

1. Conceptos básicos en los Sistemas de Tiempo Real

Contenido

1. CONCEPTOS BÁSICOS EN LOS SISTEMAS DE TIEMPO REAL	1
1.1 INTRODUCCIÓN	2
1.2 PROPIEDADES IMPORTANTES.....	4
1.3 CLASIFICACIÓN.....	4
1.4 JUSTIFICACIÓN.....	5
1.5 NECESIDADES DE LOS SISTEMAS DE TIEMPO REAL.....	5
1.5.1 <i>Especificación, análisis y verificación</i>	6
1.5.2 <i>Programación en tiempo real</i>	6
1.5.3 <i>Sistemas operativos de tiempo real</i>	6
1.5.4 <i>Arquitecturas de tiempo real</i>	7
1.5.5 <i>Comunicaciones de tiempo real</i>	8
1.6 ASPECTOS PARTICULARES	8
1.6.1 <i>Sistemas empotrados</i>	8
1.6.2 <i>Tolerancia a fallos</i>	9
1.7 EJEMPLOS	9

1.1 Introducción

Los ordenadores se han estado utilizando, y se usan cada vez más, para el desarrollo de aplicaciones en las que el tiempo es un factor muy importante para el correcto funcionamiento del sistema.

Estamos muy habituados a hacer programas en los que, a partir de unos datos de entrada y un algoritmo más o menos complejo, se obtienen unos datos de salida. Diremos que el programa funciona correctamente si los resultados son los esperados para los datos de entrada pero, a menudo, no se han impuesto restricciones de tiempo en la obtención del resultado. Normalmente, la única restricción de tiempo impuesta es no tener que esperar demasiado a obtener el resultado.

Los sistemas informáticos que interactúan con el mundo real, estarán sometidos a las restricciones de tiempo que imponga su entorno. Puede ocurrir que después de haber obtenido unos resultados correctos a partir de los datos de entrada, estos ya no sean útiles porque se han obtenido demasiado tarde.

Teniendo en cuenta esta circunstancia podemos formular las siguientes definiciones:

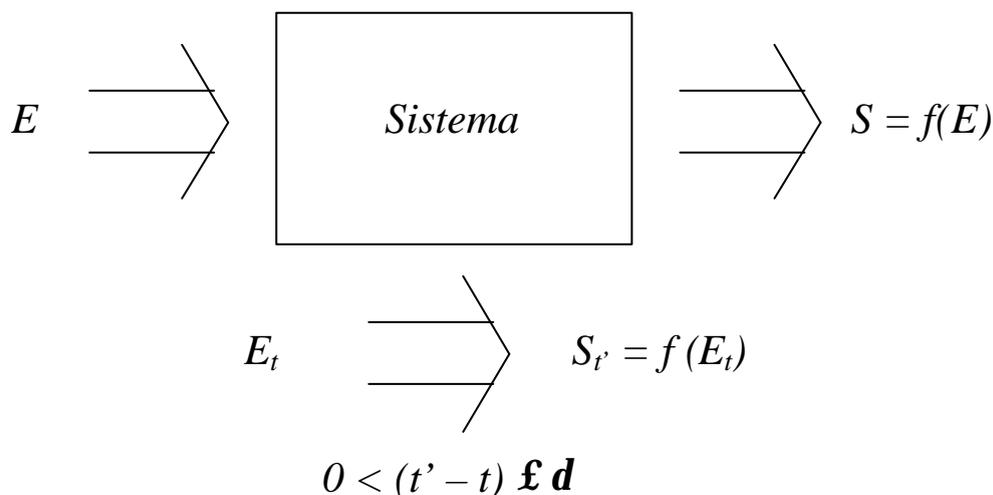
Definición:

Las aplicaciones de tiempo real son las aplicaciones informáticas en las que la obtención de los resultados está sujeta a unas restricciones temporales impuestas por el entorno en que se ejecutan.

Definición:

Los *sistemas* que soportan la ejecución de aplicaciones de tiempo real y aseguran el cumplimiento de los requerimientos temporales se denominan *sistemas de tiempo real*.

