UNA METODOLOGÍA PARA EL DESARROLLO DE HARDWARE Y SOFTWARE EMBEBIDOS EN SISTEMAS CRÍTICOS DE SEGURIDAD

A. Perez, O. Berreteaga, A. Ruiz de Olano, A. Urkidi, J. Perez
Ikerlan S. Coop.
Apdo. 146 Pº. J. Mª. Arizmendiarrieta, 2
20500 Arrasate-Mondragón (Gipuzkoa)
{aperez, oberreteaga, arolano, aurkidi, jmperez}@ikerlan.es

RESUMEN

Viendo que la implantación de soluciones basada en sistemas electrónicos embebidos de alta confiabilidad va en aumento y los requisitos de los entornos a los que nosotros nos dirigimos son principalmente el coste, el tiempo y la confiabilidad, se observa la necesidad de una metodología, como herramienta, que facilite el desarrollo de sistemas electrónicos embebidos de alta confiabilidad.

En este trabajo se describe cómo el conocimiento de las distintas herramientas de desarrollo de cada etapa de diseño, así como la consideración de los distintos estándares de confiabilidad actualmente existentes, nos han llevado a establecer una metodología. La herramienta generada abarca aspectos tanto de Software (SW) como de Hardware (HW), porque pretende dar solución a problemas reales donde resulta ineludible la interacción de ambos. De esta forma, y apoyándonos en el ciclo de desarrollo clásico en V, veremos el conjunto de técnicas y métodos propuestos para mejorar la confiabilidad a lo largo de todo el ciclo de vida. Para validar dicha metodología, hemos realizado una experiencia práctica basada en un *Steer-by-Wire*, que es un sistema intuitivo y sencillo, y el sector al que pertenece (automovilístico) uno de los más exigentes en lo que respecta a coste y tiempo.

Finalizaremos discutiendo las mejoras obtenidas y la aplicabilidad en casos reales de diferentes dominios.

Palabras Claves: confiabilidad, sistemas embebidos, tolerancia a fallos, metodología.

1. INTRODUCCIÓN

Expansión de los sistemas electrónicos

La presencia de las tecnologías de la información con base electrónica sigue creciendo con ritmo acelerado, posibilitando múltiples funciones de automatización. Hoy en día, ámbitos tan cercanos para el ciudadano como la automoción, el transporte ferroviario, la elevación, los electrodomésticos o las comunicaciones los están integrando abundantemente en sus funcionalidades, lo que antes sólo había sido posible para entornos tan singulares como el militar o el aeroespacial. Esta abundancia creciente de aplicaciones dependientes del buen estado de salud de los dispositivos electrónicos que las posibilitan, requiere altos niveles de confiabilidad [1] en mayor número de ellos y sin precisar componentes especiales, sino manteniendo el uso de componentes de calidad comercial (Comercial-Off-The-Shelf: COTS). Éste es el ámbito que promueve la orientación de esfuerzos para conseguir *Sistemas Electrónicos Embebidos con alta Confiabilidad*.

Función y repercusión de los sistemas electrónicos embebidos sobre los ciudadanos

En sectores como la automoción, las comunicaciones o el ferroviario, aparte de la confiabilidad, priman el coste y el tiempo de comercialización (*time-to-market)*. Esta característica hace que se tengan que aceptar compromisos serios a la hora de realizar los diseños, de forma que no se puede pretender diseñar sistemas electrónicos embebidos con la misma metodología con la que se trabajaría en la industria aeronáutica o militar.

En este artículo se presenta la necesidad (para los desarrolladores de sistemas embebidos con alguna responsabilidad del tipo indicado) de crear una metodología propia, que tuviese ciertas características como la combinación de SW y HW de confiabilidad realzada, y dirigido a diversos campos de aplicación industrial donde priman las limitaciones de coste, tiempo y confiabilidad.

