

Tema I

Introducción

En este tema vamos a tratar de introducir al alumno en los denominados sistemas empotrados. En este tema introduciremos una posible definición de estos sistemas, así como una posible clasificación, un flujo de diseño típico, y una serie de aplicaciones reales de estos sistemas.

1.1. Introducción. Definiciones

El interés por los sistemas, que hoy día se denominan empotrados (embedded), ha sufrido un aumento considerable en la última década con su aplicación a los sistemas denominados de tiempo real. No obstante, podemos encontrar referencias a este tipo de sistemas que datan de la época de los 70.

Pero antes de entrar en materia, debemos dar una definición de los sistemas que vamos a tratar. En primer lugar debemos indicar que no existe ninguna definición estándar de lo que se entiende por sistema empotrado; es más, dependiendo del área en el que se le pregunte nos darían respuestas diferentes. Así, nos podemos encontrar con las siguientes definiciones:

- un sistema empotrado es un sistema operativo ejecutándose en un micro de pocos recursos.
- un sistema empotrado es un artefacto (hardware + software) no susceptible de modificación del algoritmo que define su comportamiento.
- un sistema empotrado es un procesador, con sus elementos externos que desarrolla una función específica de manera autónoma.
- un sistema empotrado es un sistema computador de propósito especial construido en un dispositivo mayor.
- un sistema empotrado es una mezcla de hardware y software que constituye un componente dentro de un sistema más complejo y se espera que funcione sin intervención humana.

Así, nosotros vamos a considerar una definición que englobará a la mayoría de las definiciones anteriores. La definición que vamos a manejar es la siguiente:

Un sistema empotrado es un sistema que usa un computador para realizar una función específica, pero ni es usado ni es percibido como un computador.

Por lo tanto, un sistema empotrado tiene como integrante fundamental al computador, ya sea en sus versiones de microprocesador o de microcontrolador. Lo esencialmente importante es que su funcionamiento se lleve a cabo mediante programación. Pero como el componente computador no se puede ver como tal, se debe incluir los datos de entrada, y obtener los datos de salida, de un modo no convencional. Esta situación implica que al sistema computador hay que añadirle componentes hardware para tal efecto. Por lo tanto, un sistema empotrado genérico seguirá el esquema mostrado en la figura 1.1.

Luego un sistema empotrado es un sistema con un relativo grado de heterogeneidad por la combinación de hardware a medida (como es el dispositivo ASIC) y software empotrado (como son los programas que controlan tanto al microcontrolador como a los DSP o elementos programables). Incluso, dentro de cada uno de estos grandes grupos, existe heterogeneidad ya que los dispositivos ASIC utilizarán diferentes estilos de diseño (por lo general); y en cuanto al componente software, habrá diferencia de estilos en la programación del microcontrolador y de los DSP.

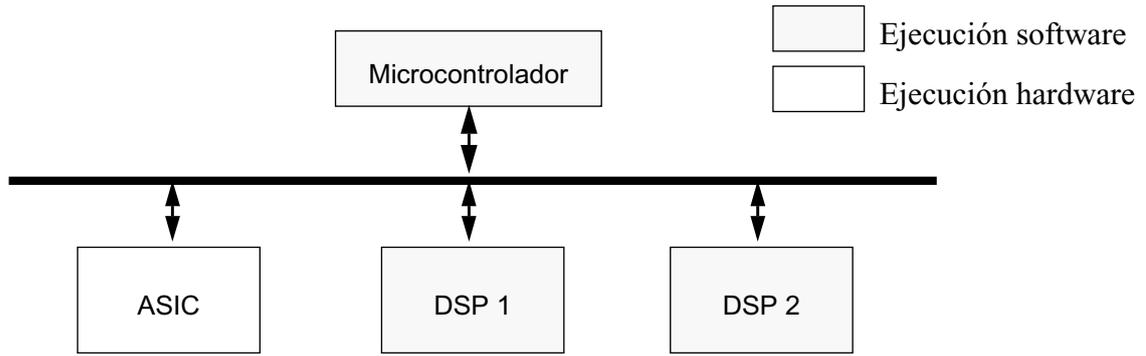


Figura 1.1.- Arquitectura típica de un sistema empotrado

Esta heterogeneidad impone una de las principales partes del diseño de cualquier sistema empotrado: decidir qué funcionalidad se implementará vía software y cuál vía hardware. De hecho esta decisión impondrá la arquitectura del sistema completo, ya que nos indicará los diferentes componentes que son necesarios desarrollar.

