



Sistemas embebidos

INTRODUCCIÓN A LOS SISTEMAS EMBEBIDOS

IGNACIO ANDRADE SALAZAR

Fundamentos de sistemas en tiempo-real

Indicaciones: Responda las siguientes preguntas con base en el capítulo 1 del libro de texto.

1. ¿Cuál es la definición de sistemas en tiempo-real para el diccionario inglés *Random House*?

Relacionado a aplicaciones en las que la computadora debe responder tan rápido como el usuario o como necesite el proceso ser controlado.

2. Mencione un ejemplo de una aplicación donde el procesamiento de datos se lleve a cabo con una tasa regular.

Procesamiento de pulsos del acelerómetro para determinar la posición de un avión.

3. Mencione un ejemplo de una aplicación que requiera de una respuesta rápida a eventos que ocurran a tasas no regulares.

Manejo de una falla por sobrecalentamiento en una planta de energía nuclear.

4. Defina en dos palabras lo que se entiende por el término “tiempo-real” con base en los ejemplos anteriores.

At once (instantáneamente)

5. Un pasajero llega al área de registro de su aerolínea para recoger su pase de abordaje para un vuelo de Nueva York a Boston que sale en cinco minutos. La recepcionista ingresa la información apropiada al sistema y en pocos segundos su pase de abordaje se imprime. ¿Este sistema es en tiempo-real y si es así, por qué?

Si esta en tiempo real debido a que la información debe procesarse en un intervalo de tiempo específico o se corre riesgo de que ocurra una falla.

A continuación, defina cada uno de los siguientes términos con una orientación para ingenieros prácticos:

Hardware. Ejecuta de manera repetida el lenguaje máquina.

Software. Instrucciones de lenguaje máquina programa de sistema y programa de aplicación.

Programa del sistema. Consiste en software que interacciona con el hardware de la computadora subyacente, tal como drivers, interruptores, programas de las tareas que actúan como herramienta para el desarrollo de análisis de los programas de aplicación.

Compilador. Traducen los programas de alto nivel a código ensamblador.

Ensamblador. Convierte el código ensamblador a código binario o máquina.

Enlazadores/Localizadores. Preparan el código objeto para ejecutarse en un ambiente de hardware específico.

Sistema operativo. Un sistema operativo es una colección especializada que maneja un sistema de programas que administra los recursos físicos de programa.

Programa de la aplicación. Son programas específicos para resolver problemas, por ejemplo, la localización de un banco de ascensores en un edificio.

6. Enuncie la definición concisa de “sistema”.

Mapeo de un conjunto de entradas y conjunto de salidas.

7. ¿Cuándo se puede considerar que un sistema es de caja negra?

Cuando los detalles internos no son de particular interés.

8. Mencione las cinco propiedades generales que según Vernon (1989) debe poseer un sistema.

- **Un sistema es un ensamble de componentes conectados entre sí de forma organizada.**
- **Un sistema es fundamentalmente alterado si un componente entra o sale de él.**
- **Tiene propósito.**
- **Tiene un grado de desempeño.**
- **Ha sido definido para ser de particular interés.**

9. ¿Qué representan las entradas en un sistema informático? Dé algunos ejemplos.

Las entradas representan datos digitales provenientes de dispositivos de hardware u otros sistemas de software. Las entradas son frecuentemente asociados a sensores, cámaras, etc.

10. ¿Qué representan las salidas digitales del sistema de cómputo?

Las salidas digitales pueden ser transformadas en salidas analógicas para controlar dispositivos externos como actuadores.

11. Defina el tiempo de respuesta de un sistema.

Es el tiempo entre la actualización del conjunto de entradas hasta que todas las salidas se actualicen mientras el comportamiento requerido es realizado.

12. Escriba la definición (I) de sistema en tiempo-real.

Un sistema de computo que debe satisfacer las restricciones acotadas en su tiempo de respuesta, o de lo contrario puede presentar fallas.

13. Defina una falla.

La incapacidad del sistema para desempeñarse conforme a sus especificaciones,

14. Defina lo que es un sistema fallado.

Incluyendo restricciones de tiempo debido a que con base en ellos se definirá si el sistema esta fallando o no.

15. ¿Por qué es necesaria una especificación rigurosa de los criterios de operación de un sistema?

Es aquel cuya conectividad esta basada en las salidas y plazos se comporten según sus especificaciones.

16. Escriba la definición (II) de sistema en tiempo-real.

Es un sistema que contiene una o más computadoras(procesadores) tienen un rol en la totalidad del sistema, pero el sistema no es explícitamente llamado computadora.

17. Enuncie la definición de sistema embebido. Mencione algunos ejemplos.

Un automóvil moderno (aire acondicionado, bolsas de aire, etc).

