

Interrupciones en los sistemas operativos

Autora
Saray Cubillos García

July 3, 2020

Las interrupciones tanto internas como externas permiten directamente que el sistema operativo ejecute varias funciones al mismo tiempo haciendo referencia principalmente a la transferencia y el manejo de datos a gran escala.

1 ¿Qué es una interrupción?

Una interrupción es una señal del hardware que cumple con la función de avisar que se ha presentado un evento externo a la ejecución normal del sistema. De esta forma el sistema operativo debe tomar el control del procesador para solucionar el problema. Lo que más importa en el tratamiento de las interrupciones es que al tratar de terminar la interrupción, se retoman los datos del proceso de ejecución desde el punto donde había ocurrido la interrupción, el programa seguirá ejecutándose como si no hubiera ocurrido nada.

Generalmente, el procesador reconoce dos clases de eventos que pueden dar lugar a estos cambios: las interrupciones y las excepciones o también llamadas interrupciones por software. La presencia de interrupciones le facilita al sistema operativo el manejo de los recursos del sistema:

- En un ambiente de multiprogramación permite al sistema operativo tomar control del procesador para actuar cuando se arroje un problema del hardware o del programa que se está ejecutando.

- Permite la ejecución de operaciones de input/output y el uso del procesador de manera simultánea, debido a que éste sólo recibe el aviso de que ha comenzado o acabado la operación correspondiente, con lo que la propia operación es "transparente" al procesador.

- Autoriza dividir el tiempo del procesador entre todos los procesos existentes en el sistema, facilitando de esta forma la multiprogramación. El sistema operativo asigna a cada proceso un período de tiempo hacia otro.

- Admite reconocer eventos ajenos que deba controlar el sistema, ya que al producirse, darán lugar a una interrupción que podrá ser atendida de manera adecuada.

Las interrupciones pueden organizarse por prioridades, de tal forma que una interrupción de menor jerarquía no interrumpa a una más importante — debido a que las interrupciones muchas veces indican que hay datos disponibles en algún buffer, el no atenderlas a tiempo podría llevar a la pérdida de datos. Hay un límite de interrupciones definido para cada arquitectura, mucho más reducido que el número de dispositivos que tiene un equipo de cómputo actual. Las interrupciones son, por tanto, generadas por el controlador del canal en que son producidas.

Las interrupciones y timers son herramientas clave para lograr un manejo efectivo del tiempo y los eventos en un sistema embebido. En la programación estándar, en una PC las

interrupciones son un recurso del hardware que se usa a nivel del S.O el kernel, los drivers, también llamados "de bajo nivel". La interrupción es disparada por un evento externo al CPU. Puede ser el cambio de estado de un pin (interrupción externa) o cierta señal de un dispositivo interno del pc, por ejemplo el desbordamiento de un timer, el estado del ADC, etc.

El microcontrolador dispone de mecanismos para activar, desactivar, priorizar, inhibir, etc, distintas fuentes de interrupciones y para asociar cada una de ellas con rutinas ISR que son programadas por el usuario.

Polling vs interrupciones: Las interrupciones permiten evitar el polling, el cuál consiste en consultar permanentemente la ocurrencia de cierto evento desde el programa, contrario a la interrupción, el polling es sincrónico y determinista. Una de las principales ventajas de la interrupción sobre el polling es que permite al procesador hacer más cosas "simultáneamente". Una importante terminología es la vectorización, esta significa que cada fuente de interrupción posee una rutina de atención especializada, propia, que se ubica en alguna parte de la memoria (del programa) que el programador puede asociar de alguna manera, lo contrario a esto es que todas las interrupciones sean atendidas por la misma rutina y que dentro de ella se haga polling para determinar el origen.

2 ¿Qué tipo de interrupciones existen?

De programa: Son causadas por el programador haciendo uso de alguna condición que se produce como resultado de la ejecución de una instrucción, como el desbordamiento aritmético, la división por cero, el intento de ejecutar una instrucción ilegal de la máquina o hacer una referencia a una zona de memoria fuera del espacio permitido al usuario.

