

El Software Embebido y los Retos que Implica su Desarrollo

Nota de divulgación

M. C. José Isidro Hernández Vega

Departamento de Ingeniería en Sistemas Computacionales

Instituto Tecnológico de Nuevo León, Av. Eloy Cavazos No. 2001, Col. Tolteca, Ciudad Guadalupe, Nuevo León,

C. P. 67170, Tel: 01 (81) 8157-0500 Ext. 161, Fax: 01 (81) 8157-0501 Ext. 306. Página web: www.itnl.edu.mx

Correo electrónico: pepe1602002@yahoo.com.mx

Resumen

Este artículo presenta al software embebido como un tipo de aplicación muy particular en la ingeniería de software, su conceptualización, aplicación en las diversas actividades del ser humano, características específicas que se demandan al construirlo. Se realiza un análisis de los retos que se presentan durante su proceso de desarrollo tomando en cuenta factores que repercutan en una mala calidad del producto y del software embebido implementado, donde la confiabilidad es fundamental. Además se habla de las tendencias que se tienen a futuro en la mejoría de sus herramientas de diseño, así como en la cobertura de sus aplicaciones.

Palabras clave: Software embebido, sistemas embebidos, aplicaciones de software embebido, sistemas mecatrónicos, FPGA, tiempo real, sistemas de tiempo real.

Abstract

This article presents what is the embedded software as a type of very particular application in the software engineering, its conceptualization, application in the human being diverse activities, characteristic specific that are demanded when building it. It is carried out an analysis of the challenges that are presented during their development process taking in bill factors that rebound in a bad quality of the product and of the implemented absorbed software, where the dependability is fundamental. It is also spoken of the tendencies that are had to future in the improvement of their design tools, as well as in the covering of their applications.

Key words: Embedded Software, embedded systems, Embedded Software applications, mechatronic systems, FPGA, Real Time, Real-time systems.

Introducción

Los microprocesadores son cada vez más poderosos y baratos, se pueden encontrar hoy en día productos que contienen microprocesadores para dotarlos de cierta inteligencia. Los relojes, los automóviles, las cámaras fotográficas y de video, los reproductores de música entre otros productos; todos ellos ejecutan software. El software embebido se refiere a los sistemas de cómputo que reside en muchos casos, sin que el usuario se entere, dentro de estos productos.

Éste software forma parte de un sistema embebido el cual podemos entender como un subsistema electrónico de procesamiento, programado para realizar una o pocas funciones para cumplir con un objetivo específico. Generalmente es parte integral de un sistema heterogéneo mayor, que puede incluir partes mecánicas, eléctricas y/o electromecánicas. La historia de los sistemas embebidos se remonta a comienzos de los años 60, cuando dispositivos basados en microprocesadores y microcontroladores comenzaron a emplearse en el control de tareas aeronáuticas y espaciales. Las limitaciones de alto costo y diseño de estos primeros dispositivos provocaron una espera hasta 1992 en el que se creó el consorcio PC/104 [1], formado por Ampro, RTD y otros fabricantes.

El software como parte de un sistema embebido se utiliza para controlar los productos electrónicos a nivel lógico [2]. Desarrollarlo involucra retos completamente diferentes a los que la mayoría de los programadores de software está acostumbrado, entender en forma clara y precisa lo que es este tipo de software es fundamental para su desarrollo, visualizar área de aplicación y por consecuencia de comercialización y de solución de problemas que impliquen un avance tecnológico contribuirá al desarrollo del país.

Comprender las características de este tipo de software, así como lo que debemos tomar en cuenta para su desarrollo contribuirá a tener aplicaciones exitosas.

Todos estos planteamientos son los que se tratan en la presente publicación, además de dar un panorama de las tendencias de desarrollo del software embebido.

Desarrollo

Definición de software embebido

Los productos inteligentes se han convertido en algo común en casi todos los mercados de consumo industriales. El software embebido o empotrado reside en memoria de sólo lectura y se utiliza para controlar productos y sistemas de los mercados industriales y de consumo [3]. El software embebido puede ejecutar funciones específicas, como por ejemplo el control de las teclas de un horno de microondas, o suministrar una función significativa y con capacidad de control, funciones digitales en un automóvil, tales como el control de la gasolina sistema de frenado, entre otras.