18. ¿Qué se entiende por sistema reactivo? Dé un ejemplo.

Un sistema reactivo es un tipo de sistema embebido que responde continuamente a eventos externos o estímulos en tiempo real, sin depender necesariamente de una secuencia predefinida de acciones. Estos sistemas están diseñados para interactuar con su entorno y responder de manera rápida y precisa a cambios en las condiciones de entrada.

Un ejemplo común de un sistema reactivo es un sistema de control de semáforos en una intersección de calles. Este sistema debe monitorear constantemente las condiciones del tráfico, como la presencia de vehículos en diferentes carriles y las señales de peatones, y tomar decisiones instantáneas sobre cuándo cambiar las luces del semáforo para garantizar un flujo seguro y eficiente del tráfico.

19. Defina sistema en tiempo-real suave.

Es aquel cuyo desempeño es degradado, pero no destruido por la falla, por el incumplimiento de las descripciones determinadas.

20. Defina sistema en tiempo-real rígido.

Es aquel en el cual la falla se presenta por incumplir un solo de sus plazos, lo que le puede conducir a una falla completa o catastrófica del sistema.

21. Defina sistema en tiempo-real firme.

Tienen plazos definidos rígidamente y esos plazos pueden ser tolerados. Unos pocos plazos incumplidos no conducirán a la falta total, pero incumpliendo algunos más allá de estos puede conducir a una falla completa o catastrófica.

22. Escriba un ejemplo de cada uno de ellos y explique por qué se encuentra en cada clasificación.

Control de misil de un avión

Control de un robot herbicida

Consola de videojuegos

23. ¿Cuál es el objetivo principal de la ingeniería en sistemas en tiempo-real?

Procesa cada cosa tan lenta como sea posible para no saturar al procesador.

24. Mencione el principio básico para el diseño de los sistemas en tiempo-real.

Es priorizar la predictibilidad y la capacidad de respuesta del sistema para cumplir con los requisitos de tiempo específicos, asegurando así un comportamiento predecible y confiable en situaciones en tiempo real.

25. Mencione una regla de dedo práctica para sistemas en tiempo-real rentables y robustos.

Una regla de dedo práctica para sistemas en tiempo real rentables y robustos es la regla del 80/20. Según esta regla, se busca diseñar el sistema para que el 80% del tiempo se dedique al manejo de operaciones normales y el 20% restante para manejar excepciones y situaciones imprevistas.

26. Defina la puntualidad de un sistema en tiempo real y mencione ejemplos en donde es particularmente importante.

La puntualidad de un sistema en tiempo real se refiere a la capacidad del sistema para completar sus tareas críticas dentro de límites de tiempo específicos y predefinidos, conocidos como restricciones temporales.

Sistemas de control de tráfico

Sistemas de control industrial

Sistemas médicos

27. Utilice como ejemplo la puerta de un ascensor que se opera de manera automática y que tiene la capacidad de detectar posibles pasajeros que se encuentran entre las hojas de la puerta que se cierra para calcular de forma aproximada el tiempo de respuesta de la reapertura de la puerta antes de sus hojas toquen al pasajero y amenacen su seguridad.

La puntualidad del sistema en tiempo real es esencial para garantizar la seguridad de los pasajeros al calcular de forma aproximada el tiempo de respuesta necesario para la reapertura de la puerta antes de que sus hojas toquen al pasajero.

28. ¿Cuál es el elemento que predomina en la obtención del tiempo de respuesta general del sistema?

Mencione ejemplos de declaraciones que representen una toma de decisión en un diagrama de flujo y diga qué representan en cualquier lenguaje de programación.

El elemento que predomina en la obtención del tiempo de respuesta general del sistema es el tiempo de ejecución de la tarea más larga o crítica, también conocido como tiempo de respuesta máximo. Esto se debe a que el tiempo de respuesta general del sistema no puede ser más rápido que el tiempo necesario para completar la tarea más larga.

Ejemplo(Python):

if condicion:

Si la condición es Verdadera

Siguiente Paso A

else:

Si la condición es Falsa

Siguiente Paso B

29. Defina evento en el ámbito de sistemas de software.

Algo que ocurre o ocasiona que el contador del programa cambie de manera secuencial, cambiando así el flujo de control.

30. ¿Cuál es la diferencia entre un evento síncrono y uno asíncrono?

La diferencia fundamental entre un evento síncrono y uno asíncrono es cómo afectan el flujo de ejecución del programa. Los eventos síncronos bloquean la ejecución hasta que se complete el evento, mientras que los eventos asíncronos permiten que la ejecución continúe sin esperar a que se complete el evento.

31. ¿Cuáles son los eventos aperiódicos?

