

# Proyecto de investigación:

## Interrupciones

VALENTINA BOTERO VIVAS

DOCENTE

AUGUSTO SALAZAR JIMENEZ

UNIVERSIDAD DE ANTIOQUIA

INFORMÁTICA II

2 DE JULIO DEL 2020

Las interrupciones son recursos o mecanismos del microcontrolador para responder a eventos, permitiendo suspender temporalmente el programa principal, para ejecutar una subrutina, una vez la subrutina esté terminada, se reanuda la ejecución del programa principal.

Entonces, la interrupción viene determinada por la ocurrencia de una señal, la cual provoca un desvío a una dirección específica de memoria, esto interrumpe momentáneamente la ejecución del programa. A partir de esta dirección se encuentra la rutina de tratamiento que se encarga de realizar la subrutina, devolviendo después el control al punto interrumpido del programa. Este proceso se ilustra en la figura 1

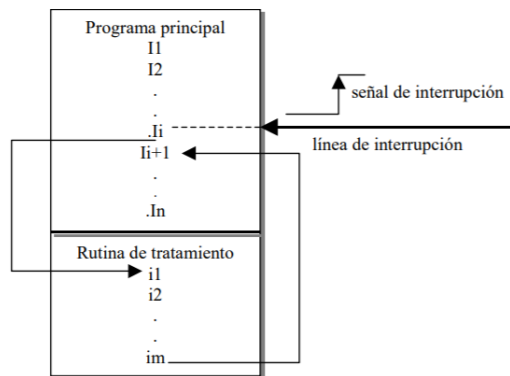


Figura 1: Interrupción ocasionada por señal externa. Tomado de [3]

Las interrupciones son de gran importancia ya que aumentan el rendimiento de los sistemas.

## Historia

Si se busca hablar de la historia de las interrupciones, es importante aclarar que éstas fueron necesarias y aparecieron como solución a un problema. Surgen de las necesidades que tienen los dispositivos periféricos de enviar información al procesador principal de un sistema de computación.

Por ejemplo, en los primeros sistemas cuando una aplicación necesitaba leer la pulsación de una teclado, interrogaba continuamente al teclado, es decir, sondeaba el dispositivo cada cierto tiempo hasta que la tecla fuera presionada, teniendo en cuenta que mientras se esperaba una tecla, no se podían ejecutar otras tareas. La solución a este problema apareció con la llamada interrupción de teclado en donde el controlador del dispositivo, en este caso el teclado, es quien genera una interrupción sólo cuando el dispositivo está listo para transferir datos.

En este caso, el microprocesador, no sondea ningún dispositivo, sino que queda a la espera de que estos le avisen cuando deba ejecutar una subtarea.

En los microcontroladores se emplea un dispositivo denominado controlador de interrupciones o unidad de interrupciones que tiene como función controlar la solicitud de eventos o dispositivos periféricos.

## Tipos de interrupciones

Las interrupciones pueden ser:

- Síncronicas: Generadas por el sistema al ejecutar instrucciones.
- Asíncronicas: Generadas por otros dispositivos que no están alineados con el reloj del sistema.

Debido a la fuente que los produce, pueden clasificarse en tres tipos:

## Interrupciones de hardware

Estas interrupciones son asincronas, es decir se pueden producir en cualquier momento sin importar lo que este haciendo el sistema. No son producidas por ninguna instrucción de un programa sino por las señales que emiten los dispositivos periféricos para indicarle al procesador que necesitan ser atendidos.

- Internas: Generadas por eventos durante la ejecución de alguna tarea. Son manejadas en su totalidad por hardware y no es posible modificarlas.
- Externas: Generadas por los dispositivos periféricos.No es posible desactivarlas.

Estas interrupciones son señales generadas desde dispositivos físicos. Así le informan al procesador que el dispositivo requiere atención, ahí es cuando se detiene la ejecución principal. Pueden ser originadas desde dispositivos como teclados, relojes, impresores, puerto serie, etcétera.

## Interrupciones de software

Estas interrupciones son sincronicas, ya que se producen cuando un usuario solicita un llamado al sistema. Es decir, son generadas por un programa mientras se está ejecutando. Este tipo de interrupciones se pueden separar en dos tipos:

- Interrupciones del sistema operativo.
- Interrupciones del usuario.

El sistema operativo es un sistema de control y como tal controla la ejecución de los programas. Las interrupciones por software dependen de las llamadas al sistema y estas no dependen de los lenguajes de programación, ni del hardware, ya que la interrupción solo es una petición o llamado al sistema operativo.

## Excepciones

Estas son sincrónicas, se dan cuando se detecta una situación de error mientras se ejecutaba una instrucción. Su principal diferencia es que mientras que en las interrupciones el programa principal se pone en pausa, en las excepciones la instrucción a bordo es abortada.

## Orden de prioridad:

1. Excepciones.
2. Interrupciones de software.
3. Interrupciones de hardware.

Ahora en los microprocesadores hay dos tipos de interrupciones, la primera corresponde a eventos que generan un estado lógico. El segundo tipo corresponde a las interrupciones que pueden ser cambiadas por software en los pines del microprocesador.

El ejemplo práctico fue realizado en la plataforma tinkercad. (Se adjunta el código usado debidamente comentado.) Se buscó implementar con interrupciones, variar la velocidad de un tren de leds por medio de dos pulsadores.

Para trabajar con las interrupciones se usó la función `attachInterrupt`, la cual es dependiente de una señal externa y requiere tres parámetros. El primer parámetro recibe un valor entero equivalente al ordinal del pin que se va a usar de interrupciones. En Arduino uno son el pin 2 y el 3. Es decir 0 y 1 en ordinal. Para evitar la confusión de estos pines se usa la función `digitalPinToInterrupt` la cual retorna el ordinal equivalente a un pin. A estos pines conectamos los 2 pulsadores. El segundo parámetro es la función o tarea que se va a ejecutar cuando se detecte la señal. Es importante aclarar que la función no recibe, ni retorna parámetros. En este caso son las funciones de aumentar y disminuir velocidad. El tercer parámetro es el tipo de señal que se detecta, en este caso se usó `RISING`, la cual lanza la interrupción cuando el pin cambie de estado de bajo a alto.

Debido a que es una interrupción, sin importar en que led vaya en el momento de enviar la señal, pausará esta ejecución y aumentará o disminuirá la velocidad para luego continuar con el proceso.

Así cada que se pulse alguno de los pulsadores se aumentará o disminuirá la velocidad según el caso.

La demostración del funcionamiento se ve en el video adjunto.

# Bibliografía

- [1] Concepto de interrupciones. (s.f) .Lenguaje ensamblador. <https://leoyac.wixsite.com/lenguaje-ensamblador/el-concepto-de-interrupciones>
- [2] Jorge Luis Tinoco.(2011). Interrupciones del microprocesador. <https://es.slideshare.net/jorg-leoxd/interrupciones-del-microprocesador>
- [3] Interrupciones.(s.f). Estructura de computadores,Facultad de Informática,UCM. <http://www.fdi.ucm.es/profesor/jjruz/WEB2/Temas/Curso05-06/EC9.pdf>
- [4] Kishin Aimura.(2014).Interrupciones y excepciones del S.O.<https://prezi.com/k-shlyu6hqr2/interrupciones-y-excepciones-del-so/>
- [5] Interrupciones de entrada y salida.(2012).<http://mariolopezchavez.blogspot.com/>
- [6] Luis del Valle Hernández.(s.f).Interrupciones con Arduino a través de un ejemplo práctico. <https://programarfacil.com/blog/arduino-blog/interrupciones-con-arduino-ejemplo-practico/>