Con los criterios mencionados, y la colaboración de expertos de diversas especialidades (diseño, fabricación, prototipos, etc.), se ha definido una metodología, y para su validación se ha realizado una experiencia de desarrollo de un sistema de control piloto centrada en un caso de función *Steer-by-Wire*. La experiencia se ha desarrollado con un objetivo experimental y didáctico. Es una práctica en la que se han seguido los pasos de la metodología previamente diseñada, ejecutando cada fase y utilizando las técnicas sugeridas. La elección de este caso de aplicación se justifica por la clara visión de su requisito de alta confiabilidad de forma continuada.

2. UNA METODOLOGÍA, LA MEJOR HERRAMIENTA

Modelo en V del ciclo de vida

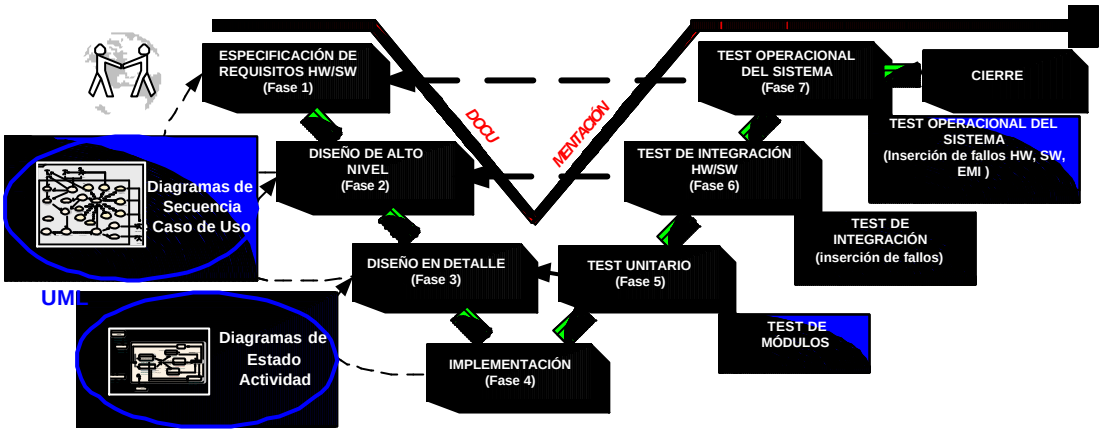

Figura 1: Modelo en V del ciclo de vida

El modelo en V define las siguientes etapas de desarrollo:
DEFINICIÓN DE ESPECIFICACIONES (Fase 1): Se deben definir y documentar los diferentes requisitos del sistema a desarrollar, identificando los valores numéricos más concretos posibles. Entre ellos debe estar la especificación del nivel de integridad, o SIL, en caso de ser requerido. Las posibles técnicas que se pueden utilizar en esta fase se mencionan en la tabla 1, cuya referencia se toma en el artículo [2].
DISEÑO GLOBAL (Fase 2): También llamado diseño de alto nivel. Su objetivo es obtener un diseño y visión general del sistema.
DISEÑO EN DETALLE (Fase 3): Consiste en detallar cada bloque de la fase anterior.
IMPLEMENTACIÓN (Fase 4): Es la fase en la que se materializa el diseño en detalle.
TEST UNITARIO (Fase 5): En esta fase se verifica cada módulo HW y SW de forma unitaria, comprobando su funcionamiento adecuado.
INTEGRACIÓN (Fase 6): En esta fase se integran los distintos módulos que forman el sistema. Como en el caso anterior, ha de generarse un documento de pruebas. Por una parte, se debe comprobar en todo el sistema el funcionamiento correcto, y por otra, en caso de tratarse con un sistema tolerante a fallos, debe verificarse que ante la presencia de un fallo persiste el funcionamiento correcto. Se comprueba el cumplimiento de los requisitos establecidos.
TEST OPERACIONAL DEL SISTEMA (Fase 7): Se realizan las últimas pruebas pero sobre un escenario real, en su ubicación final, anotando una vez más las pruebas realizadas y los resultados obtenidos.