Un sistema de tiempo real es un sistema informático en el que es significativo el tiempo en el que se producen sus acciones. No es suficiente que las acciones del sistema sean correctas lógicamente, sino que, además, es necesario que se produzcan en el instante adecuado (antes de un tiempo máximo o deadline).



Esto es debido a que el sistema está conectado a un proceso externo del que se recibe estímulos a los que debe responder con suficiente rapidez para evitar que evolucione a un estado indeseable.

No se debe confundir este comportamiento con que la salida depende del tiempo en que se produce la entrada: $S = f(E, t)$

Unas características importantes de los sistemas en tiempo real son:

- Determinismo: Conocer exactamente como se comporta el entorno del sistema. Tener seguridad que no van a aparecer situaciones a las que el sistema no pueda responder adecuadamente.
- Comportamiento predecible. Saber como se comporta el sistema de manera que no pueda aparecer ninguna situación que altere el comportamiento temporal

Si los estímulos aparecen en instantes de tiempo no determinados, se tendrán que hacer unas asunciones para convertir nuestro sistema en predecible. Por ejemplo, suponer que entre dos estímulos consecutivos transcurre al menos un tiempo, o que en un determinado intervalo no ocurren más de N estímulos.

Algunas características de los sistemas informáticos que se suelen utilizar y que los hacen inadecuados para el tiempo real por hacerlos no predecibles son:

- Memoria caché y pipe-line: Una misma instrucción de código máquina puede ejecutarse en más o menos tiempo según la historia del programa.
- Interrupciones no controladas: El tratamiento de las interrupciones puede introducir una sobrecarga excesiva que puedan impedir que un programa finalice en el tiempo adecuado.
- Memoria virtual. La paginación de memoria en disco introduce retardos en los procesos que pueden retardar en exceso su activación.
- Protocolos de comunicación no deterministas. Colisiones, reenvíos, etc. En el caso de un sistema distribuido los tiempos consumidos en el paso de información debe estar acotado.

Hay que distinguir entre sistemas de tiempo real y sistemas rápidos. Por ejemplo, un sistema UNIX no es de tiempo real por ejecutarse en una máquina muy rápida pues utiliza mecanismos que no son predecibles.

En los sistemas de tiempo real existen otras características comunes que conviene destacar:

- Concurrencia: En general, un mismo sistema ha de responder a distintos estímulos realizando distintos procesos ligados entre si o independientes. Se deben realizar procesos de control concurrentes, por lo que es necesario disponer de herramientas que permitan programación concurrente.
- Mantenimiento: Un problema importante en los sistemas de tiempo real es la labor de mantenimiento. Cualquier cambio requiere una nueva verificación detallada para asegurar la validez del comportamiento tanto funcional como temporal, pues la modificación de un determinado proceso puede afectar al comportamiento temporal del resto.

1.2 Propiedades importantes

Existen unas propiedades que permiten distinguir los sistemas de tiempo real de los que no lo son.

La forma de ejecutar las tareas concurrentes en los sistemas de tiempo real debe asegurar que se cumplen algunas propiedades, distintas de las que se exigen en otros tipos de sistemas, como podrían ser los sistemas de tiempo compartido. Veamos en un cuadro comparativo estas propiedades respecto a unos factores:

Factor	Sistemas de tiempo real	Sistemas de tiempo compartido
Capacidad	Garantía de plazos	Flujo
Reactividad	Tiempo de respuesta máximo	Tiempo de respuesta medio
Sobrecarga	Estabilidad	Equidad

- **Garantía de plazos:** un sistema de tiempo real funciona correctamente cuando los plazos de todas las tareas están garantizados, es decir, todas las tareas ejecutan su actividad dentro de plazo cada vez que se activan. En un sistema de tiempo compartido lo más importante es asegurar un flujo (número de activaciones por segundo) lo más elevado posible. Se habla de deadline en los STR o de MIPS en los STC.
- **Tiempo de respuesta máximo:** en un sistema de tiempo real se trata de acotar el tiempo de respuesta en el peor de los casos de todas las tareas. En un sistema de tiempo compartido, se trata de conseguir que el tiempo de respuesta medio sea lo más corto posible.
- **Estabilidad:** Si a causa de una sobrecarga del sistema no se pueden ejecutar todas las tareas dentro de plazo, se debe garantizar que, al menos, un subconjunto de tareas críticas cumpla los plazos. En un sistema de tiempo compartido, el criterio es asegurar la equidad en la ejecución de las tareas (por ejemplo, que ninguna se vea postergada indefinidamente).