Veamos un ejemplo más detallado, como el que se muestra en la figura 1.2. En este sistema podemos apreciar la CPU y la memoria, junto con una amplia variedad de interfaces que permite al sistema medir, manipular e interactuar con el entorno exterior. En esta figura podemos ver que existen dos tipos de comunicación: una comunicación en el interior del sistema (que se realizará a través de un bus de sistema), y una comunicación externa (del sistema con el exterior).

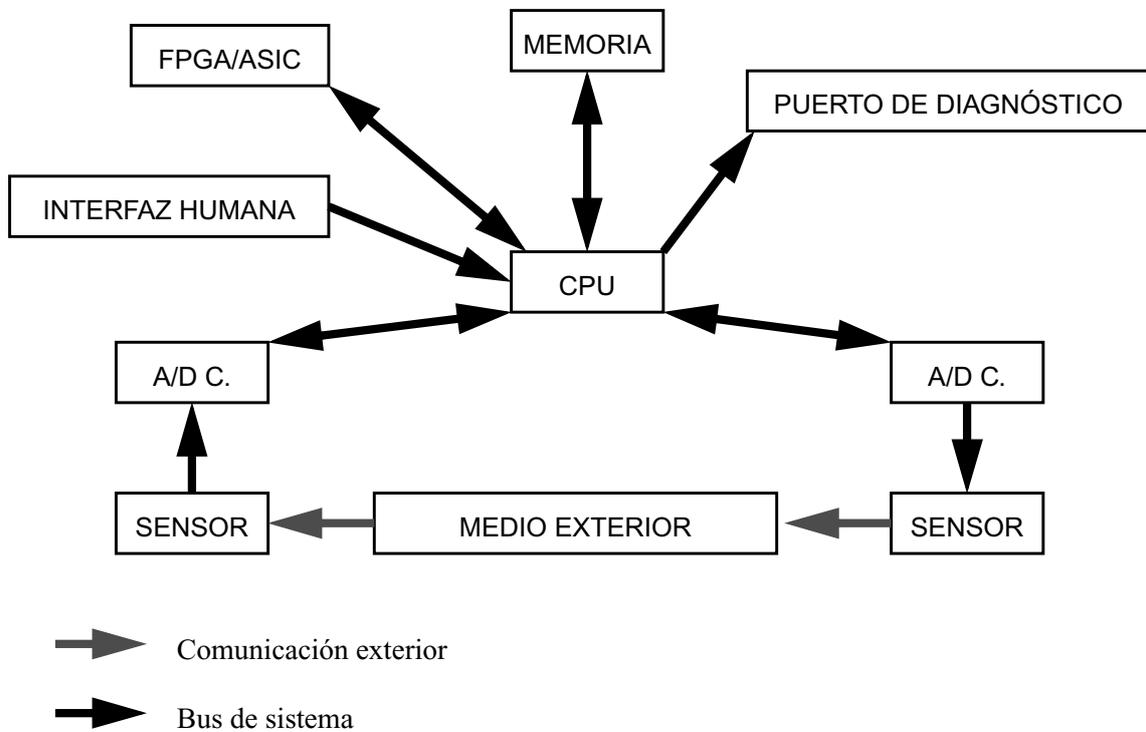


Figura 1.2.- Arquitectura típica de un sistema empotrado

Algunas diferencias con un sistema computador pueden ser las siguientes:

- La interface humana (que en un sistema computador es generalmente la pantalla y el teclado) puede ser tan simple como un diodo led o tan compleja como el sistema de visión de un robot.
- El puerto de diagnóstico ya no sirve únicamente para chequear el sistema computador (o, extrapolando, el sistema empotrado), sino que también puede chequear el entorno que hay que controlar.
- Se suelen utilizar dispositivos de aplicación específica para aumentar las prestaciones del sistema completo, ya sean digitales o analógicos.
- Por lo general, el software tiene una misión fija y específica a la aplicación.

Por lo tanto, la mayoría de estas diferencias vienen determinadas por las diferencias en los campos de aplicación. Mientras que en el sistema computador, las principales tareas serán la ejecución de hojas de cálculo, procesadores de textos o análisis mediante simulación; en los sistemas empotrados se realizarán tareas de ejecución de leyes de control, máquinas de estado finito y algoritmos de procesamiento de señal para poder realizar la manipulación del entorno.