Técnicas y medidas utilizadas en el ciclo de vida de seguridad: En cada una de las fases de desarrollo son de aplicación un conjunto de las técnicas y medidas que se listan a continuación:

Acción	Técnica	Fase
Gestión del proyecto	✍✍ Herramientas de gestión de proyectos de Ikerlan.	Todas
	✍✍ Definición de tareas y responsables.	Todas
	✍✍ Gestión del estado de las tareas.	Todas
Definición de requisitos, plan de seguridad y garantía de calidad	✍✍ Definición con UML, estructurando los requisitos por apartados.	1
	✍✍ Separación de requisitos de seguridad y los que no lo son.	1
	✍✍ Incluir requisitos normativos ambientales, vibraciones, EMI.	1
	✍✍ Análisis y revisión de especificaciones con UML.	1
	✍✍ Auditoria interna y externa.	1 y 2
	✍✍ Revisión tras cada cambio y etapa.	Todas
	✍✍ *Checklist.*	Todas
Documentación	✍✍ Metodología documentada.	Todas
	✍✍ UML considerado documentación.	Todas
	✍✍ Plantillas predefinidas (Revisión, *Checklist*, etc.).	Todas
	✍✍ *Checklist* de documentos.	Todas
	✍✍ Ayuda a la documentación del SW con UML.	Todas
Recursos	✍✍ Personal altamente cualificado.	Todas
	✍✍ Curso sobre el estándar (IEC61508) [3].	Previo

	✍ Curso sobre la metodología propuesta por Ikerlan. ✍ Grupos multidisciplinares.	Previo Todas
Codificación	✍ Aplicación de MISRA-C [4]. ✍ Medición de la complejidad del SW.	4 4
Implementación	✍ Guía de diseño del *layout*.	4
Verificación y Validación	✍ AMFE y FTA. ✍ Establecimiento de Plan de Validación. ✍ Simulación. ✍ Auditoria interna y externa. ✍ Revisiones. ✍ Autodiagnóstico. ✍ Test de caja negra. ✍ Test función a función y limites. ✍ Test funcional. ✍ Reutilización de módulos. ✍ Inyección de Fallos (físicas, EMI y SW). ✍ Incluir test normativos ambientales, vibraciones y EMI. ✍ Aplicación del Plan de Validación.	1, 2, 3 y 4 1 2 y 3 Todas Todas 2, 3 y 4 5 y 6 5 7 4 5 y 6 5 y 6 Todas
Instalación y Mantenimiento	✍ Personal altamente cualificado. ✍ Plan definido y amigable. ✍ Posibilidad de Inserción en caliente.	Posterior 2 y 3 6
Modificaciones	✍ Repetir todos los anteriores a los que afecte el cambio. ✍ Control de modificaciones y documentación de los mismos..	Todas Todas

Tabla 1 Fases, técnicas y acciones del ciclo de vida

Técnicas y medidas: En las fases de diseño e implementación se podrán utilizar las siguientes técnicas que se agrupan de acuerdo a los componentes que generalmente forman un sistema.

Relativo a:	Técnica
Arquitectura	✍ Redundancias hasta TMR del sistema global. ✍ Redundancia de subconjuntos. ✍ Uso de arquitecturas TTA. ✍ Diversificación HW y SW. ✍ Cálculos de fiabilidad del HW.
ECU	✍ Comparador. ✍ Votación por mayoría. ✍ Autotest del SW. ✍ Autotest del HW. ✍ Comparación recíproca por SW.
Intervalos de memoria invariable	✍ Cálculo del CRC de 2 palabras (16 bits). ✍ Replicación de bloques. ✍ Comparación recíproca.
Intervalos de memoria variable	✍ "Galpat" o "transparent Galpat". ✍ Duplicación de RAM.
Interfaces y unidades E/S	✍ Entradas multicanal. ✍ Salidas multicanal. ✍ Test de salidas (Lectura de la salida). ✍ No reconfigurables en marcha. ✍ *Test Pattern*.
Bus de datos	✍ Redundancia HW. ✍ *Test pattern*. ✍ Protocolo de transmisión. ✍ Redundancia de transmisión. ✍ Redundancia de información. ✍ No reconfigurable.
Alimentación	✍ Procedimiento *Power-Down*. ✍ Test de arranque. ✍ Protección de sobretensión. ✍ Detección de la caída de tensión. ✍ Alimentación N+1.
Watchdog	✍ *Watchdog* Externo con fuente de tiempos independientes. ✍ Control de la secuencia lógica del programa.
Reloj	✍ Sistemas con comprobación reciproca en sistemas redundados.
Comunicación	✍ Sistemas aislados de comunicación. ✍ Separación espacial de los buses de comunicación. ✍ Aumento de inmunidad de interferencia. ✍ *Test pattern*. ✍ Protocolo de transmisión. ✍ Redundancia de transmisión. ✍ Redundancia de información. ✍ No reconfigurable.

