Chalmers University of Technology
Department of Computer Engineering
412 96 Göteborg

1999-08-17

Node-level Fault Tolerance for Fixed Priority Scheduling

An ARTES supported research project

Progress report

Johan Karlsson

1 Project summary

Techniques for error detection and fault tolerance for distributed real-time systems using fixed priority preemptive scheduling will be studied. Specifically, techniques for achieving node-level transient fault tolerance using massive time redundancy will be developed.

Error detection and fault tolerance mechanisms will be implemented in a small real-time kernel and evaluated using both VHDL-based fault injection and fault injection in a real system. The real-time kernel will be implemented for the THOR microprocessor developed by Saab Ericsson Space AB, who will supply a detailed VHDL simulation model of THOR and computer boards.

The project supports one Ph.D. student, Joakim Aidemark, for 2 years.

2 Progress

Since the projected started in April 1999, Joakim has studied the design of the THOR microprocessor and have learned how to run fault injection experiments on the VHDL model of THOR using the MEFISTO fault injection tool. He has conducted a fault injection campaign to evaluate the effectiveness of the hardware error detection mechanisms included in THOR. Approximately 20 000 faults were injected into the THOR processor during execution of a sorting program to estimate error detection coverage. One goal of these experiments was to study how variations in input data affect the error detection coverage.

The results of this fault injection campaign will provide important input and experience for the fault injection campaigns that will carried out with the real-time kernel.

Joakim has also made a literature survey and studied several papers on fault-tolerant scheduling. In addition, he has served as teaching assistant in a course on real-time computing, which has consolidated this knowledge of the foundations of real-time computing.

Recently, he has started working on the design of the real-time kernel that will support time redundant executions of tasks to achieve high error detection coverage and transient fault-tolerance.

3 Scientific merits

Designs of fault-tolerant distributed real-time systems are always based on assumptions about the failure semantics (failure behavior) of the computer nodes. Deterministic and well-behaved

failure semantics such as fail-silent behavior makes it possible to use simple and efficient protocols to tolerate node failures at the system level. Tolerating arbitrary node failures, also called Byzantine failures, requires complex and time-consuming protocols which in many cases are too costly to use in real-time systems.

To ensure deterministic failure semantics, it is necessary to incorporate internal error detection in the computer nodes. However, it is not possible to guarantee that the desired failure semantics will be fulfilled for all types faults that may occur inside a node, if the cost of the internal error detection is to be kept at a reasonable level. Thus, it is not reasonable to assume that fault coverage is perfect. (In this context, fault coverage is usually defined as the conditional probability that the specified failure semantics are fulfilled given that fault has occurred in the computer node.)

An important goal for system designers is to achieve high fault coverage at low cost. The main scientific contribution of this project is to provide new techniques for achieving this goal for real-time systems that rely on fixed-priority scheduling. We have chosen fixed-priority scheduling as the framework of our research, since we previously done similar work for time-triggered scheduling. Another reason for choosing fixed-priority scheduling, is that response time analysis is well understood for this scheduling technique, which is imperative for systems used in critical applications.

The most important contribution of the project will be to investigate techniques for achieving transient fault-tolerance at the node-level by using time redundant execution and checkpointing. Recent developments in microprocessor technology have shown tremendous increase in processing power, which has made time-redundancy increasingly attractive for the design of fault-tolerant real-time systems. With the use of high-performance microprocessors, it is now feasible in many applications to use time redundancy on a massive scale to achieve not only error detection both also transient fault tolerance at the node-level.

Another important contribution will be to refine and further develop techniques for fault coverage estimation. We have access to a highly sophisticated fault injection environment, which we have developed in our previous research projects. This allows us to inject faults both in a highly detailed simulation model of the THOR-processor, and in a real THOR-based system via the processor’s internal scan-chains. The simulation model provides a very high degree of observability and controllability, while the real system allows us to study the impact of faults at full execution speed with excellent controllability and observability.

4 Industrial relevance and exploitation expectations

The problem of designing internal error detection and transient fault tolerance in computer nodes of distributed real-time systems is highly relevant for a large number of real-time applications areas such as automotive electronics, factory automation, telecommunications and aerospace systems.

It is expected that the project will provide significant guidance to system designers, since it will give a general indication of the level of fault coverage that can be achieved by combining time-redundancy and behavior-based error detection. However, it will not be possible to translate the actual numbers obtained for the THOR-system to other systems. Any system that uses these techniques must be individually validated by comprehensive fault injection experiments, since both the software and the hardware architecture, as well as implementations details, may have a significant impact on fault coverage. Therefore, we also expect that the fault injection methodology developed in the project will provide valuable insights for engineers working with system validation.

The results of the study are also expected to be of significant value to Saab Ericsson Space for future development of the THOR-processor and on-board computer systems for launchers and satellites.

5 Mobility

We have not applied for funds for mobility. Since both Chalmers and Saab Ericsson Space are located in Göteborg, the partners can easily visit each other when ever it is necessary. So far one progress meeting has taken place at Saab Ericsson Space.

6 Publications

We have not yet published any reports or papers within this project. Some references to earlier work is given below. Our earlier evaluation of the MARS system is described in [1]. The MEFISTO tool is described in [2], and the FIMBUL tool in [3]. Examples of fault injection based evaluation can be found in [4] and [5].

[1] Johan Karlsson, Peter Folkesson, Jean Arlat, Yves Crouzet, Günther Leber and Johannes Reisinger, "Application of Three Fault Injection Techniques to the Experimental Assessment of the MARS Architecture", *Proc. 5th IFIP International Working Conference on Dependable Computing for Critical Applications (DCCA-5*), IEEE Computer Society Press, Urbana, IL, USA, September 1995.

[2] Eric Jenn, Jean Arlat, Marcus Rimén, Joakim Ohlsson and Johan Karlsson, "Fault Injection into VHDL Models: The MEFISTO Tool", *24th Int. Symp. on Fault Tolerant Computing (FTCS-24)*, IEEE, Austin, TX, USA, June 1994.

[3] Peter Folkesson, Sven Svensson and Johan Karlsson, "A Comparison of Simulation Based and Scan Chain Implemented Fault Injection", *Proc. 28th Int. Symp. on Fault Tolerant Computing (FTCS-28)*, IEEE Computer Society Press, Munich, Germany, June 1998

[4] Marcus Rimén, Joakim Ohlsson and Johan Karlsson, "Experimental Evaluation of Control Flow Errors", *Proc. 1995 Pacific Rim International Symposium on Fault Tolerant Systems (PRFTS)*, IEEE Computer Society Press, Newport Beach, CA, USA, December 1995.

[5] Ghassem Miremadi, Joakim Ohlsson, Marcus Rimén and Johan Karlsson, "Use of Time and Address Signatures for Control Flow Checking", *Proc. of the 5th International Working Conference on Dependable Computing for Critical Applications (DCCA-5)*, IEEE Computer Society Press, Urbana Champaign, IL, USA, September 1995

7 Web page

The project WEB-page is not yet available.