A Hardware-Based Approach for Fault Detection in RTOS-Based Embedded Systems

D. Silva, K. Stangherlin, L. Bolzani, F. Vargas
Electrical Engineering Dept., Catholic University – PUCRS
Av. Ipiranga 6681, 90619-900, Porto Alegre, Brazil
leticia@poehls.com

Abstract—**This paper presents a new hardware-based approach able to monitor the execution flow of the Real-Time Operating System (RTOS) in order to detect faults changing the tasks' execution flow in embedded systems. Results obtained during experiments performed according to the IEC 61000-4-29 standard demonstrate that the proposed approach is able to provide higher fault coverage with respect to the fault detection mechanisms natively provided by the RTOS.**

Keywords- RTOS, Hardware-Based Approach

I. Introduction and The Proposed Approach

Real-Time Operating Systems (RTOSs) have become an interesting solution to simplify the design of safety-critical embedded systems. Due to the environment's always increasing hostility, real-time systems are often subject to transient faults originated from several sources including Electromagnetic Interference (EMI), which can affect the system changing its correct behavior [1][2]. In this paper a new passive real-time scheduling monitoring approach implemented by a new I-IP, named RTOS-Guardian (RTOS-G), is presented. Differently from the previously presented approach, the RTOS-G monitors the task's execution flow based on preemptive algorithm [3][4]. The RTOS-G is connected to the embedded system's bus and is composed of four functional blocks: (1) *Task Controler* (TC), which identifies the task in execution, (2) *Function Identifier* (FI) that identifies the functions assosiated with the task scheduling, (3) *List Monitor and Error Generator* (LMEG), which receives the *Scheduler_Event* signal and the task in execution in order to classify all tasks in two seperate lists, *ready tasks* and *blocked tasks*, each organized according to their state and priority and (4) *Content-addressable memory* (CAM1) as well as (5) (CAM2), which save the lists of tasks labeled ready or blocked, respectively. It is important to note that the LMEG implements the scheduling algorithm and indicates errors when a scheduling misbehavior is detected.

II. Experimental Results

To evaluate the hardware-based approach we adopted a case study composed of a Von Neumann 32-bit RISC Plasma microprocessor running an RTOS. The adopted RTOS provides a basic mechanism able to monitor the task's execution flow and to manage particular situations when faults cause misbehaviors of the RTOS's essential services. Thus, the fault detection capability of the RTOS-G has been evaluated with respect to the native fault detection mechanisms of the RTOS. During the experiments we considered three different, suitable developed benchmarks that exploit great part of the resources offered by the Plasma's RTOS. Finally, we performed 1000 fault injection experiments with each benchmark, applying voltage dips to the FPGA *Vdd* pin according to IEC 61000-4-29. It is important to note that an experiment finishes when a fault is detected by the RTOS or the RTOS-G.

III. Final Considerations

The main contribution of this paper consists of providing significantly more robust embedded systems based on RTOSs. The proposed approach has been evaluated applying power supply disturbances. The RTOS-G provides nearly 100% fault coverage, introducing only about 10% area overhead. Moreover, it is important to highlight that the non-hardened system was able to provide only 2.4% in BM1 or 25.9% in BM2, respectively. Further the introduction of the RTOS-G reduces the fault latency to about 2% with respect to the native RTOS' latency.

To conclude, we are convinced that the hardware-based approach proposed in this paper represents an interesting solution for hardening real-time embedded systems based on RTOSs.

References

[1] N. Ignat, B. Nicolescu, Y. Savari, G. Nicolescu, "Soft-Error Classification and Impact Analysis on Real-Time Operating Systems", IEEE Design, Automation and Test in Europe, 2006.

[2] E. Touloupis, J. A. Flint, V. A. Chouliaras, D. D. Ward, "Study of the Effects of SEU Induced Faults on a Pipeline Protected Microprocessor", IEEE TC, 2007.

[3] J. Tarrillo, L. Bolzani, F. Vargas, "A Hardware-Scheduler for Fault Detection in RTOS-Based Embedded Systems", IEEE 12th EUROMICRO Conference on Digital System Design, 2009.

[4] J. Tarrillo, L. Bolzani, F. Vargas, E. Gatti, F. Hernandez, L. Fraigi, "Fault-Detection Capability Analysis of a Hardware-Scheduler IP-Core in Electromagnetic Interference Environment", IEEE 7th East-West Design & Test Symposium, 2009.

1530-1877/11 $26.00 © 2011 IEEE
DOI 10.1109/ETS.2011.64