



UNIVERSIDAD DE
Belgrano
BUENOS AIRES - ARGENTINA



**Técnico en Programación de
Computadoras**

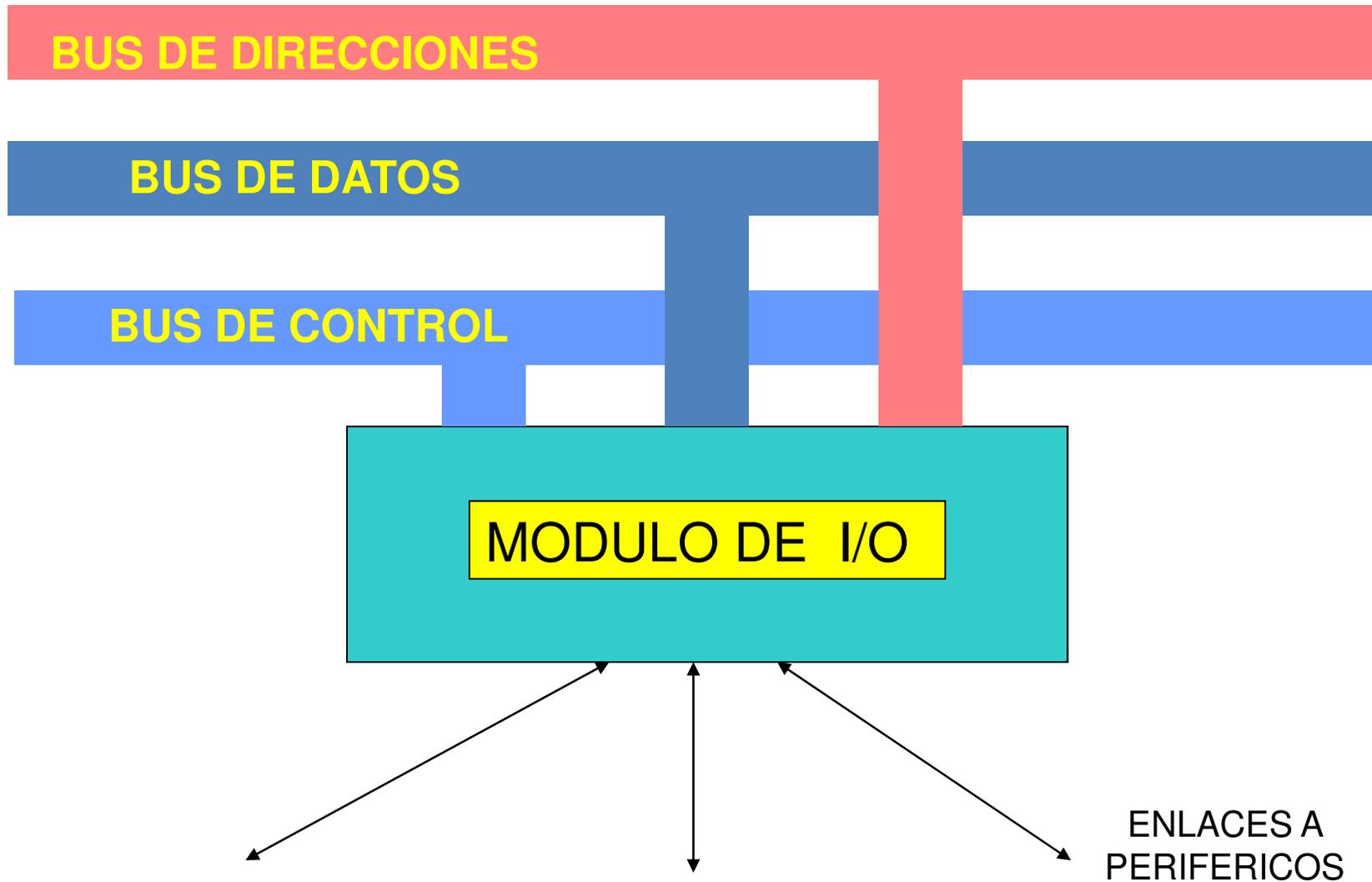
**SISTEMAS
OPERATIVOS**

2014 - Plan2014

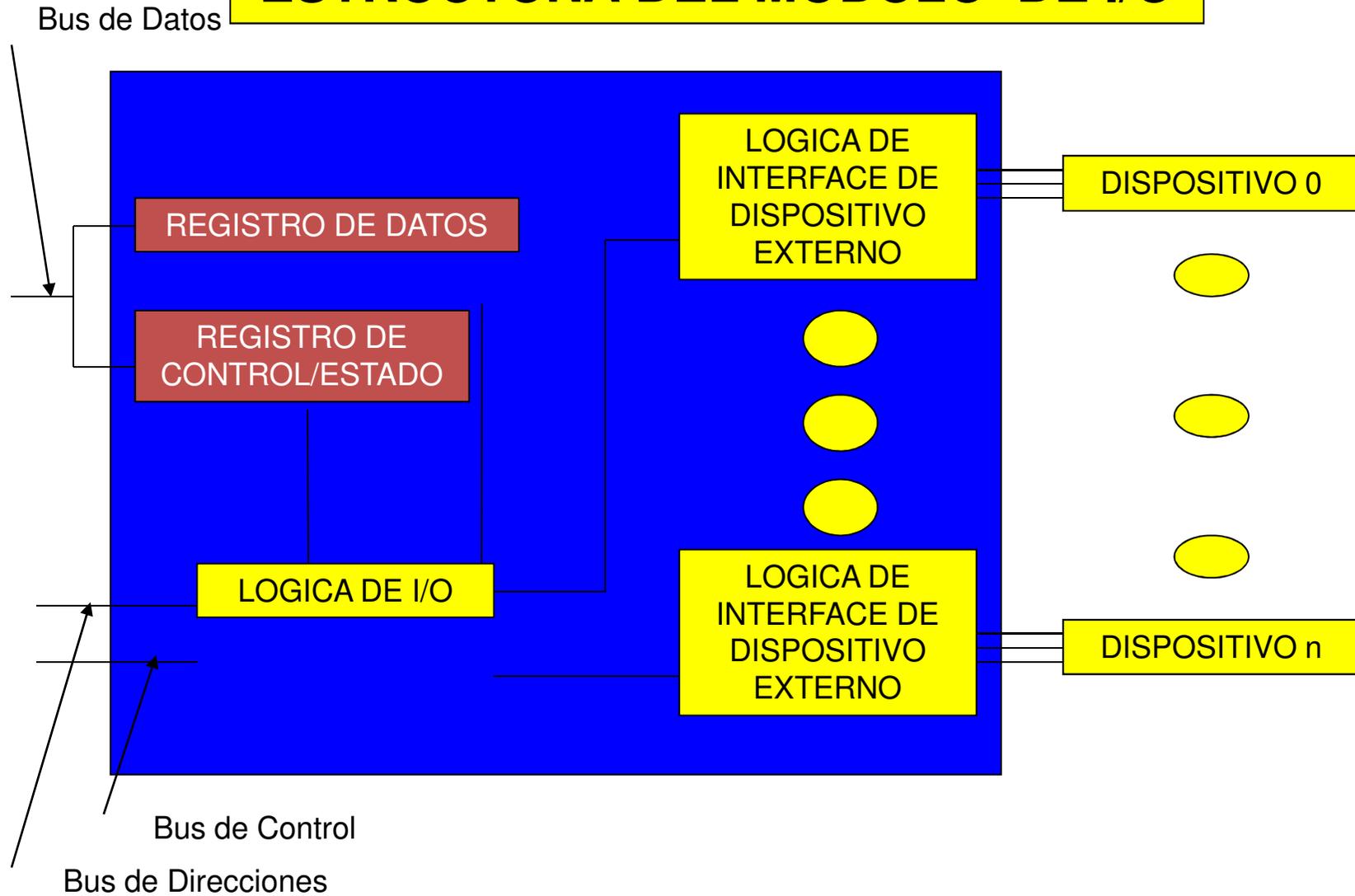