Tabla 2 Técnicas de Aplicación en el Modelo

3. EXPERIMENTO PARA LA VALIDACIÓN DE LA METODOLOGÍA

En este apartado se va a describir el proceso de la experiencia práctica desarrollada mediante el *Steer-by-Wire*.

Definición del *Steer-by-Wire*

El experimento que se ha desarrollado se basa en el control de la dirección de un automóvil de forma electrónica, es decir, sin transmisión mecánica directa. Se trata de una implementación práctica, sin intenciones de portarse a una aplicación real pero siguiendo la metodología propia y el estándar IEC 61508.

Para implementar el sistema se ha elegido un *joystick* de entrada, un servo como sistema actuador (ataca mediante una señal PWM) y un sistema de control distribuido. El sistema de control tiene unos módulos sensores encargados de recibir las consignas del *joystick*, procesarlas, generar la consigna equivalente para el servo y enviarla a la parte de actuación por medio del bus TTCAN replicado (tolerante a un fallo).

Objetivos: Los objetivos impuestos a la hora de realizar la experiencia han sido la de validar la metodología, ver los resultados de su aplicación y ratificar que lleva a diseños más confiables en menor tiempo y coste que utilizando métodos no orientados a sistemas confiables.

Aplicar arquitecturas TTA para su mayor conocimiento, dada la expansión que han tenido en aplicaciones de requisitos de confiabilidad con sistemas embebidos.

Profundizar en el estudio de cálculo de la tasa de disfunción [5, 6] y ver su interpretación así como las exigencias de la norma IEC 61508.

Desarrollo del *Steer-by-Wire*

Recogida de especificaciones: En el *Steer-by-Wire* se ha realizado de forma estructurada siguiendo el siguiente esquema jerárquico. Esta técnica se describe en la IEC 61508-7B.2.1. La recogida de especificaciones se ha llevado a cabo mediante grupos multidisciplinares: uno dedicado a HW y otro a SW.