Las principales características de un sistema empotrado son las siguientes:

- Estos sistemas emplearán una combinación de recursos hardware y software para realizar una función específica.
- Estos sistemas realizan un única función o un conjunto muy limitado de funciones (no suelen ser de propósito general)
- La potencia, el coste y la realizabilidad suelen ser los principales factores de coste.
- El diseño de procesadores de aplicación específica suelen ser un componente significativo de estos sistemas.

Aunque todas las características anteriores son inherentes a cualquier sistema empotrado, para que el sistema sea útil, también deberá tener las siguientes:

- Concurrencia. Los componentes del sistema funcionan simultáneamente, por lo que el sistema deberá operar a la vez.
- Fiabilidad y seguridad. El sistema debe ser fiable y seguro frente a errores, ya que puede requerir un comportamiento autónomo. El manejo de estos errores puede ser vía hardware o software; aunque la utilización software nos dará un sistema menos robusto.
- Interacción con dispositivos físicos. Los sistemas empotrados interaccionan con el entorno a través de dispositivos E/S no usuales, por lo que suele ser necesario un acondicionamiento de las diferentes señales.
- Robustez. El sistema empotrado se le impondrá la necesidad de la máxima robustez ya que las condiciones de uso no tienen porqué ser “buenas”, sino que pueden estar en el interior de un vehículo con diferentes condiciones de operación.
- Bajo consumo. El hecho de poder utilizar el sistema en ambientes hostiles puede implicar la necesidad de operaciones sin cables. Por lo tanto, un menor consumo implica una mayor autonomía de operación.

- Precio reducido. Esta característica es muy útil cuando estamos hablando de características de mercado. Esta situación no es nada inusual en el campo de los sistemas empotrados ya que tienen una gran cantidad de aplicaciones comerciales, tanto industriales como de consumo.
- Pequeñas dimensiones. Las dimensiones de un sistema empotrado no dependen sólo de sí mismo sino también del espacio disponible en el cual dicho sistema va a ser ubicado.

En este tipo de diseño son de particular importancia los parámetros de mercado, caracterizado en su máximo nivel por la característica de precio reducido. Este hecho es debido a que los sistemas empotrados surgieron antes como un elemento de mercado (gracias al auge de los circuitos integrados) que como una disciplina académica. Prueba de ello es que existan en el mercado tantos sistemas empotrados en menos de veinte años de disciplina. Como ejemplo, mostramos en la figura 1.3 los tipos de sistemas empotrados que podemos encontrarnos en un hogar medio. En ella podemos distinguir cuatro grandes grupos de elementos: los elementos basados en PC (como pueden ser el ordenador, un PDA o cualquier elemento de entrada/salida como una impresora); los elementos relativos a las comunicaciones (como pueden ser los teléfonos o intercomunicadores); los elementos relativos al entretenimiento (como pueden ser los televisores, videos y DVD); y los elementos relativos al confort y comodidad (como pueden ser los electrodomésticos y climatizadores).