UNIDAD 5

GESTION DE ENTRADA/SALIDA

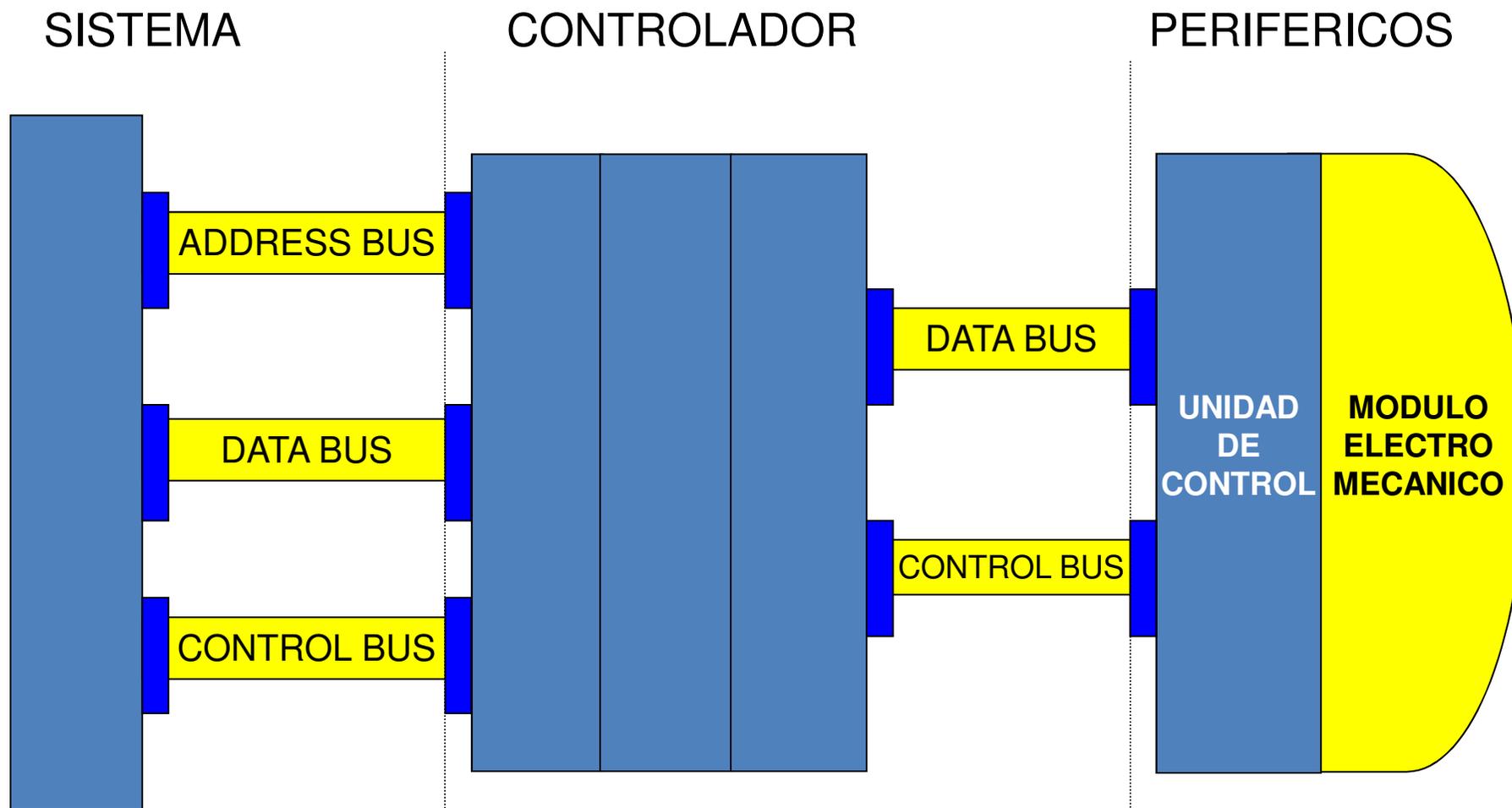
MODULO GENERICO DE I/O



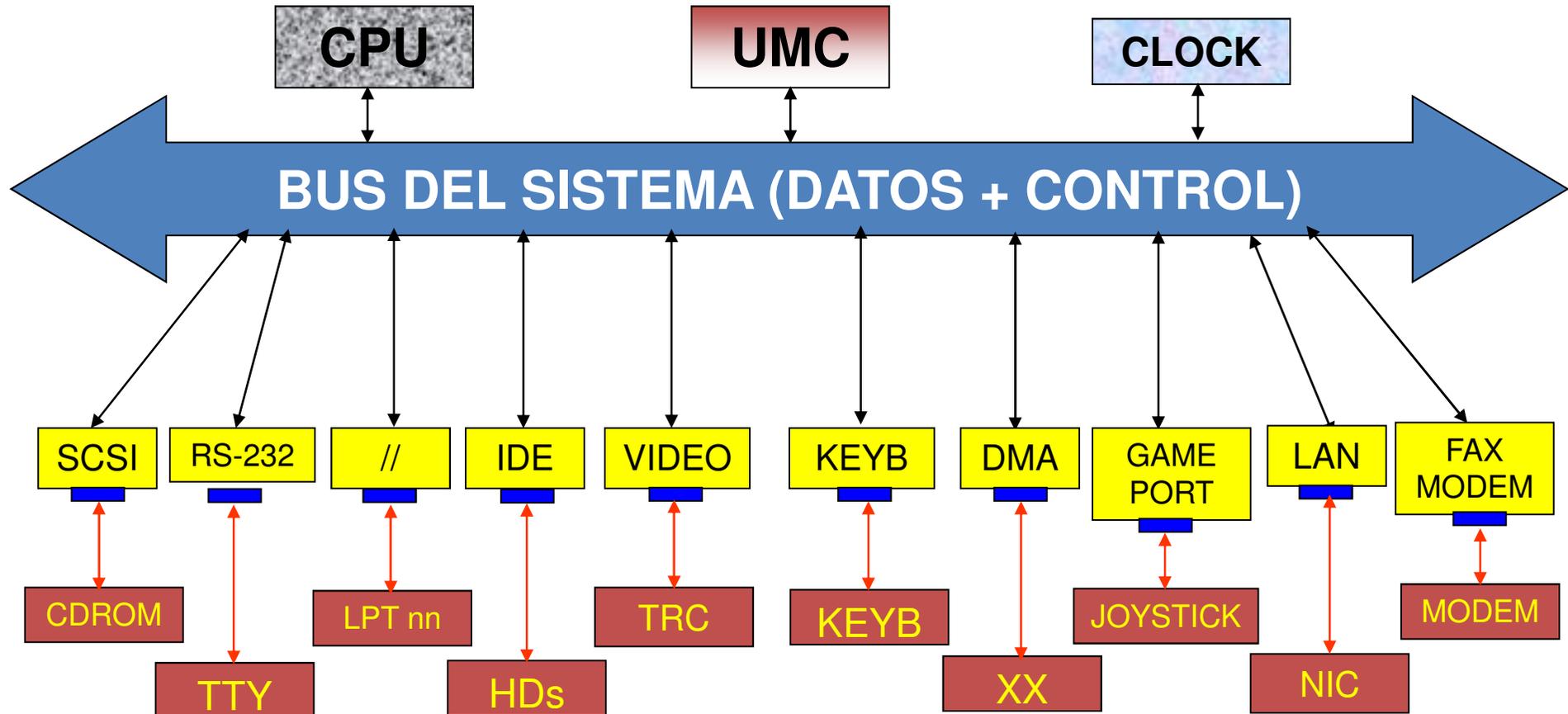
ESTRUCTURA DEL MODULO DE I/O



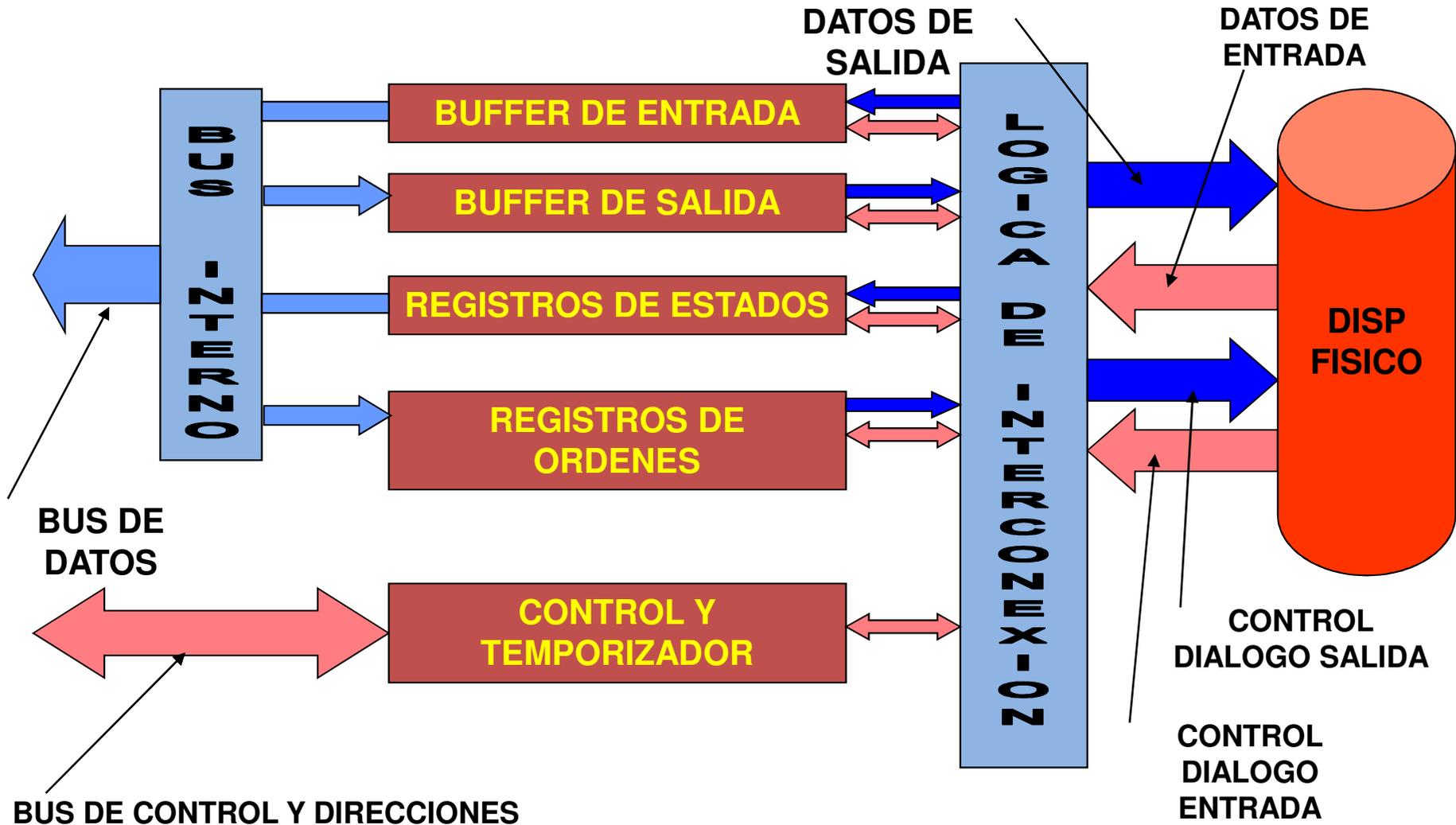
