

## Funciones principales de la interrupción 16h

### 1. Función 00h - Esperar a que se presione una tecla y leerla

- **Entrada:** AH = 00h
- **Salida:**
  - AL: código ASCII de la tecla presionada.
  - AH: código de escaneo de la tecla.
- **Descripción:** Espera a que el usuario presione una tecla y devuelve el código ASCII y el código de escaneo. Si la tecla no tiene código ASCII, AL se devuelve como 00h.

### 2. Función 01h - Comprobar si hay una tecla presionada (sin esperar)

- **Entrada:** AH = 01h
- **Salida:**
  - Si una tecla está disponible, ZF = 0, AL contiene el código ASCII de la tecla y AH el código de escaneo.
  - Si no hay teclas disponibles, ZF = 1.
- **Descripción:** Verifica si hay una tecla presionada sin bloquear la ejecución del programa. Si hay una tecla, devuelve su código; si no, establece el indicador ZF (zero flag).

### 3. Función 02h - Leer el estado del indicador de la tecla de cambio (Shift)

- **Entrada:** AH = 02h
- **Salida:** AL: estado de las teclas de cambio (Shift).
- **Descripción:** Devuelve el estado de las teclas especiales (Shift, Ctrl, Alt, Scroll Lock, Num Lock, Caps Lock, Insert). Los bits de AL representan el estado de estas teclas:
  - Bit 0: Shift derecho
  - Bit 1: Shift izquierdo
  - Bit 2: Ctrl
  - Bit 3: Alt
  - Bit 4: Scroll Lock
  - Bit 5: Num Lock
  - Bit 6: Caps Lock
  - Bit 7: Insert

## Función AH = 03h - Obtener el código de tecla sin esperar (Peek Key)

- **Entrada:** AH = 03h
- **Salida:**
  - AL: Código ASCII de la tecla (si es una tecla ASCII); si no es una tecla ASCII, AL será 00h.
  - AH: Código de escaneo de la tecla.
- **Descripción:** Esta función verifica si hay una tecla disponible en el buffer de teclado sin eliminarla del buffer y sin esperar a que se presione una tecla. Es decir, "**mira**" la tecla en el buffer pero no la quita, por lo que seguirá estando disponible para leerla con una llamada posterior.

- **Uso típico:** Esta función es útil cuando queremos comprobar qué tecla está en el buffer sin consumirla, permitiendo así múltiples verificaciones de la misma tecla.

### **Función AH = 05h - Colocar una tecla en el buffer de teclado**

- **Entrada:**
  - AH = 05h
  - CH: Código de escaneo de la tecla.
  - CL: Código ASCII de la tecla.
- **Salida:** Ninguna.
- **Descripción:** Esta función coloca una tecla ficticia en el buffer del teclado. Los valores de CH y CL se utilizan para definir el código de escaneo y el código ASCII de la tecla que se va a "empujar" en el buffer. **Es como si se hubiera presionado una tecla**, y otros programas o funciones pueden leerla posteriormente como si hubiera sido introducida físicamente.
- **Uso típico:** Esta función es útil para simular la entrada del teclado o para insertar teclas en el buffer, que luego pueden ser leídas por el programa.

### **4. Función 10h - Leer la última tecla extendida**

- **Entrada:** AH = 10h
- **Salida:**
  - AL: código ASCII de la tecla.
  - AH: código de escaneo extendido de la tecla.
- **Descripción:** Esta función permite leer teclas extendidas, que son teclas adicionales en teclados avanzados (como teclas de función extendidas o teclas de dirección en el teclado numérico).

### **Función AH = 11h - Verificar tecla extendida en el buffer (sin esperar)**

- **Entrada:** AH = 11h
- **Salida:**
  - Si hay una tecla extendida disponible:
    - ZF = 0 (zero flag en 0).
    - AH: Código de escaneo extendido de la tecla.
    - AL: Código ASCII de la tecla.
  - Si no hay tecla extendida disponible:
    - ZF = 1 (zero flag en 1).
- **Descripción:** Esta función verifica si hay una tecla extendida en el buffer del teclado **sin esperar**. A diferencia de la función AH = 00h, que espera a que el usuario presione una tecla, AH = 11h solo comprueba si hay una tecla extendida ya disponible y devuelve su código ASCII y de escaneo, en caso de que exista.
- **Uso típico:** Esta función se utiliza cuando se necesita verificar la presencia de teclas extendidas (como las flechas de dirección o teclas de función) sin bloquear el programa mientras espera la entrada.

### Función AH = 12h - Verificar el estado de la tecla extendida y obtenerla (si está disponible)

- **Entrada:** AH = 12h
- **Salida:**
  - Si hay una tecla extendida disponible:
    - ZF = 0 (zero flag en 0).
    - AH: Código de escaneo extendido de la tecla.
    - AL: Código ASCII de la tecla.
    - La tecla se elimina del buffer (ya no estará disponible para futuras lecturas).
  - Si no hay tecla extendida disponible:
    - ZF = 1 (zero flag en 1).
- **Descripción:** Similar a AH = 11h, esta función verifica si hay una tecla extendida en el buffer. Sin embargo, si la tecla está disponible, **la extrae del buffer**, lo cual significa que ya no estará disponible para otras lecturas posteriores. En otras palabras, esta función consume la tecla extendida si la encuentra.
- **Uso típico:** Se usa cuando se quiere procesar una tecla extendida de manera inmediata, quitándola del buffer para evitar que sea leída de nuevo.

### Diferencia clave entre AH = 11h y AH = 12h

- **AH = 11h:** Solo verifica si hay una tecla extendida sin eliminarla del buffer.
- **AH = 12h:** Verifica si hay una tecla extendida y la elimina del buffer si está disponible.