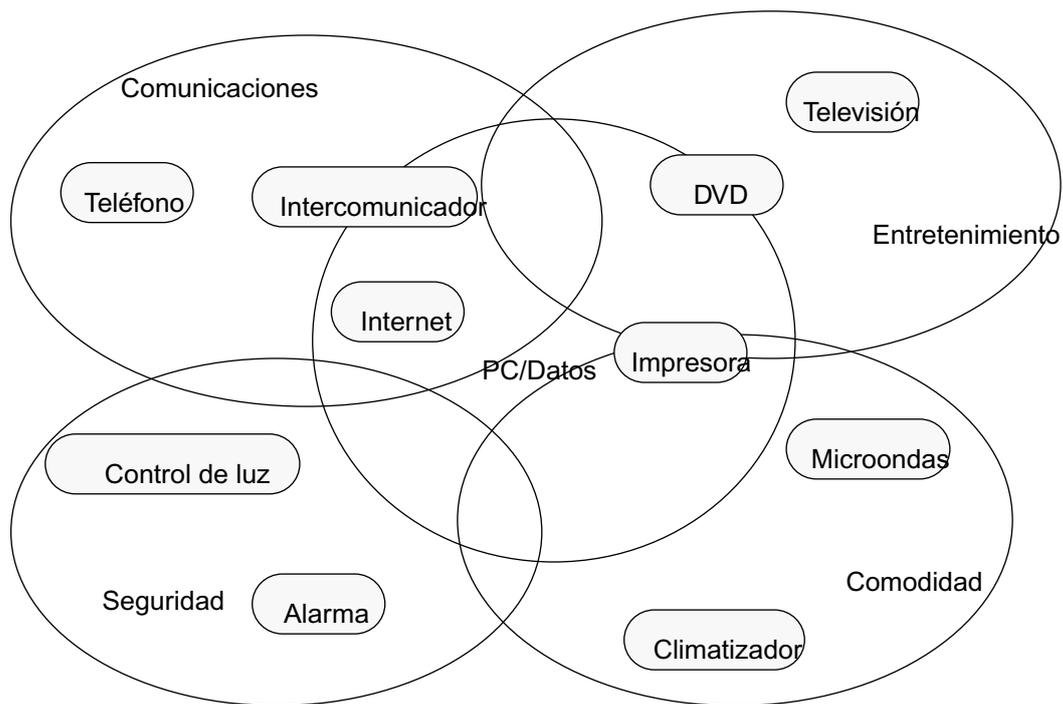


Figura 1.3.- Sistemas empotrados en un hogar medio.

1.2. Clasificación de sistemas empotrados

Dentro de la definición expuesta anteriormente, una gran cantidad de dispositivos pueden ser considerados como sistemas empotrados. Podemos encontrar diferentes clasificaciones

de este tipo de sistemas. Una de las principales se puede encontrar si consideramos su interacción con el resto del entorno. Atendiendo a esta característica, podemos encontrar los siguientes sistemas:

- **Sistemas reactivos.**- son aquellos sistemas que siempre interactúan con el exterior, de tal forma que la velocidad de operación del sistema deberá ser la velocidad del entorno exterior.
- **Sistemas interactivos.**- son aquellos sistemas que siempre interactúan con el exterior, de tal forma que la velocidad de operación del sistema deberá ser la velocidad del propio sistema empotrado.
- **Sistemas transformacionales.**- son aquellos sistemas que no interactúan con el exterior, únicamente toma un bloque de datos de entrada y lo transforma en un bloque de datos de salida, que no es necesario en el entorno.

Dentro de los sistemas reactivos podemos incluir el sistema de control aérea de un aeropuerto, ya que la velocidad del sistema dependerá de la velocidad con la que lleguen los datos de los diferentes aviones que se acerquen o salgan del mismo. En cuanto a los sistemas interactivos, podemos incluir a cualquier tipo de máquina de videojuegos, ya que la velocidad del sistema depende de él mismo, y el exterior (es decir el usuario del videojuego) se debe adecuar a su velocidad. Por último, dentro de los sistemas transformacionales podemos incluir a los postes de publicidad electrónicos, en los que no existe ningún tipo de interactividad excepto la entrada de datos iniciales y la salida de datos finales.

Aunque estos tres tipos de sistemas cumplen con la definición de sistema empotrado, se suelen tomar como tal a los sistemas reactivos, ya que su auge surgió cuando se adaptó estos diseños a este tipo de problemas. De hecho, los sistemas reactivos son más comúnmente conocidos como sistemas de tiempo real.

1.3. Flujo de diseño de Sistemas Empotrados

Como en cualquier tarea de diseño, el llevar a buen término esta tarea implica la consecución de un conjunto de subtareas, las cuales son conocidas comúnmente como flujo de diseño.

No obstante, antes de mostrar el flujo de diseño, veamos los niveles de abstracción que vamos a recorrer en el flujo de diseño; puesto que la tarea de diseño no es más que pasar de una descripción de abstracción de mayor nivel a otra de menor nivel. En la tabla 1.1 vemos los diferentes niveles de abstracción de cualquier diseño de circuitos electrónicos.

Según la tabla anterior, los sistemas empotrados se encuentran en el nivel de sistema. En este nivel, los componentes que forman al sistema son procesadores y memorias. No obstante, no debemos generalizar a los procesadores como procesadores de propósito general, sino como cualquier dispositivo capaz de procesar información, es decir, cualquier dispositivo de aplicación específica también estaría dentro de estos elementos como vimos en la figura 1.1. Y la tarea de diseño debe acabar en la descripción física del nivel de dispositivo. No obstante, la descripción final no tiene porqué ser un único circuito integrado monolítico (como sucede normalmente en Microelectrónica), sino que puede ser una tarjeta impresa, o un dispositivo MCM (modulos multi-chip).