Los eventos aperiódicos son aquellos que no siguen un patrón regular de ocurrencia y pueden ocurrir en momentos impredecibles o irregulares, lo que los diferencia de los eventos periódicos que ocurren en intervalos regulares y predecibles.

32. ¿Cuáles son los eventos esporádicos?

Los eventos esporádicos son aquellos que ocurren de forma ocasional o irregular, sin seguir un patrón regular en su ocurrencia, y pueden estar asociados con una variedad de factores o condiciones específicas que no ocurren regularmente. Estos eventos pueden tener un impacto significativo en sistemas físicos, tecnológicos y ambientales, y a menudo requieren medidas de respuesta y mitigación adecuadas.

33. Indique un evento periódico pero síncrono.

Un ejemplo de un evento periódico pero síncrono es la señal de reloj en un sistema digital. La señal de reloj es un evento periódico que ocurre a intervalos regulares y está sincronizado con el ciclo de funcionamiento del sistema. En un sistema digital, la señal de reloj se utiliza para sincronizar y coordinar las operaciones internas del sistema, asegurando que las operaciones se realicen en momentos específicos y predecibles.

34. Dé un ejemplo de un evento síncrono pero aperiódico.

Un ejemplo de un evento síncrono pero aperiódico es la recepción de datos en un sistema de comunicación serie síncrono. En este caso, la recepción de datos está sincronizada con una señal de reloj, lo que hace que sea síncrona en el sentido de que ocurre en momentos específicos determinados por el ciclo de reloj. Sin embargo, la llegada de los datos puede ser aperiódica, es decir, no sigue un patrón regular en cuanto a su ocurrencia en el tiempo.

35. Dé un ejemplo de un evento esporádico y síncrono.

Un ejemplo de un evento esporádico y síncrono podría ser la recepción de una señal de alerta en un sistema de monitoreo de seguridad. Supongamos que un sistema de monitoreo está configurado para recibir alertas de intrusión a través de una conexión de red. Cuando se detecta una intrusión, el sistema envía una señal de alerta que se transmite de forma síncrona a todos los dispositivos de seguridad conectados al sistema.

36. Defina tiempo de liberación en el ámbito de sistemas de software.

En sistemas de tiempo real, el tiempo de liberación se refiere al momento en el cual una tarea o proceso debe haber completado su ejecución y liberado sus recursos asignados dentro de un límite de tiempo específico para garantizar el cumplimiento de los requisitos temporales del sistema.

37. Enuncie la definición de sistema determinístico.

Un sistema determinístico es aquel en el cual el comportamiento futuro del sistema puede predecirse completamente a partir de condiciones iniciales conocidas y reglas bien definidas, sin ninguna aleatoriedad o incertidumbre en su evolución.

38. Defina el factor de utilización del CPU.

El factor de utilización del CPU es una métrica que indica la proporción de tiempo que el CPU de un sistema informático está ocupado ejecutando tareas en relación con el tiempo total de observación, expresado como un porcentaje. Es una medida importante para evaluar el rendimiento y la eficiencia del CPU en un sistema.

39. Indique las disciplinas que intervienen en la ingeniería de sistemas en tiempo-real.

ingeniería de software

ingeniería de sistemas

Computación y sistemas embebidos

ingeniería de control

Ingeniería electrónica y eléctrica

Matemáticas y teoría de la computación

Ingeniería de redes y comunicaciones

Seguridad y Fiabilidad

40. Enuncie problemas prácticos que se deben atender en el diseño e implementación de sistemas en tiempo-real.

Garantía de tiempos de respuesta

Gestión de recursos

Tolerancia a fallos

Sincronización y coordinación

Diseño de interfaces hombre-máquina

Seguridad y protección

Escalabilidad y adaptabilidad

41. Especifique algunas aplicaciones en tiempo real para las siguientes áreas: aeroespacial, civil, industrial, médica y multimedia.

Aeroespacial:

- **Control de vuelo de aeronaves**
- **Navegación y posicionamiento**
- **Telemetría y seguimiento**

Civil:

- **Control de tráfico**
- **Sistemas de gestión de energía**
- **Monitoreo de infraestructuras críticas**

Industrial:

- **Control de procesos**
- **Automatización de fábricas**
- **Mantenimiento predictivo**

Médica:

- **Monitoreo de pacientes**
- **Cirugía asistida por computadora**
- **Sistemas de imágenes médicas**

Multimedia:

- **Transmisión de video en tiempo real**
- **Videojuegos en línea**
- **Realidad virtual y aumentada**

42. ¿Qué aspectos de los sistemas embebidos mejoraron con los trabajos teóricos realizados en la década de los 80's?

- **Técnicas de diseño y modelado**
- **Arquitecturas de hardware y software**
- **Técnicas de optimización y compilación**
- **Tolerancia a fallos y seguridad**
- **Sistemas distribuidos y en red**