

# Estructura general de manejo de dispositivos con interrupciones de hardware

El manejo de los dispositivos se escribe en niveles, según su grado de acceso a los recursos del sistema. Los niveles 2 y 3 se llaman servicio.

Niveles:

## 1. Nivel de aplicación de usuario:

**Propósito** – la aplicación del usuario.

**Acceso al dispositivo** – a través de interrupción del software (nivel 2).

Ejemplo: MOV AH, 2 // INT 16h

**Tipo de programas** – todos los tipos de aplicaciones, escritos a bajo o alto nivel, que utilizan este dispositivo o esta clase de dispositivos si se trata de un servicio unificado (por ejemplo todos interfaces serie se utilizan de la misma manera del software, independiente de la versión del BIOS o el tipo del chip UART).

**Prioridades de programación** – según la aplicación, pero en general:

- Versatilidad.
- Buena estructura.
- Fácil manejo de los fuentes.

**Acceso de los recursos del SO**

- Total (salvo en caso de driver del SO).

**Dificultades de programación y depuración** – pocas.

**Subniveles:**

- Drivers del SO (a veces se mezcla con nivel 2).
  - Librerías estándar del uso de dispositivos (stdio).
- En el caso de practica 2 – uso directo del nivel 2.

## 2. Nivel de interrupción del software:

**Propósito** –

- Aislar el nivel del usuario del manejo del dispositivo.
- Manejar el dispositivo en manera independiente del hardware
- En sistemas multitarea – garantizar la serialización del uso del dispositivo (no permitir que dos programas clientes utilicen el dispositivo en el mismo tiempo en manera asíncrona).
- Manejar un dispositivo unificado en la misma manera para este grupo de dispositivos.

**Tipo del acceso al dispositivo** –

- A través de DATA POOL (DP) – método principal.
- Uso directo de los recursos del hardware del dispositivo.

**Tipo de programas**

- En general programas de bajo nivel (ensamblador), orientados a un arquitectura del ordenador (IBM PC) y un sistema operativo (MS DOS).

**Acceso de los recursos del SO**

- Poco restringido. En el caso del DOS – no restringido.

**Prioridades de programación:**

- Interfaz unificado y con funcionalidad completa.
- Facilidad de verificar
- Velocidad
- Uso modesto de recursos del sistema.
- En caso del SO multitarea – uso no monopolista de los recursos (si es posible).

**Dificultades de programación y depuración**

– media - baja.

**Subniveles:**

- 2.1. Rutinas de mantenimiento,
  - Modos de funcionamiento del servicio.
  - Instalación/desinstalación del servicio.
  - Diagnósticos del servicio
- 2.2. Rutinas que manejan sólo es data pool

- Transferir datos de/a dispositivo.
  - Control del timing del dispositivo.
  - Esperas de terminación de operaciones
  - Sincronización y serialización de tareas (si no se hace del SO).
- 2.2. Rutinas que manejan el dispositivo (instrucciones in-out).
- Arrancan el autómata finito del dispositivo, para manejar una operación de entrada - salida.
  - Otras manipulaciones necesarias del hardware.
  - En caso de manejo muy simple del hardware – sustituye la parte 3.2.

### 3. Data Pool (DP)

Estrictamente no es ningún nivel del software, sino de estructuras de datos en la memoria del sistema. Si es un nivel del servicio, tal como este es el modo de comunicación entre el nivel de la interrupción del software y las interrupciones del hardware.

El DP esta compuesto por estructuras de datos que manejan es software en estado coherente con el estado del dispositivo, es estado del propio servicio y el estado del SO.

Las estructuras más utilizadas en el DP son:

- Banderas *Flags* (mirar la instrucción TSET).
- Contadores
- Enumeradores del estado de autómatas
- Buffers cíclicos
- Colas
- Buffer Pool

### 4. Nivel de manejo de interrupciones del hardware:

#### Subniveles:

Por su carácter las interrupciones son

- 4.1. Interrupciones propias (del dispositivo) y
- 4.2. Interrupciones del sistema (por ejemplo del timer - 1Ch, del DMA etc.).

#### Propósito

- Manejar el dispositivo con instrucciones in-out obedeciendo las limitaciones de su hardware.
- Manejar los timeouts, el timing del dispositivo etc.

#### Tipo del acceso al dispositivo –

- Uso directo de los recursos del hardware del dispositivo según los datos en el DP.

#### Tipo de programas

- De bajo nivel (ensamblador), orientados a un arquitectura del ordenador (IBM PC) y el dispositivo. Pocas veces interactúan con el SO.
- Funcionamiento asíncrono con respecto con otros programas (incluso SO).

#### Acceso de los recursos del SO

- Muy restringido. Casi no existente en la mayoría de los SO (se utilizan bibliotecas muy limitadas y pequeñas para acceder a unos de los servicios de SO).
- En DOS - hay manera de acceder a los servicios de DOS si el momento de la interrupción es oportuno, es decir si no hay otros programas funcionando en las secciones críticas de DOS. Esto hace los programas que utilizan DOS en este nivel del tipo funcionar con probabilidad cerca de 1. Es decir – por suerte.

#### Prioridades de programación:

- Simplicidad
- Uso no monopolista de los recursos del sistema. Incluso en los casos del sistema de un usuario y una tarea las interrupciones son otra tarea y pueden bloquear el sistema.
- Facilidad de verificar
- Velocidad

#### Dificultades de programación y depuración

- Alta - muy alta, por lo tanto se hace lo mínimo posible en este nivel y en la manera más simple posible. La técnica más utilizada es de autómatas finitos.

## Manejo de dispositivos