Nivel de abstracción	Comportamiento	Estructura	Físico
Nivel de sistema	Procesos Prestaciones	Procesadores Memorias	Particiones básicas
Nivel algorítmico	Algoritmos	Unidades funcionales	Particiones básicas
Nivel de transferencia de registros	Transferencia de registros	Unidades funcionales Registros	Macroceldas
Nivel lógico	Ecuaciones booleanas Transiciones de estados	Puertas lógicas Latches	Celdas básicas
Nivel de dispositivo	Ecuaciones eléctricas	Transistores Conexiones	Metal Polisilicio Difusión

Tabla 1.1. Niveles de abstracción de circuito electrónicos.

Un flujo de diseño bien establecido para los sistemas empotrados es el mostrado en la figura 1.4. Para ir pasando de etapa en etapa hay que realizar una serie de subtareas. En primer lugar se debe realizar un modelado del sistema completo. El modelo obtenido de esta subtarea servirá para fijar los diferentes algoritmos que están involucrados en la operación del sistema (ya sean algoritmos de control o de operación con datos). Un modelo formal debería contar con los siguientes componentes: una descripción funcional; un conjunto de propiedades; un conjunto de índices de desarrollo; y un conjunto de restricciones.

La siguiente subtarea es el particionado del sistema. Una vez que haya concluido, tendremos el comportamiento del sistema dividido en varias funciones de menor complejidad, con el propósito de abordar el problema completo como varios problemas desconectados entre sí. Seguiremos con el particionado hardware/software, subtarea que determinará qué funciones se realizarán mediante dispositivos de aplicación específica (hardware) o mediante programación (software). Las funciones realizadas mediante programación no implica que deba realizarse en la CPU, sino que lo hará un dispositivo capaz de entender e interpretar un programa, como puede ser un DSP.

Una vez que se ha realizado la separación, se debe realizar la tarea de síntesis de cada una de las partes. Estos procesos de síntesis no deben realizarse de forma separada ni independiente, sino que hay que considerar una tarea de scheduling, en la cual se realizará una temporización de todos los dispositivos para que el flujo de datos esté de acuerdo con la operación a realizar.

Finalmente habrá que realizar una subtarea de mapeado que traducirá la descripción funcional en programas que pueden ser ejecutados en unos determinados procesadores o en una conexión de dispositivos hardware digitales y/o analógicos (dependiendo de la zona de particionado de la descripción funcional).

Aunque se ha tratado como un flujo secuencial de tareas, esta situación no se adapta a la realidad. Realmente, todas las subtareas están relacionadas entre sí, y no únicamente las adyacentes. Por ejemplo, el hecho de tener un determinado procesador con sus propias funciones (estaríamos hablando de la subtarea de mapeado), puede causar que el comportamiento global