ESQUEMA DE HARDWARE DE I/O

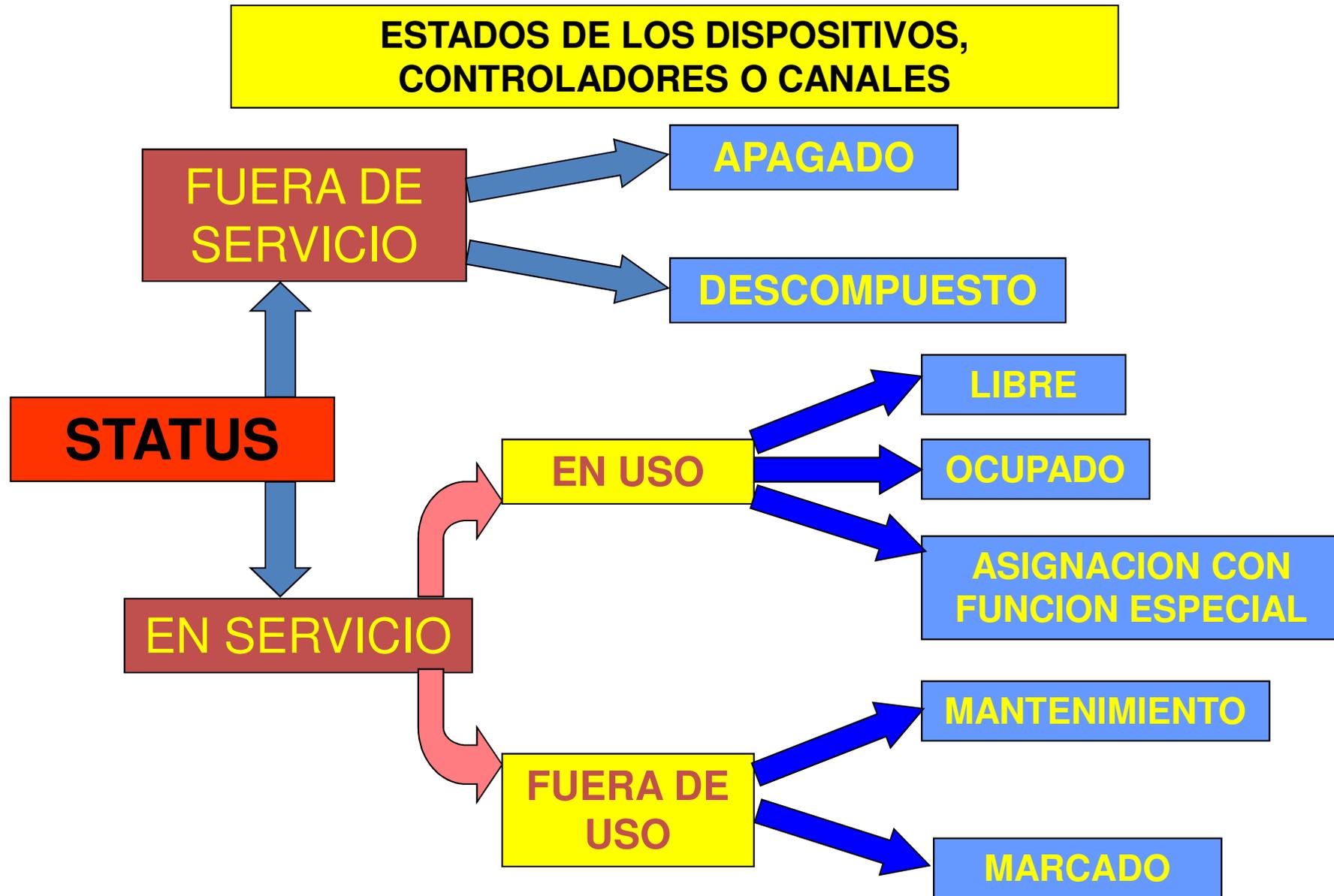


EJEMPLO DE INTERFASE DE CONEXION A DISPOSITIVOS DE I/O

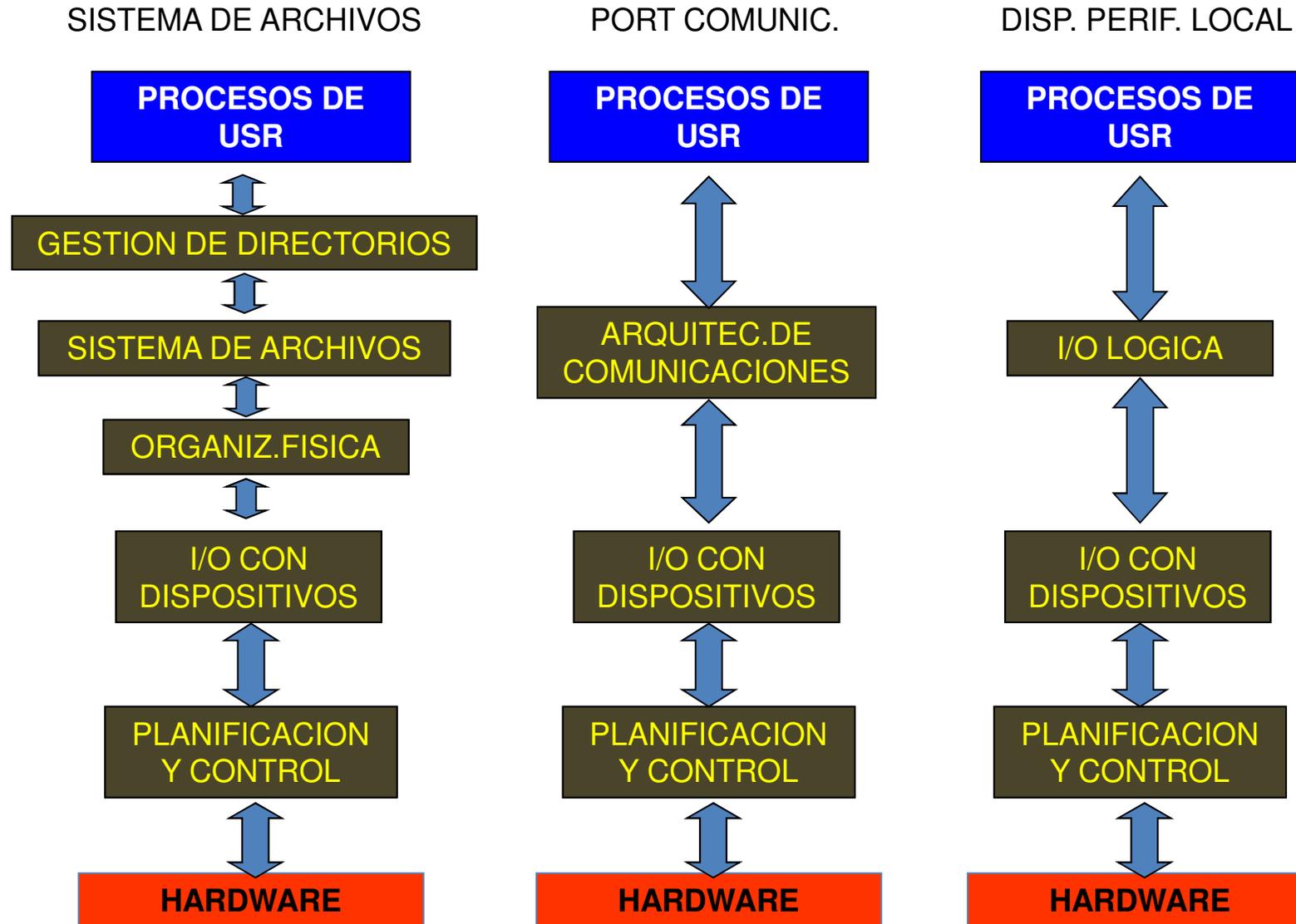


EJEMPLO DE CONTROLADOR BIDIRECCIONAL