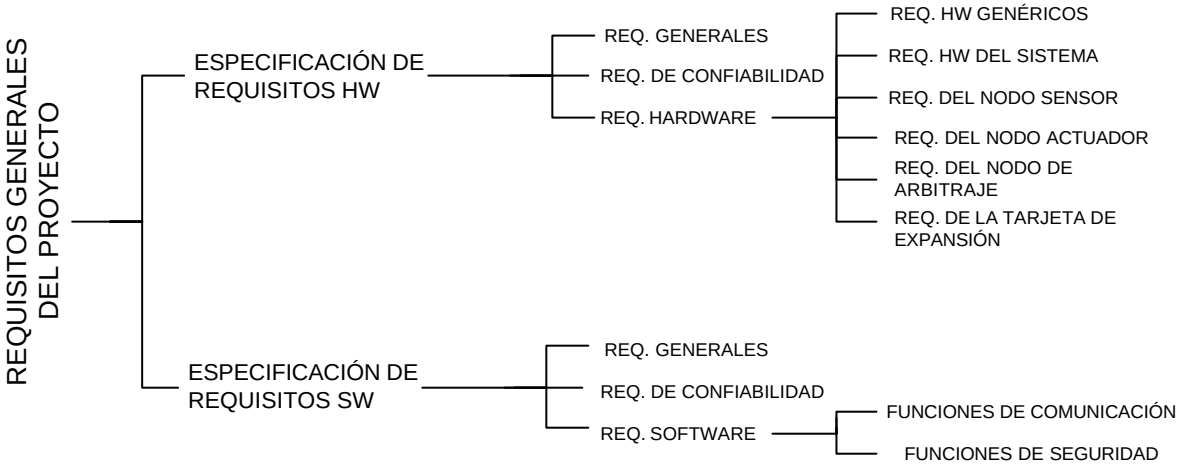

Figura 2 Jerarquía de especificaciones

Diseño global: Se ha seguido un diseño estructurado para generar un diseño global en bloques. También se han seguido criterios de modularidad para simplificar el diseño, y se han empleado componentes ya conocidos para minimizar el riesgo de la ocurrencia de fallos no detectados o desconocidos.