1.3 Clasificación

Atendiendo al nivel de exigencia temporal, se pueden distinguir cuatro tipos de sistemas:

- **Críticos:** (hard real time systems) son aquellos en los que el tiempo de respuesta debe garantizarse a toda costa. Una respuesta tardía puede tener consecuencias fatales. *Ej: Sistema de navegación de un avión.*
- **Esenciales:** (soft real time systems) son aquellos sistemas con restricciones de tiempo en las que una respuesta tardía no produce graves daños pero si un deterioro del funcionamiento global. *Ej: Sistema de comunicaciones (no se debe producir overflow en sistemas asíncronos, ni underflow en sistemas síncronos). Sistema de videoconferencia.*
- **Incrementales:** la calidad de la respuesta obtenida depende del tiempo disponible para su cálculo. Si se les da más tiempo la respuesta mejora. *Ej: Algoritmo de cálculo iterativo. Programa de ajedrez.*
- **No esenciales:** corresponden con las tareas sin restricciones temporales.

Es posible que en un mismo sistema convivan procesos con distinto nivel de exigencia temporal. Un ejemplo claro es la labor de control de un proceso industrial y su monitorización. El control debe cumplir con exactitud las exigencias temporales (actual sobre válvulas, etc.), mientras que la presentación del estado del sistema para el operador puede sufrir un leve retraso.

Otros criterios de clasificación:

Atendiendo a la arquitectura hardware utilizada

- **Propietarios:** Desarrollados para un hardware específico y con un desarrollo particular para cada aplicación
- **Abiertos:** Basados en sistemas abiertos, incorporando estándares industriales: microprocesadores estándar, sistemas operativos estándar, protocolos de comunicaciones estándar y buses estándar.

Atendiendo a la arquitectura del sistema

- **Centralizados:** Los procesadores están localizados en un único nodo del sistema y la velocidad de comunicación entre procesadores es semejante a la velocidad del procesador.
- **Distribuidos:** Los procesadores están situados en distintos puntos del sistema y la velocidad de comunicación entre procesadores es muy baja respecto a su velocidad de proceso.

1.4 Justificación

Nos podemos hacer la siguiente pregunta: ¿Vale la pena profundizar en las particularidades de los sistemas de tiempo real? Veamos porqué.

Los sistemas informáticos en tiempo real presentan características que los distinguen de los sistemas informáticos convencionales, pues en estos el tiempo juega un papel esencial.

Ocurre que existe gran cantidad de aplicaciones del mundo real en la que se necesitan estos tipos de sistemas, por lo que merecen una atención especial.

La rápida evolución de los componentes informáticos, su elevada velocidad y su reducido coste, cada día hace más posible el desarrollo de aplicaciones software que anteriormente solo eran abordables directamente con el hardware. Esto aumenta la flexibilidad y las posibilidades de estos sistemas pero se hace necesaria la aplicación de una teoría que garantice el correcto funcionamiento de estos sistemas.

1.5 Necesidades de los sistemas de tiempo real

Existen varios asuntos que son esenciales en los sistemas de tiempo real y que conciernen al diseño y desarrollo de este tipo de sistemas. Estos se pueden clasificar en los siguientes grupos:

- a) Especificación, análisis y verificación de los sistemas de tiempo real
- b) Programación de los sistemas de tiempo real.
- c) Sistemas operativos de tiempo real

- d) Arquitecturas para tiempo real
- e) Comunicaciones en tiempo real

A lo largo del curso estudiaremos estos temas. Ahora solo comentamos algunos aspectos muy generales sobre ellos.

1.5.1 Especificación, análisis y verificación

El mayor problema en la especificación, análisis y verificación de los sistemas en tiempo real es como incluir las restricciones temporales en las especificaciones, y como verificar que el sistema cumple estas restricciones según la especificación.