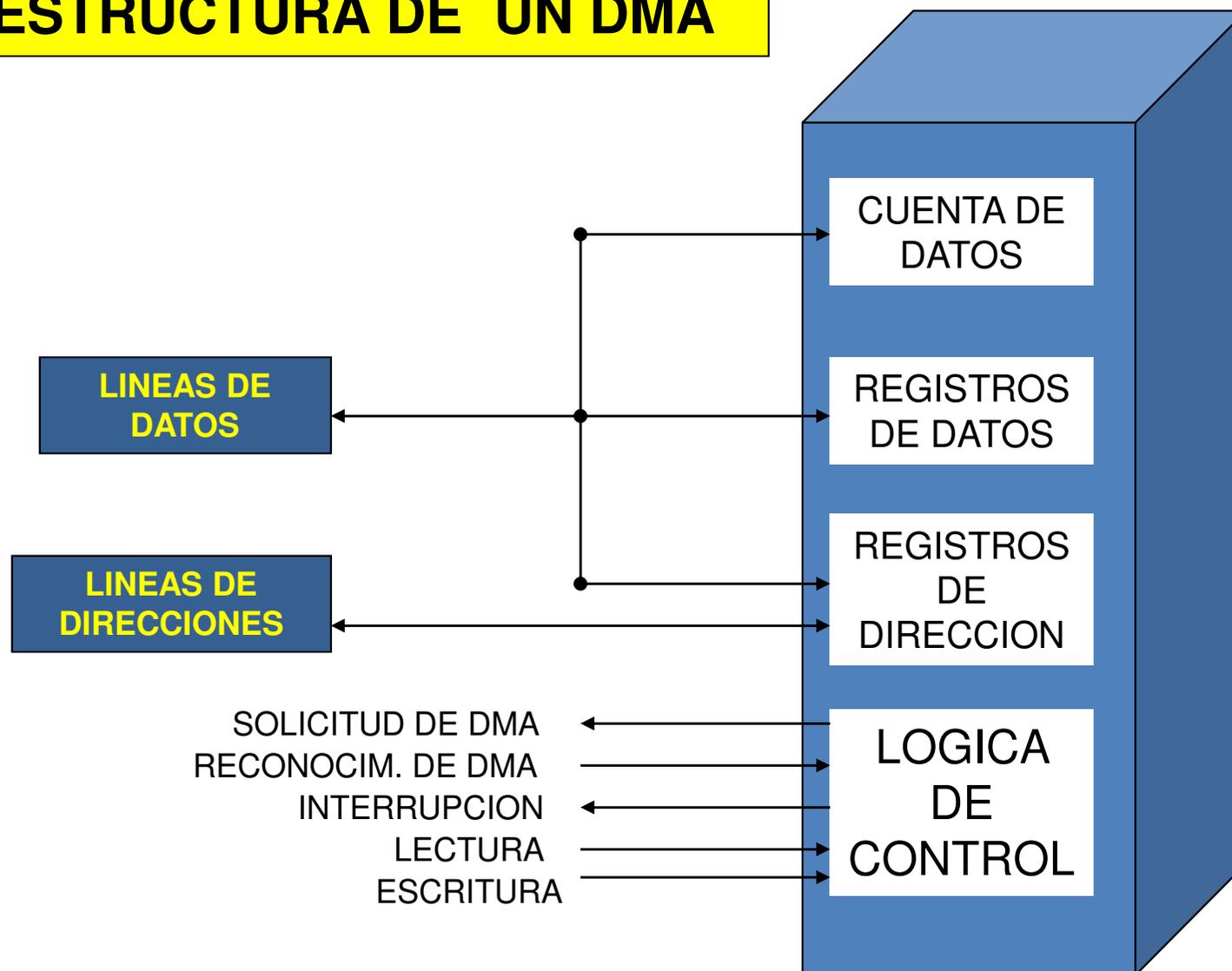




ORGANIZACION DE I/O

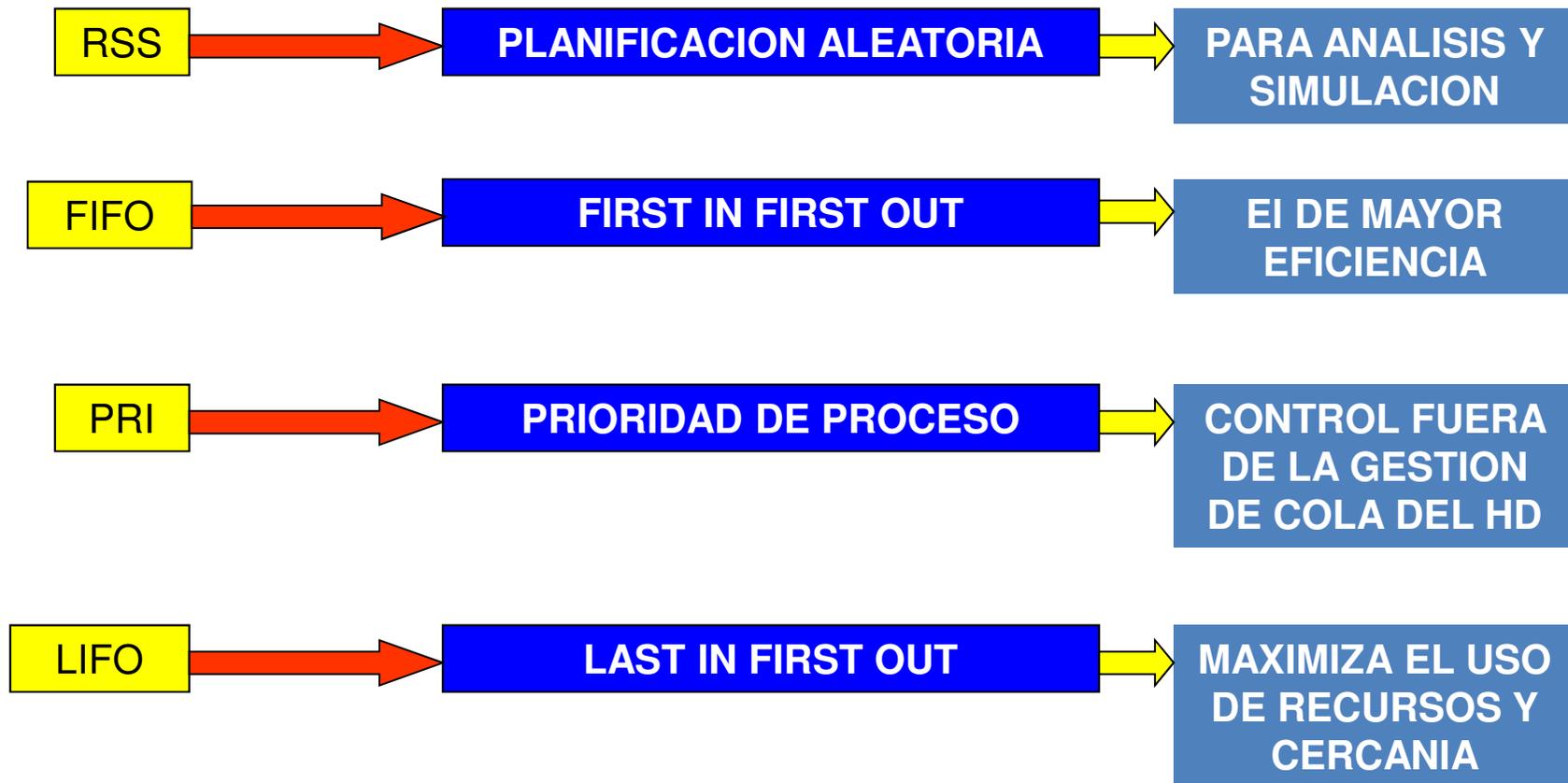


ESTRUCTURA DE UN DMA



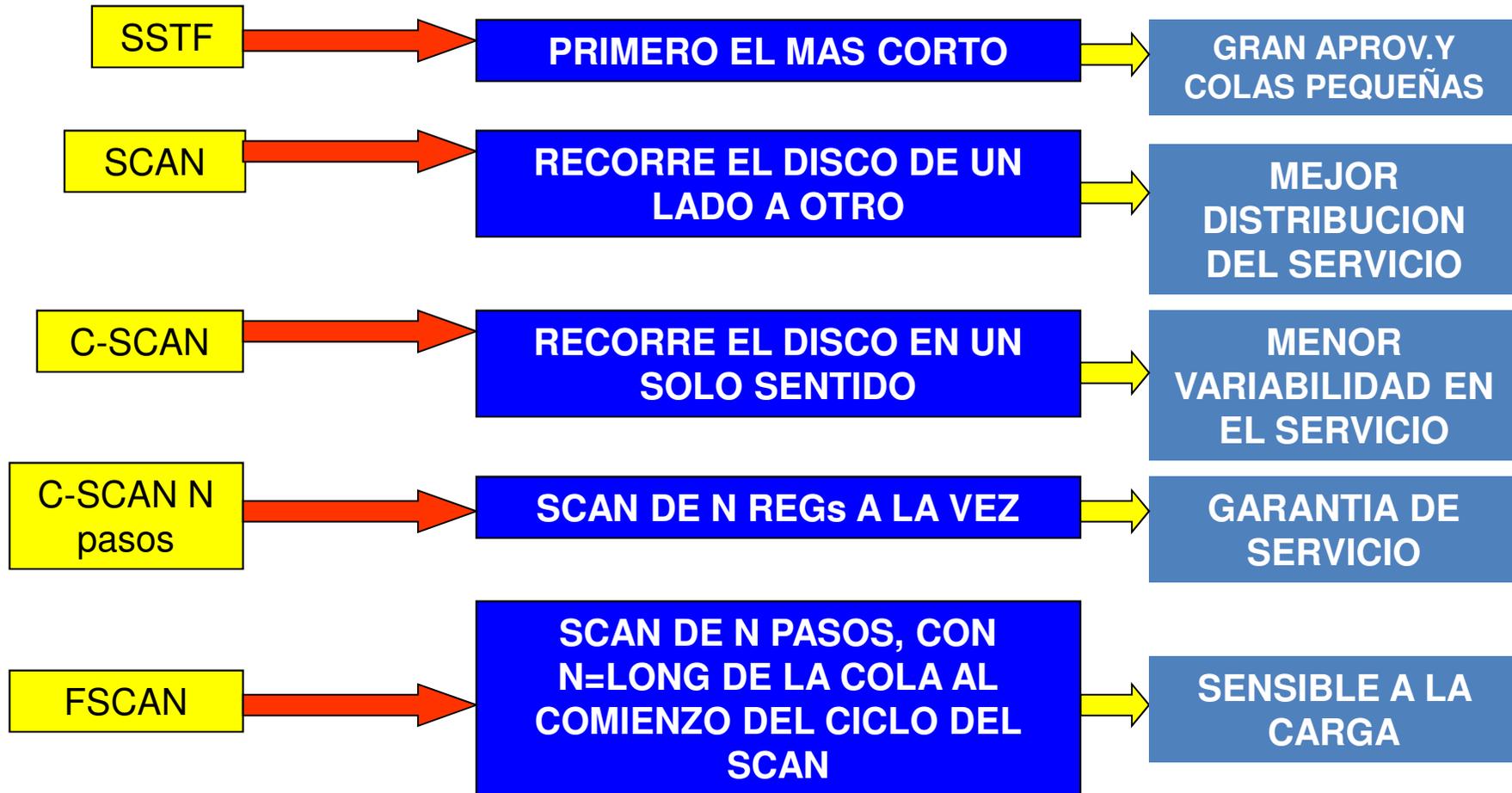
ALGORITMOS DE PLANIFICACION DE DISCOS

SELECCION EN FUNCION DEL DEMANDANTE



ALGORITMOS DE PLANIFICACION DE DISCOS

SELECCION EN FUNCION DEL ELEMENTO SOLICITADO



NIVELES DE RAID (Redundant Array of Independent Disks)

| CATEGORIA | NIVEL | DESCRIPCION |
|----------------------|-------|---|
| BANDAS | 0 | NO REDUNDANTE |
| ESPEJO | 1 | COPIA ESPEJO |
| ACCESO PARALELO | 2 | REDUNDANCIA POR CODIGO HAMMING |
| | 3 | PARIDAD POR INTERCALACION DE BITS |
| ACCESO INDEPENDIENTE | 4 | PARIDAD POR INTERCALACION DE BLOQUES |
| | 5 | PARIDAD POR INTERCALACION DISTRIBUIDA DE BLOQUES |
| | 6 | PARIDAD POR INTERCALACION DOBLEMENTE DISTRIBUIDA DE BLOQUES |