De reloj: Producidas por un reloj interno del procesador. Esto permite al sistema operativo ejecutar algunas funciones con una determinada regularidad.

De E/S: Originadas por un controlador de E/S, éste notifica que una operación ha terminado correctamente o de lo contrario con diversos errores arrojados.

Por fallo del hardware: Provocadas por fallos físicos del sistema, tales como un corte de energía o un error de paridad de la memoria.

3 ¿Cómo se hace la implementación de interrupciones a nivel de hardware?

Son las generadas por el hardware del sistema informático. También suelen conocerse como interrupciones asíncronas ya que se producen inesperada, simultánea e independientemente del procesamiento. Se pueden clasificar por sus características como:

1. Interrupciones de Entrada/Salida: Como su nombre lo indica son iniciadas por los dispositivos de entrada/salida o por los elementos hardware de conexión de los mismos al ordenador como por ejemplo una impresora, una pantalla, o las tarjetas controladoras de las mismas.

2.Interrupciones externas: Son causadas por diferentes elementos hardware del ordenador, como el reloj de interrupción, señales entre procesadores en un sistema multiprocesador, etcétera.

3.Interrupciones de reinicio: Se produce por un error de "alimentación", o al pulsar la tecla de reinicio ("bootstrap") provocando que el sistema operativo vuelva a ejecutarse desde el principio, causando el mismo efecto que si se hubiese apagado y vuelto a encender el sistema.

4 ¿Cómo se implementan las interrupciones por software?

La interrupción software por "excelencia es la llamada al sistema operativo", en la cuál el proceso es interrumpido para que el sistema operativo pueda ejecutar el servicio que se ha perdido.

Este tipo de interrupciones siempre dan lugar a un cambio de contexto, aunque no suelen dar lugar a un cambio de contexto, aunque no suelen dar lugar a un cambio de proceso debido a que el proceso solicita un servicio, una vez atendido por el sistema operativo, seguirá ejecutándose el proceso que lo solicitó. Sólo se producirá un cambio de proceso si el servicio solicitado es una operación de entrada, salida, ya que el proceso deberá esperar a que se acabe dicha operación para volver al procesador.

El número de niveles de interrupciones varía dependiendo del fabricante, como ejemplo, podríamos elegir 32 niveles de interrupción, los 16 primeros para las interrupciones software, mientras que los 16 restantes para las interrupciones hardware, que suelen ser más prioritarias.

5 Conclusión

No se encontraron datos históricos sobre las interrupciones, hay algo que me permite explicar que ha habido un avance importante y un aprovechamiento de las interrupciones sobre la historia de la informática cito "Por ejemplo, tarjetas con un reloj con batería, pues el PC perdía la hora al apagarlo, tarjetas de vídeo con posibilidades gráficas y que por lo tanto soportaban un monitor gráfico y a veces en colores, tarjetas de comunicaciones como por ejemplo tipo modem o telex, y otras muchas posibilidades."

Las interrupciones precisamente son las que nos permiten guardar toda la información que tenemos, aunque el sistema sea interrumpido al momento de retomar el proceso la ejecución deberá arrojar datos correctos, gracias al uso de las interrupciones es que podemos seguir con el funcionamiento manual del sistema, las ventanas emergentes, el retroceso del tiempo ejecutable en el programa o ver datos erróneos en el proceso de "Debugging" son los que nos permiten ver lo que está haciendo el procesador, algunas veces debemos interferir nosotros manualmente, así como el procesador lo hace la mayoría de veces en tiempo de ejecución, permitiendo así la ejecución de funciones simultáneas.

References

<https://es.slideshare.net/pablogindel/microcontroladores-6-interrupciones> (Empresa en Uruguay dedicada a dar cursos de Arduino y Robótica) <https://www.um.es/docencia/barzana/II/li04.htm> (Universidad de Murcia)