Los métodos tradicionales para la especificación y análisis de sistemas informáticos, tales como las redes de Petri, las máquinas de estados finitos, etc. no son aplicables directamente a los sistemas de tiempo real. No queda claro como se pueden extender estas técnicas para que incluyan las restricciones temporales.

Las técnicas de ingeniería del software se complican en problemas grandes, resultando de difícil aplicación en los casos reales. Es necesaria una adaptación para poderlas utilizar.

1.5.2 Programación en tiempo real

La característica que más distingue a la programación en tiempo real surge de la necesidad de manejar el tiempo en todos los aspectos de la aplicación: especificación, diseño, desarrollo, prueba, verificación y ejecución.

Se han añadido características nuevas a los lenguajes para permitir la expresión de los requerimientos temporales en una tarea individual. Se necesitan nuevas técnicas y planteamientos para verificar que los programas cumplen correctamente sus propiedades temporales.

Para la programación en tiempo real juegan un papel muy importante el lenguaje que puede facilitar o dificultar la expresión de los requerimientos temporales, y el sistema operativo que ofrece las propiedades necesarias de los sistemas de tiempo real.

Alrededor de estos deben utilizarse unas técnicas de especificación adecuadas y unas buenos métodos de prueba y verificación.

1.5.3 Sistemas operativos de tiempo real

Los sistemas operativos juegan un papel clave en la mayoría de los sistemas de tiempo real. En los sistemas de tiempo real, el sistema operativo y la aplicación están fuertemente acoplados, mucho más que en los sistemas de tiempo compartido (la transportabilidad es más complicada).

Un sistema operativo en tiempo real debe ser capaz de responder a determinados eventos internos o externos en un margen de tiempo predecible, so pena del impacto que se puede producir sobre otras tareas o procesos en ejecución. Esta es la diferencia de los sistemas de tiempo compartido, los cuales deben compartir los recursos del sistema (tales como CPU, memoria principal, y periféricos) igualmente entre un número de tareas en ejecución. En un sistema de tiempo compartido debidamente configurado,

ninguna tarea debe quedar esperando indefinidamente un recurso del sistema. En un entorno de tiempo real, las tareas críticas debe recibir los recurso que necesiten en el momento que los soliciten, sin importarles el efecto que esto pueda tener sobre el resto de tareas en ejecución.

Los dos elementos fundamentales en los sistemas operativos de tiempo real son:

- El planificador de tiempo real.
- El kernel de tiempo real.

La función de un algoritmo de planificación es determinar, para un conjunto de tareas dado, como se deben planificar (es decir, la secuencia en que se van a activar y el tiempo que se le da a cada una) para que en la ejecución de las tareas se satisfagan las restricciones de tiempo, de precedencia y de uso de recursos que tienen definidas.

El otro elemento fundamental es el diseño del kernel de tiempo real. Un kernel de tiempo real debe ofrecer características y primitivas que permita una eficiente intercomunicación entre las tareas, sincronización, garantizar la respuesta a las interrupciones, un rápido y fiable sistema de ficheros, y un mecanismo eficiente de manejo de memoria.

Además, las directivas deben tener un tiempo de ejecución predecible o ser reentrantes (utilizando un microkernel).

1.5.4 Arquitecturas de tiempo real

Las arquitecturas de los sistemas que han de soportar aplicaciones de tiempo real deben estar diseñadas para soportar las siguientes características críticas:

- Gran velocidad de cálculo
- Alta velocidad en la atención a interrupciones
- Alto flujo de entrada / salida.

De forma adicional, es conveniente incorporar al sistema características de tolerancia a fallos que permita la operación continua en el caso de producirse un estado anormal en el sistema o el fallo de algún componente hardware.

Tradicionalmente, los ordenadores de tiempo real se han basado en sistemas de arquitecturas propietarias, conjunto de instrucciones propietarias, y sistemas operativos propietarios. Para aumentar la eficiencia del sistema, muchas características de tiempo real del sistema están apoyadas por la arquitectura (semáforos, memoria cache, conmutación de tareas, algoritmos de planificación, algoritmos de detección y corrección de errores, y flujo elevado de entrada/salida utilizando múltiples buses). Esta alta eficiencia se consigue a costa de perder flexibilidad y aumentar las dificultades de avanzar con nuevos desarrollos del hardware.