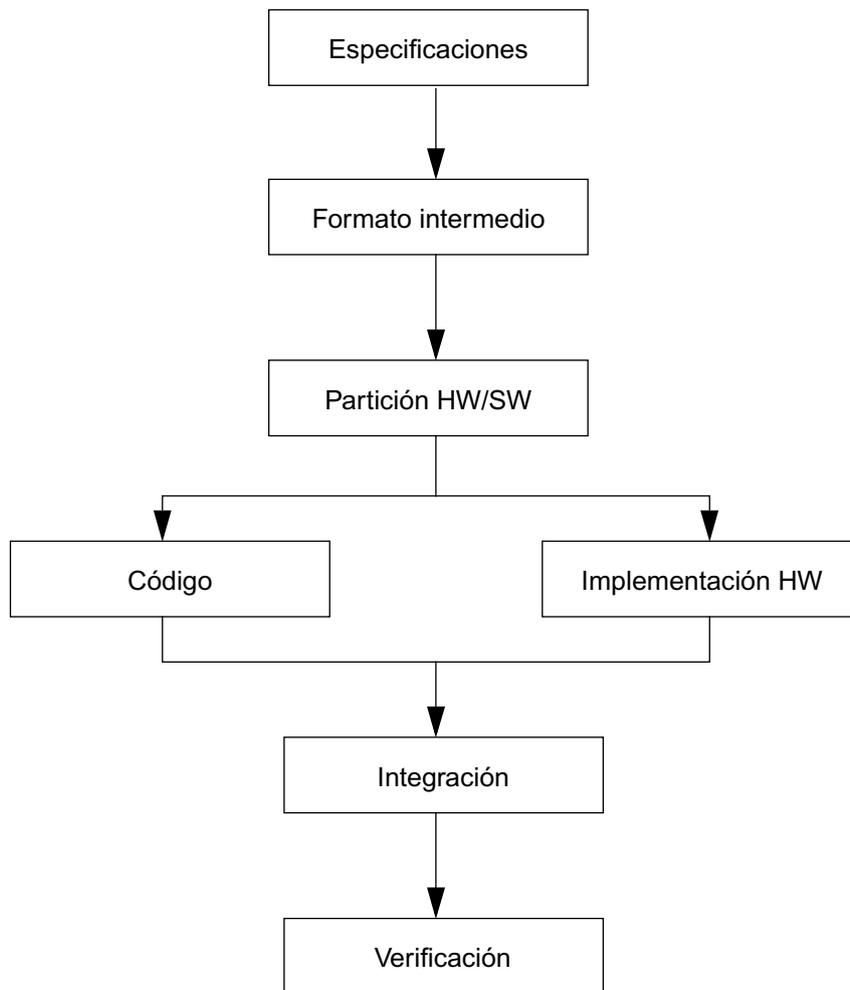


Figura 1.4.- Flujo de diseño de un sistema empujado.

sea particionado en una serie de funciones (subtarea de particionado), que podrían cambiar si se dispusiese de otro tipo de procesador.

1.3.1. Codiseño hardware/software

El término de codiseño hardware/software es relativamente reciente, de principios de la década de los 90. Este término describe la confluencia de dos problemas en el diseño de circuitos integrados:

- Las metodologías de diseño de sistemas tendrán que considerar grandes CPUs prediseñadas, como componentes de los circuitos.
- El software debería ser tratado como un componente en el diseño de circuitos, el cual estaría incluido en las CPUs mencionadas anteriormente.

Esta situación fue predicha según la ley de Moore que predecía que en esta fecha, los circuitos integrados serían lo suficientemente grandes para poder albergar una CPU y otros sistemas.

Por lo tanto, el codiseño hardware/software puede ser definido de la siguiente forma:

El codiseño hardware/software es la tarea de diseñar el sistema hardware y la generación de código del sistema software de un sistema mixto, de tal forma que el comportamiento del sistema global esté formado por la ejecución del sistema hardware y software

Es decir, es una de las tareas del flujo de diseño de sistemas empotrados mostrados en el apartado 1.3. No obstante, se le suele igualar al problema de diseño completo relegando las tareas de modelado y particionado a papeles secundarios.

1.3.1.1. Historia de los sistemas empotrados

Cuando los sistemas empotrados entraron en el rango de disciplina académica, se vio que uno de los grandes problemas radica en el particionado hardware/software; y para dar solución a este problema era necesario la creación de modelos y algoritmos. Al principio de la década de los 90, se presentaron dos sistemas para dar solución a este problema, denominados VULCAN y COSYMA, los cuales tomaron aproximaciones complementarias para realizar este particionado, es decir, para decidir qué funciones serían realizadas vía software, y cuáles vía hardware. Mientras que VUCLAN colocaba todas las funciones en los dispositivos hardware, e iba pasando algunas funciones a software para minimizar el coste; COSYMA colocaba todas las funciones vía software e iba pasando funciones a los dispositivos hardware para obtener las prestaciones requeridas. Los diseñadores de los sistemas hardware/software tenían que analizar las prestaciones en tres dimensiones: hardware, software y sistema:

- Las prestaciones hardware tenían como meta determinar la máxima frecuencia de reloj de la unidad hardware.
- Las prestaciones software tenían como meta determinar el tiempo de ejecución del caso peor (en esencia, el mismo problema que en el caso hardware); no obstante su solución es más difícil de hallar debido a la menor investigación desarrollada en este campo.
- Las prestaciones de sistema también son complicadas, ya que varios programas pueden ser ejecutados de forma concurrente (o varias partes de varios programas), que a su vez, pueden ser ejecutados concurrentemente con el procesado de aplicaciones específicas. Ambos sistemas (VULCAN y COSYMA) tratan de reducir la complejidad, evitando estas concurrencias, y por lo tanto, todas las operaciones (software y hardware) son consideradas como secuenciales.

Para realizar estos análisis de prestaciones se encontró en la cosimulación un aliado muy importante, incorporándolo como elemento esencial a la metodología de codiseño. El reto consiste en la realización de una cosimulación en niveles de abstracción mixtos para ejecutar los suficientes vectores de test para validar el diseño. Uno de los cosimuladores que primero surgieron fue PTOLEMY.