Hay varias definiciones de software embebido. Las más tradicionales lo definen como procesamiento de información que está integrado con procesos físicos [4]. Otra definición es software que se ejecuta en dispositivos distintos de una computadora personal o un servidor de cómputo [4]. La primera definición es relevante para entender los retos a los que se enfrenta el desarrollador de software embebido, y la segunda para entender la oportunidad de negocio que representa el mercado para dispositivos. La principal regla del software embebido es interactuar con el mundo físico, mientras que en el desarrollo de aplicaciones empresariales, se busca olvidarse del mundo físico y enfocarse en abstracciones como entidades de información y proceso de negocios, en el desarrollo del software embebido sucede todo lo contrario. Su papel principal no es la transformación de datos, sino la interacción con el mundo físico. Se ejecuta en máquinas que no son computadoras, estas máquinas son automóviles, aviones, teléfonos, equipo del audio, robots, aparatos, juguetes, los sistemas de seguridad, armas, menús de televisiones, copiadoras, escáneres, clima, control de sistemas, sistemas industriales, entre otros [5].

Cualquier sistema eléctrico, mecánico o químico que incluya entradas, decisiones, cálculos, análisis y salidas es candidato para implementarse como un sistema embebido. Los sensores eléctricos, mecánicos y químicos reúnen información [6]. Las interfaces electrónicas convierten las señales de un sensor a una forma que acepta la microcomputadora. El software embebido efectúa las decisiones, los cálculos y los análisis necesarios, el software de los sistemas embebidos sólo resuelve una gama limitada de problemas [7], es por eso que el software de sistema embebido se le llama software de propósito específico.

Categorías de software embebido

Un sistema embebido está integrado por circuitos integrados programables, memoria flash o ROM, el correspondiente circuito impreso y el software embebido como parte esencial del mismo sistema, conocido en inglés como embedded software. El software se utiliza para controlar los productos electrónicos y usualmente se ejecuta sobre un microprocesador interno, un microcontrolador, un procesador digital de señal (DSP), una compuerta programable en campo (FPGA), un controlador lógico programable (PLC) y a veces en una PC de propósitos generales adaptada para fines específicos.

En el software embebido se pueden distinguir diferentes categorías: El original o básico, indispensable para el funcionamiento del aparato, el cual constituye su sistema operativo ad-hoc. El lenguaje de programación assembler, C/C++ o VHDL. Este tipo de software, de carácter eminentemente tecnológico, está incorporado en el aparato desde la salida al mercado de este último. En general, requiere un elevado esfuerzo de creación inicial, a cargo de mano de obra de muy alta especialización. Su costo se reparte en la cantidad de equipos entregados al mercado. En la mayor parte de los casos no se modifica a lo largo de su vida activa, siendo reemplazado al aparecer un nuevo modelo de producto. Un segundo tipo es el requerido por alguno de los equipos incluidos en el aparato anterior, pero que por condiciones específicas, como pueden ser requerimientos regulatorios de una determinada comunidad, modalidades o costumbres, se hace necesario adaptar el equipo original al uso específico requerido. Otro tipo es el dedicado a la supervisión y control de sistemas complejos cuyo funcionamiento implica la interrelación de gran cantidad de equipos o aparatos, por ejemplo el funcionamiento y control de los distintos procesos en la industria manufacturera [8].

Ámbitos de aplicación

Aplicaciones de sistemas embebidos con microcontroladores: los aparatos electrónicos, los sistemas de comunicación, los sistemas automotores, el equipo militar, las aplicaciones empresariales, los dispositivos médicos [7].

Sistemas mecatrónicos: La mecatrónica es una sinergia de sistemas mecánicos y eléctricos controlados por un sistema embebido.

La figura 1 muestra la relación que existe entre las diferentes disciplinas de la ingeniería involucradas en la mecatrónica, así como la interacción con un sistema embebido y dónde ubicaríamos la aplicación de un software embebido [9].

