

Alternativas de Implementación y Evaluación de Sistemas de Tiempo Real

Fernando Romero, Diego Encinas, Diego Montezanti, Emanuel Frati, Luciano Iglesias, Javier Giacomantone, Armando De Giusti, Fernando G. Tinetti¹

Instituto de Investigación en Informática LIDI (III-LIDI)
Facultad de Informática – UNLP

{fromero, dencinas, dmontezanti, fefrati, li, jog, degiusti, fernando}@lidi.info.unlp.edu.ar

RESUMEN

El objetivo de esta línea de investigación es el estudio y desarrollo de sistemas de software que poseen restricciones temporales, sobre todo en lo referente a las plataformas donde aplicaciones de este tipo pueden ejecutarse.

Se trabaja experimentalmente con diferentes sistemas operativos que soportan las características de tiempo real.

CONTEXTO

Esta línea de Investigación forma parte del proyecto "Arquitecturas multiprocesador distribuidas. Modelos, Software de Base y Aplicaciones" del Instituto de Investigación en Informática LIDI acreditado por la UNLP. El planteo que se presenta constituye una línea incipiente en el III-LIDI

Palabras Claves: Tiempo Real, Hardware, Sistemas Operativos, Sistemas Embebidos, Posix.

1. INTRODUCCION

Los sistemas de tiempo real (STR) [4] [5] [6] [11] [10] son sistemas de tiempo crítico, es decir, se trata de aplicaciones para las cuales las restricciones de tiempo de cada tarea a llevar a cabo son un requerimiento esencial. Esto conlleva a un cuidadoso diseño de los sistemas operativos empleados, en particular al planificador del mismo, que será en gran parte responsable del cumplimiento de dichos plazos.

Por lo general, un STR incluye un conjunto de dispositivos relativamente independientes (hardware y software) que operan a diferentes velocidades. Deben satisfacer requerimientos de rendimiento. Comparando con un sistema tradicional, se debe tener especial control sobre la utilización de recursos, tiempos de respuesta más estrictos, manejo de prioridades y de fallos.

Existen algunas diferencias importantes entre un sistema de software tradicional y uno de tiempo real [5][9] principalmente debido al peso de los siguientes tópicos:

¹ Investigador CIC prov. De Buenos Aires.

- Planificación de tareas dirigida al cumplimiento de las restricciones de tiempo en vez de al rendimiento.
- Control de dispositivos externos.
- Mayor uso de la E/S, ya que por lo general controla un proceso físico o químico, lo que implica interactuar con sensores, detectores y actuadores.
- Procesamiento de mensajes que pueden llegar en intervalos irregulares, algunos periódicos y otros aleatorios.
- Seguridad y confiabilidad son conjuntas en sistemas de este tipo, debe haber detección y control de condiciones de falla a fin de evitar daños y un funcionamiento sin interrupciones. Debido a que muchas veces estos sistemas no cuentan con supervisión humana (autónomos) el manejo de excepciones es crítico. Se espera que un STR se ejecute en forma continua, automática y segura, teniendo un impacto en los costos de desarrollo y la seguridad.
- Modelización de condiciones concurrentes, ya que al mezclarse gran número de eventos periódicos y esporádicos es frecuente su concurrencia como la de los procesos cuya ejecución desencadenen dichos eventos, con lo que se debe tratar la aloca-ción y control de procesos concurrentes.
- Manejo de las comunicaciones entre procesos.
- Protección de datos compartidos.
- Manejo de requerimientos de tiempos y rendimiento.
- Testeo y debug de procesos concurrentes.
- Diseño de simuladores del hardware que no se encuentra disponible durante la fase de pruebas.
- Selección del hardware adecuado para soportar el diseño del software, frecuentemente estos sistemas utilizan hardware especialmente diseñado.
- Distribución de tareas (Sistemas Distribuidos de Tiempo Real).
- Por lo general se trata de Sistemas Embebidos, o sea sistemas que forman parte de un sistema /entorno mayor dentro del cual están insertados y que controlan.

