HS1602, the analysis shows that, disturbances of durations less than 225 nanoseconds(nS) have no impact on the functionality of the chip. Power disturbances greater than 225 nS can cause both single and multiple upsets. Eventually, system failure occurs, on the average, beyond 525 nS. The results show that the techniques developed can provide considerable insight into the functional behavior of the system.

Reliability Simulation:

Automated reliability analysis techniques were developed to systematically evaluate a VLSI chip to see whether it meets its reliability specifications. A methodology to locate areas or design features that are susceptible to common wear-out/failure mechanisms was developed. In particular, the wear-out mechanisms involved in each device failure their effect at the chip-wide level was developed. Using the method, we have determined the reliability distribution for the target system under a realistic operating environment. Different fabrication technology parameters were also tested. Concurrent simulation of wear-out on a dynamic scale will be performed for the entire chip. Key advantages of this approach are that it can closely mimic dynamic sequences of events in a device, that it can localize the *weak location/aspect* of a target chip, and that it allows the generation of the TTF(Time-to-failure) distribution of the entire VLSI chip. First, an accurate switch-level simulation of the target chip and its application code is performed to acquire trace data (real work load) on switch activity. Then, using this switch activity information, wear-out of each component in the entire chip is simulated using Monte Carlo techniques. For example, the process of electromigration is modeled by removing elements (metal grains) from a matrix that depicts a metal line. The probability of grain removal is based on a empirical probability distribution. This process is carried out in parallel for all the metal lines in the circuit. A metal line failure, i.e. chip failure, is assumed to occur if the matrix-connection is broken. The analysis is performed a number of times to determine the distribution of the TTF of the target chip. Concurrently, a similar Monte Carlo approach is used to simulate failures due to other wear-out mechanisms such as dielectric breakdown.

In order to perform such analysis within a reasonable time period we have employed a technique to accelerate the wear-out process in the simulation analysis. The technique is somewhat like an accelerated "burn-in" test, and it is called "Importance Sampling." It accelerates the the wear-out mechanisms by biasing the related parameters in order to increase the chance of failure. We then compensate for the distortion (bias) in the result by dividing the result(TTF) with the biased estimator.

Conclusion

The development of new automated techniques for cost-effective validation of high-dependability systems has been successfully demonstrated. The methodology used is hierarchical, in that it can allow the integration of various levels of validation. The reviews among peers have been very positive, and several major journal and conference papers have been published/submitted as result of this work. It is expected that we will maintain close cooperation with the researchers at NASA AIRLAB to exchange further developments and useful insights in this area.

References

[1] D. L. Lomelino and R. K. Iyer, "Error Propagation in a Digital Avionic Processor: A Simulation-Based Study," *Proceedings Real-Time Systems Symposium*, pp. 218-225, December 1986.

[2] S. Kim and R.K. Iyer, "Impact of Device Faults in a Digital Avionic Processor," *8th Digital Avionics Systems Conference*, San Jose, CA, Oct. 1988.

[3] S. Kim, *Fault Behavior Analysis: An Experimental Study*, M.S. Thesis, University of Illinois, 1988.

[4] R. Chillarege, *Fault and Error Latency Under a Real Workload: An Experimental Study*, Ph.D. Thesis, CSG Technical Report No. 55, University of Illinois, 1987.

[5] P. Duba and R.K.Iyer, "Investigation of Transient Fault Behavior in a Real Time Microprocessor: A Case Study," *Proceedings IEEE International Conference on Computer Design: VLSI in Computers & Processors*, Port Chester, NY, pp. 272-276, October 3-5, 1988.

[6] P. Duba, "Transient Fault Analysis: A Case Study," M.S. Thesis, CRHC Technical Report, University of Illinois, 1988.

[7] G. Choi, R. Iyer, R. Saleh and V. Carreno, "A Fault Behavior Model for an Avionic Microprocessor: A Case Study," *Proceedings of International Working Conference on Dependable Computing for Critical Applications*, University of California, Santa Barbara, CA, August 23-25, 1989.

[8] G. Choi, "An experimental study of fault propagation in a jet-engine controller," M.S. Thesis, CRHC Technical Report, University of Illinois, 1989.

[9] G. Choi, R. Iyer and V. Carreno, "Simulated Fault Injection: A Methodology to Evaluate Fault Tolerant Microprocessor Architectures," *IEEE Trans. on Reliability*, October, 1990.

[10] V. Carreno, G. Choi and R. Iyer, "Analog-Digital Simulation of Transient-Induced Logic Errors and Upset Susceptibility of an Advanced Control System," NASA Technical Memorandum 4241, November 1990.

[11] G. Choi and R. Iyer, "FOCUS: An Experimental Environment For Fault Sensitivity Analysis," Submitted to *IEEE Trans. on Computers* 1991.

[12] K. Goswami, G. Choi and R. Iyer, "Design for Dependability," Proceedings, AIAA Computing in Aerospace 8 Conference, October 1991.

[13] G. Choi and R. Iyer, "Investigation of single and multiple upsets in an avionic system", *Center for Reliable and High-Performance Computing Tech. Rept.* University of Illinois at Urbana-Champaign, 1991.

[14] G. Choi and R. Iyer, "Wear-out simulation of VLSI systems," *Center for Reliable and High-Performance Computing Tech. Rept.* University of Illinois at Urbana-Champaign, 1991.