El avance en los dispositivos hardware esta permitiendo que en la actualidad, cada vez más sistemas de tiempo real estén basados en arquitecturas abiertas, facilitando su evolución a las nuevas tecnologías.

1.5.5 Comunicaciones de tiempo real

Tanto en el diseño de sistemas en tiempo real distribuidos como centralizados, aparece la necesidad de utilizar protocolos de comunicación que ofrezcan un comportamiento determinista al sistema.

En los sistemas sin características de tiempo real, es suficiente verificar la validez lógica para disponer de una comunicación válida, mientras que en los sistemas de tiempo real, es necesario también verificar la validez temporal en la transmisión de la información. En particular, los sistemas distribuidos de tiempo real necesitan protocolos que ofrezcan un retardo acotado en el acceso a un canal de comunicación. El retardo en el tiempo de acceso a un canal se define como el intervalo de tiempo que transcurre desde que un mensaje es colocado en la cola de envío de un nodo, hasta que el mensaje es colocado en el nodo receptor. Por tanto, es la suma entre el retraso para acceder al canal y el tiempo de transmisión de un mensaje.

1.6 Aspectos particulares

1.6.1 Sistemas empotrados

En muchas ocasiones los sistemas en tiempo real son sistemas empotrados.

Desde el punto de vista del entorno.

Una característica de los sistemas empotrados es que ellos desempeñan el control de un entorno determinado en el que no es necesario el desarrollo de nuevos programas en el propio entorno. Los requerimientos de proceso no cambian en su entorno y están bien definidos dentro del entorno de trabajo. Aunque algunos sistemas empotrados se diseñan para manejar información transitoria, los requerimientos de proceso para esta información transitoria esta predefinida.

Desde el punto de vista de la funcionalidad.

Los sistemas empotrados se pueden denominar también sistemas dedicados. Estos sistemas son desarrollados para satisfacer una necesidad concretamente definida en un momento dado. Los límites de la operatividad del sistema son conocidos y pueden desempeñar la función para la que han sido diseñados durante años o décadas. De cualquier modo es importante que puedan funcionar durante un largo periodo de tiempo.

El sistema debe ser tan perfecto como sea posible utilizando la tecnología del momento en que es diseñado. Sus posibilidades para futuras aplicaciones están limitadas para pequeñas variantes de la misma aplicación. Por lo tanto no es un sistema de propósito general como puede serlo un ordenador o un microordenador, ni un prototipo que funciona correctamente en un momento dado.

Un sistema dedicado es instalado una vez y para siempre. Por lo tanto el procedimiento de desarrollo nos debe conducir a un sistema operativamente correcto pues la solución no es el resultado desarrollo incremental o sucesivas experiencias.

Si las exigencias del sistema y su entorno implican unas restricciones en el tiempo de respuesta nos encontramos ante los sistemas en tiempo real. En la práctica la mayoría de los sistemas empotrados son de tiempo real pues se deben adaptar a la evolución temporal del entorno físico que controlan y que impone sus límites de tiempo sin admitir demoras.

1.6.2 Tolerancia a fallos

Si se quiere construir un sistema de tiempo real estricto donde las consecuencias de una demora en la respuesta a un estímulo sea grave, no se puede olvidar la posibilidad de que un fallo en el sistema lo deje inoperativo y, por tanto, se alcance el tiempo límite de reacción sin que esta se produzca.

Si bien la tolerancia a fallos es un problema al que es complicado dar solución, se complica mucho más si, además, se han de tener en cuenta restricciones temporales, sobre todo si tenemos en cuenta que muchas veces, la recuperación de un fallo se consigue basándose en una demora en el proceso de cálculo (técnica de recuperación y vuelta a atrás).

Otro elemento que aumenta la complejidad de la tolerancia a fallos es la interacción con el entorno. Las operaciones de entrada salida presentan un problema difícil de resolver tanto en las técnicas de redundancia espacial como temporal, utilizadas par conseguir la tolerancia a fallos.

