

# Interrupciones

Andrés Felipe Flórez Gil  
CC. 1017269766

3 de julio de 2020

En programación una interrupción es un mecanismo a través del cual se envía una señal al microprocesador para indicarle que debe detener la ejecución del programa actual para ejecutar otro proceso prioritario. Puede suceder como un evento asíncrono y una vez que se ejecuta se devuelve al programa anterior.[1]

Hablando históricamente de las interrupciones se puede destacar el avance en ejecución y manejo que han tenido con el paso del tiempo. Existe una técnica antigua de interrupción llamada 'Polling', cuyo mecanismo se basa en un sondeo continuo, donde el microcontrolador es el encargado de verificar los estados de los eventos, lo que trae como consecuencia que no puedan ser monitoreados los datos constantemente. Esta técnica no resulta muy útil por ejemplo, en una aplicación multitareas donde lo que se busca es versatilidad para que no se vean interrumpidos todos los programas en ejecución. Debido a esto fue necesario encontrar nuevas maneras de ejecutar estas operaciones y por ello la importancia de la implementación de otros mecanismos de interrupciones como el Daisy-chain que se refiere a conexión en cadena de diferentes dispositivos en orden decreciente de prioridad. La señal emitida sólo pasará a los dispositivos siguientes en el orden establecido si el anterior no solicita el servicio del procesador para llevar a cabo la nueva operación. Es evidente como el avance en materia de procesamiento de interrupciones para hacer frente a nuevas necesidades marca una importante evolución y mejora de los programas, como tenemos la capacidad de cada vez hacerlos más eficaces y mejorar su experiencia. [3][5]

Las interrupciones se pueden clasificar según cómo se genera la señal para ejecutar la interrupción y existen diferentes tipos. De manera general se puede clasificar en síncronas y asíncronas, siendo las primeras provocadas por la ejecución de un programa dentro de la CPU, son internas al programa; y las segundas son provocadas por eventos externos al programa notificando al sistema operativo a cerca de algún cambio en el ambiente de operatividad y ejecutando una interacción para tomar una decisión respecto al cambio que no se pudo tomar autónomamente por el sistema operativo. De manera específica se clasifica las interrupciones como:

-Interrupciones internas de Hardware: Las interrupciones internas son generadas por ciertos eventos que ocurren durante la ejecución. Este tipo de interrupciones son manejadas en un cien por ciento por el hardware y, no es posible modificarlas a través del software.

-Interrupciones externas de Hardware: Son enviadas por dispositivos periféricos o por coprocesadores. No es posible desactivar este tipo de interrupciones y, no son mandadas directamente a la CPU sino a circuito integrado especializado únicamente para tratar estas interrupciones.

-Interrupciones de Software: Este tipo de interrupciones pueden ser activadas directamente por el ensamblador o generadas por un mismo programa en ejecución.

Las interrupciones ingresan como una señal proveniente del periférico la cual le dice a nuestro microcontrolador que debe parar su ejecución en curso al terminar la última instrucción en ejecución después de esto guarda el estado actual (PC y REGISTROS) en la pila y se obtiene el pc de la subrutina correspondiente a esta interrupción en caso de llegar varias interrupciones a la vez se maneja un sistema de prioridad el cual va decidir cual subrutina seguirá en curso, después de obtener el pc de la subrutina correspondiente se quita el bit correspondiente a la entrada a modo de interrupción para no quedar en ciclo infinito y procede a realizar la máquina de estado con las instrucciones de la subrutina en marcha al acabar se ejecuta la instrucción RTI la cual vuelve a cargar el punto en el cual estaba el programa (PC y REGISTROS) desde el stack, de esta manera el microcontrolador maneja las interrupciones entrantes.

Las interrupciones implementadas desde de software son usadas en los programas para hacer uso de periféricos o funciones del sistema operativo estos programas se pueden auto interrumpir para dar paso a la lectura de un disco o la ejecución de otro proceso en el sistema operativo. Estas interrupciones se pueden hacer también llamando interrupciones directamente a la bio, estas interrupciones en software son mucho más veloces cuando son implementadas directamente en la bios mediante ensamblador pero necesitando conocimiento más profundos en programación de bajo nivel con ensamblador, por el contrario las interrupciones que son implementadas sobre el sistema operativo son más lentas pero son más cómodas y fáciles de usar, estas interrupciones hechas sobre el sistema operativo son hechas con lenguajes de programación de alto nivel como puede ser C pero también van a cambiar dependiendo del compilador y el hardware, ya que, si este último cambia, el protocolo de instrucciones también puede cambiar el compilador el cual ayuda entre lo que se escribe y cómo lo lleva a bajo nivel. Para darle solución a estos procesos en gran parte se hace uso de convenios entre los compiladores para hacer una base en lenguaje de alto nivel para tratar estas interrupciones. Esto mismo sucede con los diferentes lenguajes de programación, en los cuales se observa que algunos son más eficientes que otros o fáciles para trabajar con interrupciones.

En conclusión las interrupciones y el procesamiento de las mismas han ayudado y siguen ayudando ampliamente a la informática a través de diferentes mecanismos, atendiendo distintos tipos de necesidades que se presentan en determinada situación. Una de sus más destacadas utilidades es que ayudan a disminuir el tiempo de espera en el cual se puede leer una señal externa al programa comparado al sistema anterior. Con esto trae como beneficio, con gran cantidad de periféricos, mejorar la capacidad de responder de manera veloz los requerimientos de servicios de los programas.

## Bibliografía

- [1] <http://www.ciens.ucv.ve:8080/genasig/sites/organizacion-del-comp-II/archivos/Interrupciones.pdf>
- [2] <http://index-of.co.uk/Winasm-studio-tutorial/manual2-8086.pdf>
- [3] <https://es.wikipedia.org/wiki/Interrupci>
- [4] <https://aprendiendoarduino.wordpress.com/2016/11/13/interrupciones/>
- [5] [http://www3.fi.mdp.edu.ar/electrica/opt\\_archivos/arduino/Manejo\\_de\\_interrupciones.pdf](http://www3.fi.mdp.edu.ar/electrica/opt_archivos/arduino/Manejo_de_interrupciones.pdf)
- [6] <https://books.google.com.co/books?id=I3w5DwAAQBAJpg=PA11dq=microprocesadoreshl=essa=Xved=>
- [7] <http://www.fdi.ucm.es/profesor/jjruiz/WEB2/Temas/Curso0506/EC9.pdf>
- [8] <http://mimosa.pntic.mec.es/~flarrosa/raton.pdf>
- [9] <http://www.angelfire.com/al4/pc/int.htm>