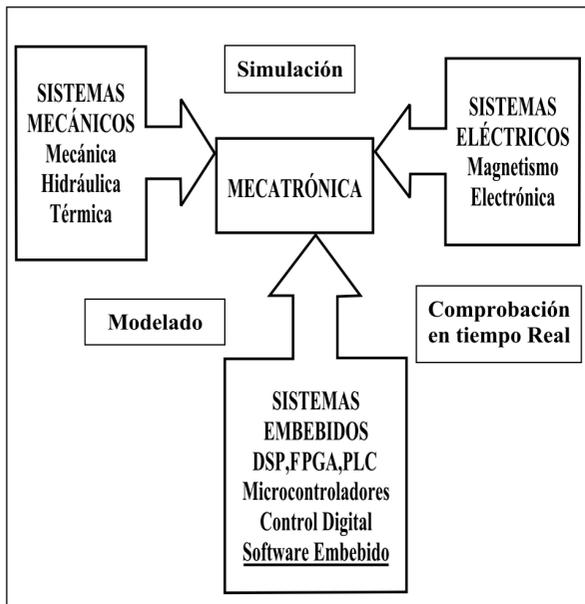


Figura 1. Relación de la Mecatrónica, sistemas relacionados y el software embebido.

Redes inalámbricas de sensores: Son pequeños dispositivos con poder de procesamiento y radio comunicación que con un par de baterías AA pueden operar por años sin mantenimiento alguno, además son lo suficientemente baratos como para integrarlos en televisores, modulares, lámparas, sensores entre otros productos, con la finalidad de habilitar el control y monitoreo remoto. Existen varias tecnologías propuestas para resolver el problema de comunicación inalámbrica en este tipo de redes, entre ellas se encuentra Zigbee [10].

Aplicaciones en FPGA (Field Programmable Gate Array): Una FPGA es un dispositivo semiconductor que contiene bloques de lógica cuya interconexión y funcionalidad se puede programar. El tamaño, estructura, número de bloques y conectividad de las conexiones varían en las distintas arquitecturas. Hay software especial para definir las conexiones de los switches y las funciones de las celdas lógicas. Existen ambientes integrados de trabajo como Xilinx ISE, Max Plus y Libero los cuales permiten desarrollar software embebido en estos dispositivos.

Estas son las principales aplicaciones de software embebido que podemos encontrar.

Características del software embebido

Los tres atributos que típicamente tienen consideraciones especiales en el desarrollo de software embebido son: confiabilidad, limitaciones en recursos de hardware y respuesta en tiempo real [4].

La figura 2 ilustra y explica dichos atributos.

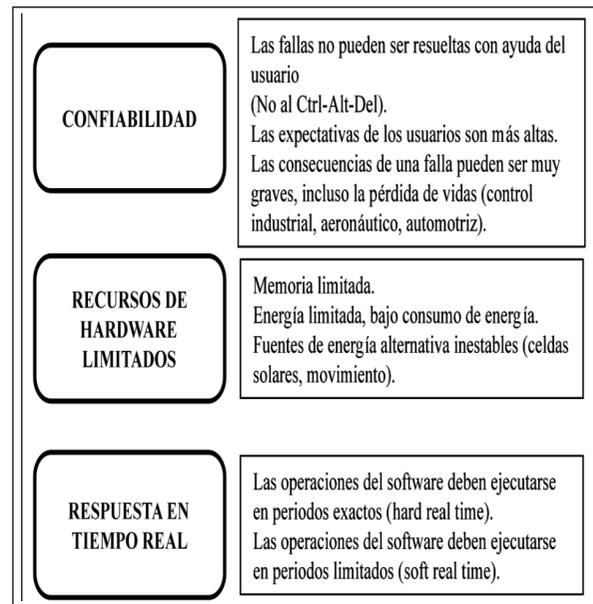


Figura 2. Atributos del software embebido

Retos al desarrollar software embebido

Su desarrollo presenta grandes retos debido a sus particularidades, el software embebido se preocupa por el mundo físico y por lo tanto se enfoca en problemas como medir el tiempo, ser capaz de detectar y responder a eventos en el ambiente, lidiar con restricciones físicas, así como dar respuestas en tiempo real [11]. Los clientes de estos sistemas desean obtener características como alto grado de reutilización, mantenibilidad y flexibilidad que históricamente no han sido prioridad en el desarrollo de software embebido. La posibilidad de combinar todos estos atributos presenta dificultades de ingeniería de software [4] todo esto en la búsqueda de mayores niveles de abstracción en el desarrollo de software embebido. Existen 4 grandes retos en su desarrollo: en la planeación del proceso de desarrollo de software embebido, en el establecimiento de una metodología, en el aseguramiento de la calidad del software embebido, en las herramientas de diseño [12].