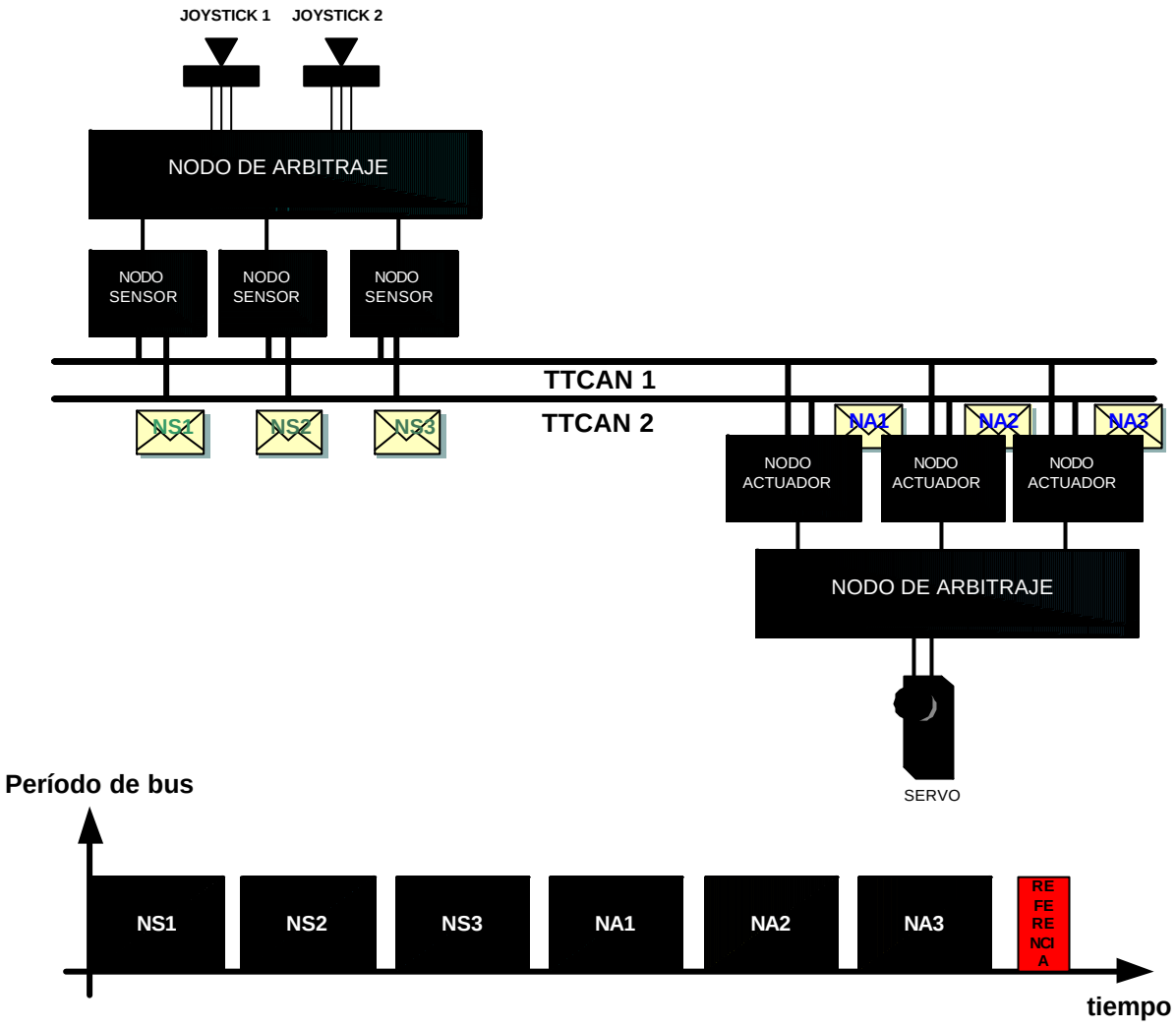

Figura 3 Diseño Global del *Steer-by-Wire*

Para identificar los puntos más débiles (o críticos) del sistema se ha realizado un AMFE tanto del SW como del HW. Teniendo en cuenta los requisitos de la aplicación y los resultados del AMFE, el diseño global obtenido para el sistema es el que muestra la figura 3. La arquitectura presenta redundancia triple tanto en sensorización como en actuación, un bus TTCAN duplicado y la entrada de *joystick* replicado. Esta arquitectura hace posible que el sistema siga funcionando de forma correcta aún en presencia de un fallo.

Diseño en detalle: Tras validar el diseño global, se ha iniciado la fase de diseño en detalle, donde se han utilizado diversas técnicas, como redundancia TMR del sistema de sensorización y actuación, redundancia de subconjuntos (alimentación, entradas, salidas, bus de comunicaciones, etc.), uso de arquitecturas TTA [7, 8] en el bus de comunicaciones implementando bus TTCAN, cálculos de la fiabilidad del HW utilizando Möbius, votación por mayoría para el caso de la sensorización y actuación tanto por HW como por SW, autotest del HW verificando las salidas de excitación en cada rutina de ejecución, autotest del SW comparando los valores generados en cada rutina de ejecución, replicación de bloques SW para comparar los resultados obtenidos, entradas multicanal, salidas multicanal, test de salidas mediante realimentación HW, *test pattern* en el bus local, procedimientos P*ower-Down*, test de arranque, protección de sobretensión, detección de la caída de tensión, alimentación N+1, *watchdog* externo con fuente de tiempos independiente, controles de secuencia lógica de programa.

Implementación: En esta fase se han materializado los PCB e implementado el SW en base al estándar MISRA-C a partir de los diagramas diseñados en UML.

Verificación Unitaria: En esta fase se ha verificado independientemente cada módulo del sistema, comprobando su correcta implementación, coherencia y cumplimiento de las especificaciones respectivas. Se han realizado revisiones del código, revisiones del PCB y pruebas de uso.

Test de integración: En esta fase se ha validado y verificado el correcto funcionamiento del sistema una

vez integrados todos sus componentes. Para su validación se ha realizado experimentos de inserción de fallos básicos, como fallos en la alimentación, fallos de conexión al bus y fallos de entrada / salida.

Test de campo: Al tratarse de un proyecto didáctico y experimental, y no estar prevista su integración real en un automóvil, la realización del test de campo especificado en la metodología no procede.

Fruto de la experiencia realizada se ha obtenido un demostrador *Steer-by-Wire* que tiene como característica propia la flexibilidad que aporta una arquitectura TMR, ya que sabiendo que el objetivo del experimento ha sido puramente didáctico, se puede transformar en 1oo1D o 1oo2D viendo así las ventajas de confiabilidad y coste que aporta cada arquitectura.

