



Sebastian Balbin Rivera

## INTERRUPCIONES

La interrupción es una función que se vio en necesidad de ser implementada ya que sin esta el sistema se gastaba más rendimiento y tiempo. Cada cierto tiempo se realizaba una verificación de las señales prioritarias que llegaban al procesador de las demás componentes del sistema, como no era fijo que en el escaneo de estas señales hubiera alguna para desplegar una acción, desperdiciaba tiempo que podía emplear en otras funciones operativas.

Las demás partes del sistema continuamente están mandando señales de interrupción al procesador, pero estas tienen un orden de prioridades ya que si llegan todas juntas pueden arriesgar el correcto funcionamiento del sistema y aun peor, perder el proceso por completo de lo que se haya estado realizando durante el colapso por estas, los procesadores tienen unos buses específicos para estas señales, ya que son de alta prioridad.

Las interrupciones tienen el funcionamiento de mandar ya sea de manera asíncrona o síncrona una señal que representa una orden para el procesador, este siempre está trabajando, pero cuando le llega una interrupción, suspende sus procesos y va al controlador de interrupciones o también llamado **ISR** (Interrupt Service Routine o Rutina de Servicio de Interrupción).

El controlador de interrupciones es el encargado de notificar al kernel el cual contiene una tabla con los eventos según la señal que llegue, para poder realizar un evento en consecuencia. Todo esto se tiene que realizar de forma rápida para que el **IST**(subproceso de servicio de interrupción) que es el que carga con gran parte del proceso de interrupción, instantáneamente finalice un proceso envíe al kernel una señal para que habilite de nuevo el sistema de interrupción, esto es muy apreciable en el administrador de tareas de un computador el cual tiene en sus procesos uno que se llama justamente interrupciones, algunos computadores por fallo de hardware tienen un porcentaje de interrupción muy alto, lo cual genera que el dispositivo se ponga muy lento o colapse, ya que como se ha dicho con anterioridad una interrupción tiene que pasar por una serie de procesos para ser manejada.

Ahora imaginemos que llegan muchas señales de interrupción al mismo tiempo, porque todas las componentes del sistema tienen un canal prioritario que se comunica directamente con la CPU. El controlador puede “enmascarar” una señal de interrupción, esto quiere decir que no ejecuta instantáneamente la señal cuando llega, sino que la guarda para ejecutarla luego ya que hay otras con más prioridad, esto

lleva a definir interrupciones enmascarables y no enmascarables, para poder tener una jerarquía y mantener estable el computador y evitar inconvenientes. Un ejemplo de cómo se puede ver el proceso de interrupciones, es abrir el administrador de tareas, y simples cosas como mantener en movimiento el mouse o presionar teclas forma una variación en interrupción que normalmente es de 0.1 %, dependiendo del computador.

Las interrupciones pueden sonar de manera general, como si todas funcionaran igual, pero realmente hay varios tipos de estas y son:

- **Interrupciones Software:** esta se genera por medio de un software, es programada. Esto quiere decir que se realiza cuando el software se necesita algo del sistema operativo. Estas interrupciones pueden simular algunas de hardware, por eso es la confusión de diferencias entre estas interrupciones.

- **Excepciones:** estas interrupciones son más superficiales, esto quiere decir que son interrupciones que se forman por problemas de ejecución como desbordamiento del stack, operaciones invalidas como división por cero, etc. Y se realizan de manera síncrona, esto quiere decir que espera a que termine un ciclo para ser recibida y ejecutada. Recordando que hay tablas para las interrupciones, en una excepción la CPU realiza una interrupción que la lleva a la **IDT** (Descriptor de Tabla de interrupción) y la lleva otra tabla para ejecutar la función respectiva.

- **Interrupciones Hardware:** este tipo no se genera por programas, sino por señales directas de los dispositivos hacia la CPU como los puertos, las tarjetas, etc. Cuando estos requieren que la CPU ejecute su controlador, y así estos dispositivos van enviando cada que necesitan atención de la CPU.

Cuando se habla de las interrupciones por un programa, se refiere a que por medio de un lenguaje se puede programar una interrupción. Cada lenguaje cuando se compila es traducido a lenguaje de máquina, por lo que no es que se llame la interrupción directamente como sucede en lenguaje ensamblador. Cuando por medio de estos programas se llama a una interrupción, esta va hacia el kernel llamándolo con una instrucción INT, por ejemplo, el 0x80 (en lenguaje ensamblador), esta instrucción puede variar dependiendo del sistema que se tenga, por lo tanto, el código de interrupción depende del hardware, ya que se tiene que traducir el código en lenguaje de alto nivel a máquina.

## REFERENCIAS:

<https://www.sciencedirect.com/topics/engineering/interrupt-service-routine>

[https://es.qwe.wiki/wiki/Interrupt\\_handler](https://es.qwe.wiki/wiki/Interrupt_handler)

<https://www.techopedia.com/definition/22195/software-interrupt>

<https://es.wikipedia.org/wiki/Interrupcion>

<http://www.int80h.org/>

[http://www.linfo.org/int\\_0x80.html](http://www.linfo.org/int_0x80.html)