Una vez que las prestaciones anteriores, tiempos de ejecución y de procesado, iban siendo resueltas; otras prestaciones iban adquiriendo una importancia creciente. Entre estas prestaciones podemos encontrar la estimación del coste hardware, el consumo de potencia, arquitecturas más generales, protocolos.

Actualmente, y gracias al auge de las FPGA, el particionado hardware/software es una tarea de diseño práctica. Varios fabricantes han desarrollado placas que combinan FPGA (en la que se implementaría los dispositivos hardware) y CPUs (en la que se implementaría el desarrollo software), conteniendo la arquitectura típica de un sistema empotrado.

1.4. Ejemplos de Sistemas Empotrados

Los sistemas empotrados están ampliamente presentes en nuestra vida cotidiana, y de hecho, raro es el momento en el que no nos encontramos cerca de uno de ellos.

- dentro de la electrónica de consumo podemos encontrar a las cámaras digitales (ya sean de fotos o de vídeos);
- dentro de los electrodomésticos podemos encontrar lavadoras, microondas; dentro del campo del automóvil podemos encontrar el sistema de frenado;
- dentro de las plantas industriales podemos encontrar cualquier controlador de una planta;
- dentro de los productos informáticos podemos encontrar las impresoras;
- dentro de los productos de comunicación podemos encontrar los faxes, los teléfonos.

Estos son algunos ejemplos de una serie de campos en los que los sistemas empotrados están muy consolidados, y algunos sistemas de ejemplo de cada campo. No obstante, también existen muchos otros campos, y otros muchos ejemplos de los campos anteriores y de otros más.

Un ejemplo típico de un sistema empotrado puede ser el sistema de navegación de un vehículo. Supongamos que tenemos un vehículo con tres sensores en la parte delantera, con un radio de acción de diez metros cada uno, tal como se muestra en la figura 1.5. Al sistema se le introduce el punto de inicio y el punto de destino, y debe ser capaz de sortear los posibles obstáculos que se encuentre en su trayectoria.

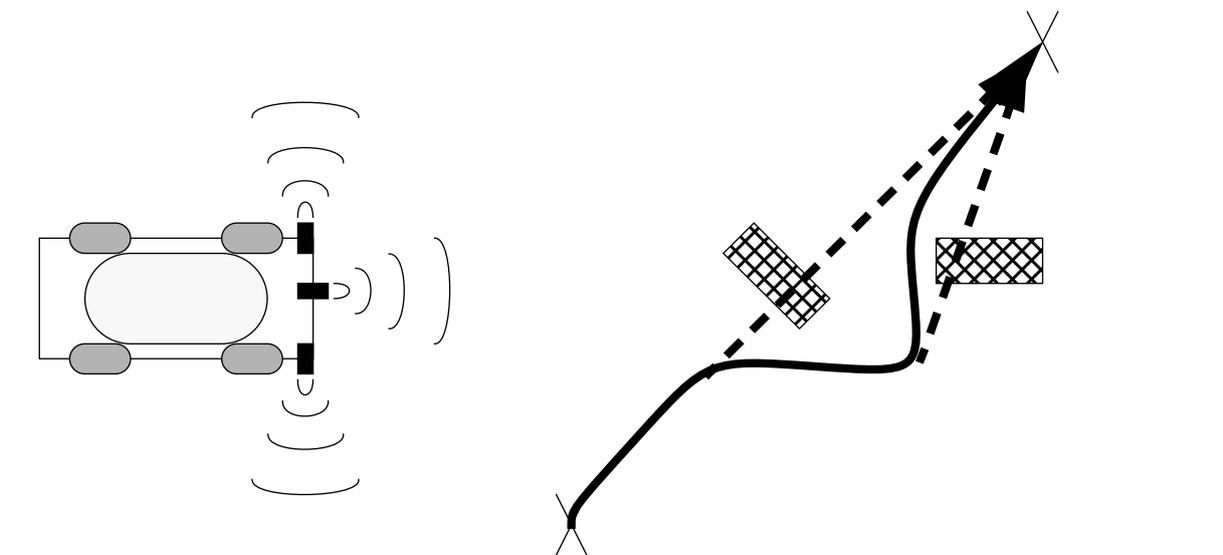


Figura 1.5.- Ejemplo del sistema de navegación de un vehículo autónomo.