2. LINEAS DE INVESTIGACION Y DESARROLLO

Se plantean como temas de estudio:

- Benchmarking de Sistemas Operativos de Tiempo Real [3], determinando bajo qué restricciones son aptos. En este tema se están desarrollando experimentos a fin de medir la latencia en la respuesta a una interrupción, o sea el tiempo transcurrido desde que se produce dicha interrupción hasta que se ejecuta la primera instrucción del programa que la atiende. Se realizan pruebas sobre MaRTEOS [1], RTLinux, RTAI, Linux 2.6 y Linux 2.6 preemptive.
- Dispositivos de Tiempo Real. Entre otras cosas, se está experimentando la implementación de sistemas sobre plataforma de microcontroladores.
- Políticas de planificación de CPU [2] [1] [8] [10]: en STRSe trabaja en simulaciones de diferentes algoritmos de planificación de uso de la CPU, utilizando la herramienta Cheddar [7]. Con esta herramienta es posible simular procesos con diferentes restriccio-

nes temporales y de procesamiento, con diversas prioridades, empleando planificadores que emplean diferentes políticas. También es posible emplear planificación a nivel de aplicación. En base a estas simulaciones, se intentará implementar planificadores experimentales tanto en MarteOS como en Linux preemptive[11].

- Recursos compartidos: Inversión de prioridades (simple e ilimitada): Se llevan a cabo experimentos sobre los mismos SOTR mencionados, a fin de evaluar ventajas y desventajas de los mecanismos implementados en los mismos para evitar los efectos desfavorables de este fenómeno, en el cual un proceso de menor prioridad, bloquea a uno de mayor prioridad.

3. RESULTADOS OBTENIDOS/ESPERADOS

De acuerdo con las tareas desarrolladas y a desarrollar, los resultados se enfocarán en varias direcciones relacionadas con sistemas operativos y con sistemas embebidos de tiempo real:

- Se estudian los sistemas operativos MaRTeOS, RT-Linux, RTAI y Linux kernel 2.6 con parche-preemptive, explorando sus posibilidades para el desarrollo de aplicaciones en tiempo real, realizando ejemplos concretos.
- Identificación y/o construcción de benchmarks sobre sistemas operativos tradicionales para evaluar la posibilidad de ejecutar sistemas con diferentes tipos de restricciones de tiempo real a considerar.
- Identificación y definición de pautas para estrategias en Planificadores y su impacto en los sistemas en STR,

ensayando las mismas en la herramienta Cheddar.

- Llevar a cabo experiencias reales en RTOS, en cuanto al desarrollo de aplicaciones, análisis de sobrecarga y planificadores. Evaluar la posibilidad de combinaciones de problemas y soluciones con análisis de evaluación objetivo.
- Uso de robots y simuladores de robots Kephra. Relación de la problemática de aplicaciones de robots y los sistemas embebidos de tiempo real.

4. FORMACION DE RECURSOS HUMANOS

En base a estos temas se están desarrollando un doctorado y dos posibles tesis de grado. También aportan trabajos de alumnos de la materia Diseño de Sistemas de Tiempo Real.

5. BIBLIOGRAFIA

1. Mario Aldea Rivas and Michael González Harbour. "MaRTE OS: Overview and Linux Version". Talk for the Real-Time Systems Group, York, November 2004.
2. N.C. Audsley , A. Burns , M. F. Richardson , A. J. Wellings. "Hard Real-Time Scheduling: The Deadline-Monotonic Approach". 1991. Proc. IEEE Workshop on Real-Time Operating Systems and Software.
3. Barbalace, A.; Luchetta, A.; Manduchi, G.; Moro, M.; Soppelsa, A.; Taliercio, C. "Performance Comparison of VxWorks, Linux, RTAI, and Xenomai in a Hard Real-Time Application". 2007. Real-Time Conference, 15th IEEE-NPSS

4. L. Buhr. "An Introduction to Real Time Systems". Prentice Hall, 1999.
5. A. Burns, A. J. Wellings. "Designing hard real-time systems". 1992. Proceedings of the 11th Ada-Europe international conference on Ada. ISBN:0-387-55585-4
6. A. Burns & A. Wellings. "Real-Time Systems and Programming Languages". Addison Wesley, ISBN 90-201-40365-x
7. "The Cheddar project: a free real time scheduling analyzer" <http://beru.univ-brest.fr/~singhoff/cheddar/>
8. Coulouris G., Dollimore J., Kindberg T. "Distributed Systems Concepts and Design", Addison Wesley 1994
9. S. Ellison, "Developing Real-Time Embedded Software". Wiley, 1994
10. A. Silberschatz, J. Peterson, P. Galvin, "Sistemas Operativos. Conceptos Fundamentales", 3rd Ed. Addison-Wesley Iberoamericana, 1994.
11. Siro Arthur. Carsten Emde. Nicholas Mc Guire. "Assessment of the Realtime Preemption Patches (RT-Preempt) and their impact on the general purpose performance of the system". 20 nov 2007. 9th RT Workshop.