4. CONCLUSIONES

Para el desarrollo de sistemas de alta confiabilidad es imprescindible seguir una metodología en la que se defina cada paso, se utilicen una serie de herramientas de análisis, arquitecturas especiales, etc., para obtener diseños realmente confiables con unas características de coste y tiempo aceptables, y donde el esfuerzo se dirija tanto al HW como al SW, ya que un mundo no puede convivir sin el otro.

Una de las características más atractivas de la metodología propuesta es que nos lleva a invertir más tiempo en la etapa de diseño, fase en la que los rediseños son económicamente más baratos y los cambios más efectivos, de forma que se reduce el coste final del producto.

La aplicación de esta metodología tiene como paso intermedio el cálculo de la confiabilidad del sistema diseñado, y puede dar una orientación del tipo de arquitectura que se necesita para cumplir con el Nivel de Integridad necesario o exigido por el cliente.

La obtención de una certificación puede ser uno de los resultados de la aplicación de la metodología, siempre y cuando se cumplan las exigencias de la Norma y se documente cada paso del proceso.

Se ha comprobado que el seguimiento de la metodología implica un gran esfuerzo de diseño, haciendo más rápida la materialización del producto final con el cumplimiento de todos los requisitos.

5. LÍNEAS FUTURAS

Como principal línea de seguimiento está la aplicación de la metodología a futuros proyectos con requisito de alta confiabilidad.

Desde el punto de vista de trazabilidad, aunque actualmente se realice mediante *checklist*, se prevé la aplicación de herramientas estándares como *Doors*.

Se ve necesaria la ampliación de los test de verificación y validación integral, realizando nuevos experimentos de inyección de faltas mediante técnicas SWIFI e inyección de faltas mediante técnicas HWIFI (EMI) [9].

Por último, es muy interesante conocer las ventajas que pueden aportar las distintas configuraciones de arquitecturas en cualquier sistema, teniendo como referencia las exigencias de confiabilidad y coste del mercado.

6. REFERENCIAS

[1] A. Avizienis, J.C. Laprie, B. Randall, "Fundamental Concepts of Dependability", *UCLA CSD*, 2000, Report no. 010028.

[2] Ching-Yao Chan, Wei-Bin Zhang, El Miloudi El Koursi, Etienne Lemaire, "Safety Assessment of Advanced Vehicle Control and Safety Systems (AVCSS): A Case Study", *California PATH Research Report UCB-ITS-PRR-2001-30*, MOU 395 October 2001 ISSN 1055-1425.

[3] International Electrotechnical Commision, "IEC 61508: 2000 Parts 1-7, Functional Safety of electrical/electronic/programmable electronic safety-related systems", 2000.

[4] "MISRA-C: 2004, Guidelines for the use of the C language in critical systems", first published october 2004 by MIRA Limited, www.misra-c.com, ISBN 0 9524156 4 X PDF.

[5] Robin A. Sahner, Kishor S. Trivedi, Antonio Puliafito, "Performance and Reliability analysis of computer systems, An Example-Based Approach Using the SHARPE Software Package ", Ed. Kluwer Academic Publishers, fourth printing 2002, ISBN 0-7923-9650-2.

[6] Dhiraj K. Pradhan, "Fault-tolerant Computer System Design", Mary Franz, ed. Prentice Hall PTR, 1996, ISBN 0-13-057887-8.

[7] H. Kopetz, "The Time-Triggered Model of Computation", *Proceedings of the 19th IEEE Systems Symposium (RTSS98)*, December 1998, Technical University of Vienna, Austria.

[8] Dr. Markus Plankensteiner, "Comparison TTP, TTCAN, FlexRay", TTTech Computertechnik AG, 2003, www.tttech.com.

[9] J.C. Baraza, "Contribución a la Validación de Sistemas Complejos Tolerantes a Fallos. Nuevos Modelos de Fallos y Técnicas de Inyección de Fallos", Tesis doctoral, Departamento de Informática de Sistemas y Computadores, Universidad Politécnica de Valencia, Octubre 2003, ISBN 84-688-4048-3.