Tendencias en el desarrollo de software embebido

Conforme el hardware continúa su mejora en la relación poder/precio, empieza a ser posible utilizar cada vez más hardware de propósito general en lugar de hardware dedicado para una solución específica. Cada vez son más los dispositivos embebidos que utilizan arquitecturas SoC (System-On-Chip) en lugar de circuitos integrados especializados.

El uso de hardware de propósito general simplificará enormemente la portabilidad de software embebido en diferentes dispositivos [4].

Para el año 2015, los sistemas computacionales en aplicaciones en paralelo se desarrollarán a nivel de los dispositivos individuales, con sistemas sobre chips, comenzarán a producirse en gran escala. Sensores y procesadores incluidos en sistemas mecánicos o biológicos, dispositivos ópticos, conexiones inalámbricas y reconocimiento de voz, son los futuros escenarios y tendencias de los sistemas y software embebido [12].

Conclusiones

La industria del software y sistemas embebidos es cada vez mayor y mantiene una creciente presencia en las diferentes actividades del ser humano. Las particularidades de este tipo de software son su alta confiabilidad en su operación, recursos de hardware limitado y respuestas en tiempo real, ha propiciado la búsqueda de mejores, métodos, técnicas, herramientas y proceso de desarrollo que garanticen productos de calidad y con una amplia gama de aplicaciones y desarrollos tecnológicos.

Los retos para desarrollar software embebido para los ingenieros de software y de hardware ha ocasionado que se amplíe la comunicación entre ambos dominios y más aún la aplicación de este tipo de sistemas involucra un trabajo interdisciplinario entre otras ramas de la ingeniería como la eléctrica, electrónica, mecánica, mecatrónica, biología entre otras.

El futuro y aplicación de este tipo de software es un desarrollo que requiere la búsqueda de mejores herramientas de diseño, ampliar la cobertura de aplicación y tendencia a mejorar las prácticas para su construcción

Referencias

[1] PC/1004 Embedded Consortium. "Specification-PC/104", http://www.pc104.org/pc104_specs.php, visitado el 18 de diciembre de 2009.

- [2] Galeano, Gustavo.(2009), *Programación de Sistemas Embebidos en C*, Alfaomega (México).
- [3] Pressman, Roger S. (2002), *Ingeniería de Software. Un enfoque práctico*, Mc Graw Hill, 5º ed. (España).
- [4] Obregón, Héctor. (2007), "Embedded Software Estado Actual y Tendencias". *SG Software Gurú Conocimiento en Práctica*. Septiembre – Octubre. Año 3. Núm 5. Naucalpan, México, pp 20-21.
- [5] Lee, Edward A. "Embedded Software". 2002. <http://ptolemy.eecs.berkeley.edu/publications/papers/02/embsoft/embsoftwre.pdf>, visitado el 4 de enero de 2010.
- [6] Reverter, Ferran y Pallás Ramón. (2009), *Circuitos de Interfaz Directa Sensor-Microcontrolador*, AlfaOmega (México).
- [7] Valvano, Jonathan. (2004), *Introducción a los sistemas de microcomputadora embebidos/ Introduction to Shrunk Microcomputer Systems*, Thomson. (México).
- [8] Cetinkunt, Sabri. (2007), *Mecatrónica*, Grupo Editorial Patria (México).
- [9] Lennon, Tony. (2009), "Diseño basado en modelos para sistemas mecatrónicos". *Mundo Electrónico*. Enero. Núm. 404. Barcelona, España, pp 34-35.
- [10] García, Christian.(2007),"Zigbee". *SG Software Gurú Conocimiento en Práctica*. Septiembre – Octubre. Año 3. Núm 5. Naucalpan, México, pp 22-24.
- [11] Miceli, Michael y otros, (2009), "System and Process for Monitoring Vehicle Fuel Level", U.S. Patent No.7587939
- [12] Ebert, Christof y Jones Capers. (2009), "Embedded software: facts, figures and future". *Innovative Technology for Computer Professionals Computer*. Abril. Volumen 42.Núm 4. Washington D.C, USA, pp 42-52.

Artículo recibido: 22 de enero de 2010

Aceptado para publicación: 30 de agosto de 2010