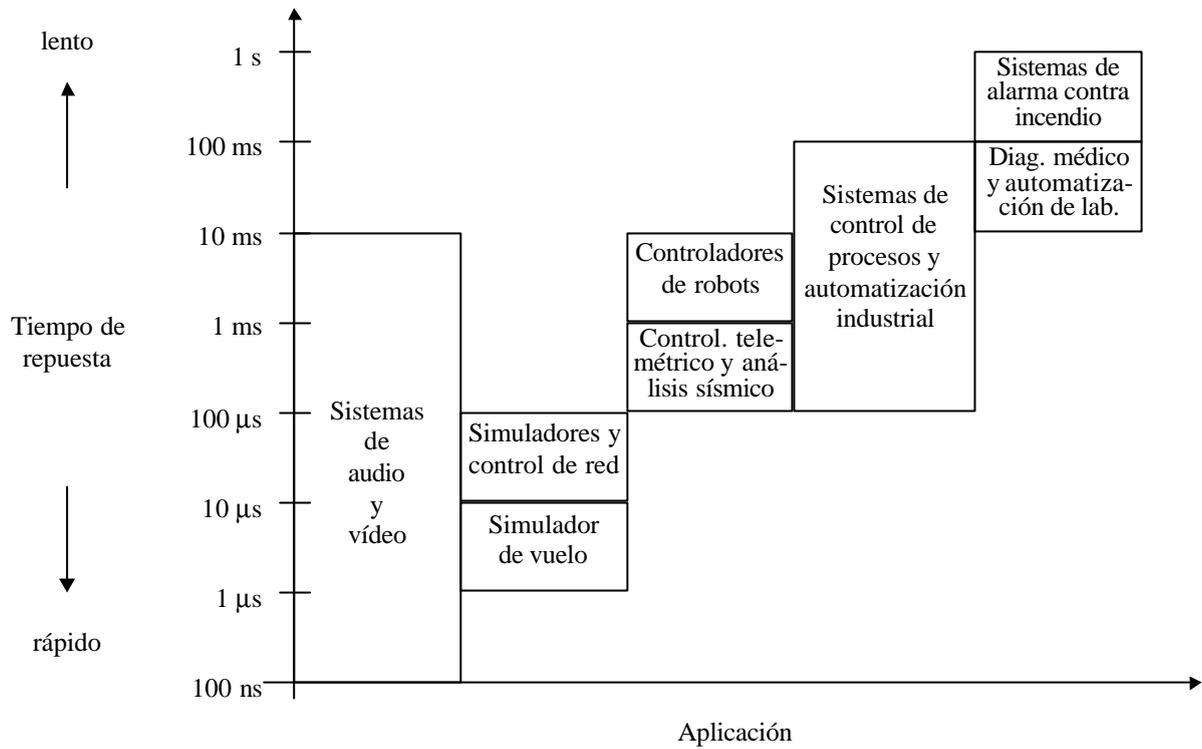
Si tenemos que con unos buenos equipos la probabilidad de producirse un fallo es pequeña (MTBF: Middle Time Between Faults) y que el conseguir tiempo real y tolerancia a fallos es complicado, muchas veces no se considera estrictamente la tolerancia a fallos en los sistemas de tiempo real. Sin embargo, es sistemas en los que un fallo puede acarrear un coste alto, se hace necesario emplear técnicas que eliminen o minimicen la posibilidad de fallos.

1.7 Ejemplos

Algunos ejemplos de aplicaciones de sistemas de tiempo real son:

- Sistemas de aviónica, tecnología espacial y armamentista.
- Sistemas de control de procesos.
- Sistemas de seguridad.
- Equipos en hospitales.
- Sistemas de robótica.
- Sistemas de fabricación.
- Sistemas de control de comunicaciones.
- Sistemas multimedia. Vídeo conferencia, reproducción audiovisual.
- Sistemas de reconocimiento.

Los ordenes del tiempo que se manejan depende mucho del tipo de aplicación. Algunos ejemplos son:



La clasificación de un sistema en una de las categorías de tiempo real no reside en el orden de magnitud de los tiempos que se manejan sino en la importancia del cumplimiento de las restricciones temporales.