RAID 0 (BANDAS)

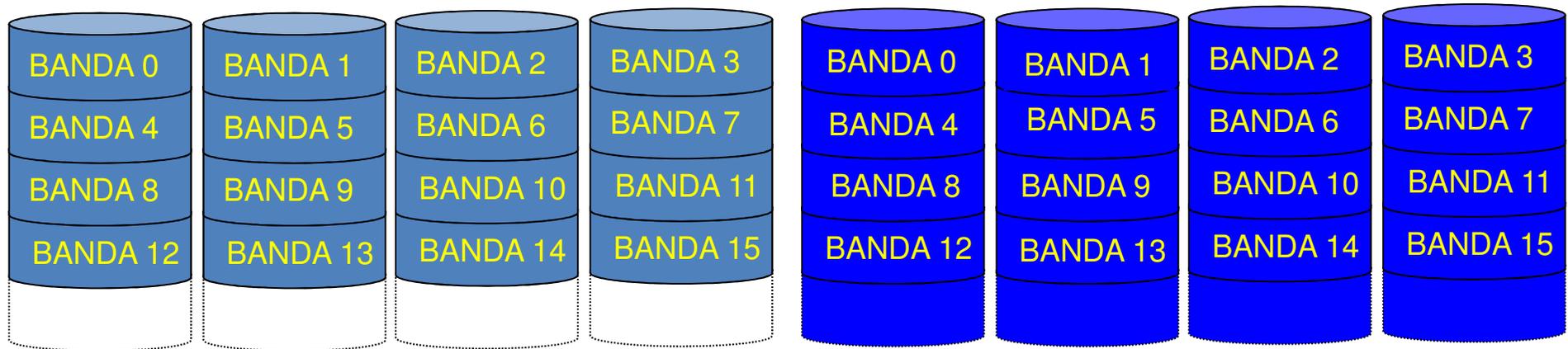
| TASA DE PETICION I/O | TASA DE TRANSFERENCIA DE DATOS I/O | APLICACION NORMAL |
|------------------------------------|------------------------------------|--|
| GRANDES DISTRIBUCIONES : EXCELENTE | PEQUEÑAS DISTRIBUCIONES: EXCELENTE | APLIC DE ALTO RENDIMIENTO EN DATOS NO CRITICOS |



RAID 0 (NO REDUNDANTE)

RAID 1 (ESPEJO)

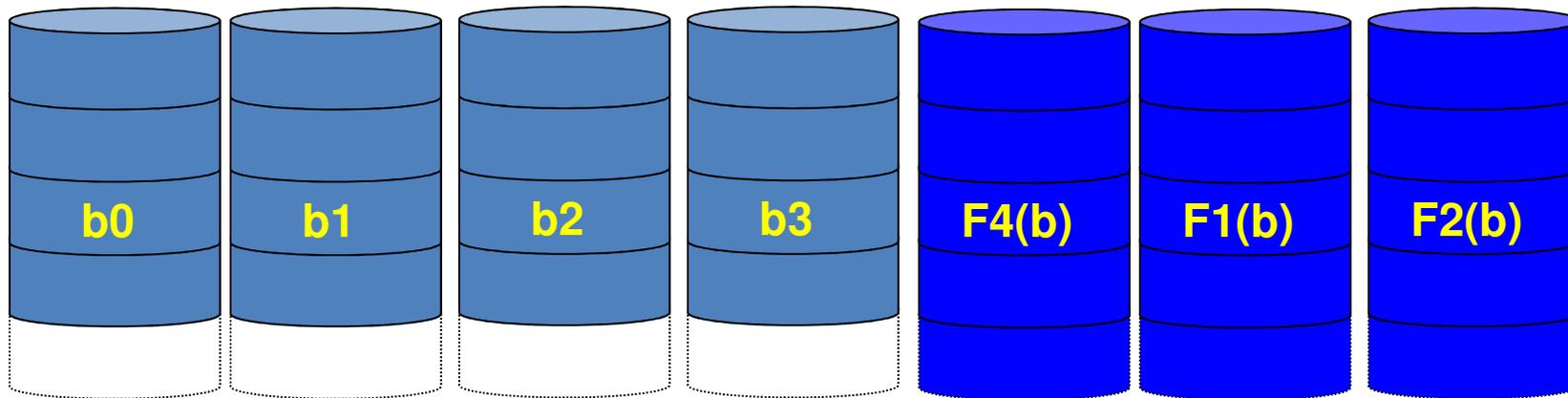
| TASA DE PETICION I/O | TASA DE TRANSFERENCIA DE DATOS I/O | APLICACION NORMAL |
|----------------------|------------------------------------|--|
| BUENO/JUSTO | JUSTO/JUSTO | CONTROLADORES DEL SISTEMA: ARCHIVOS CRITICOS |



RAID 1 (ESPEJO)

RAID 2 (ACCESO PARALELO)

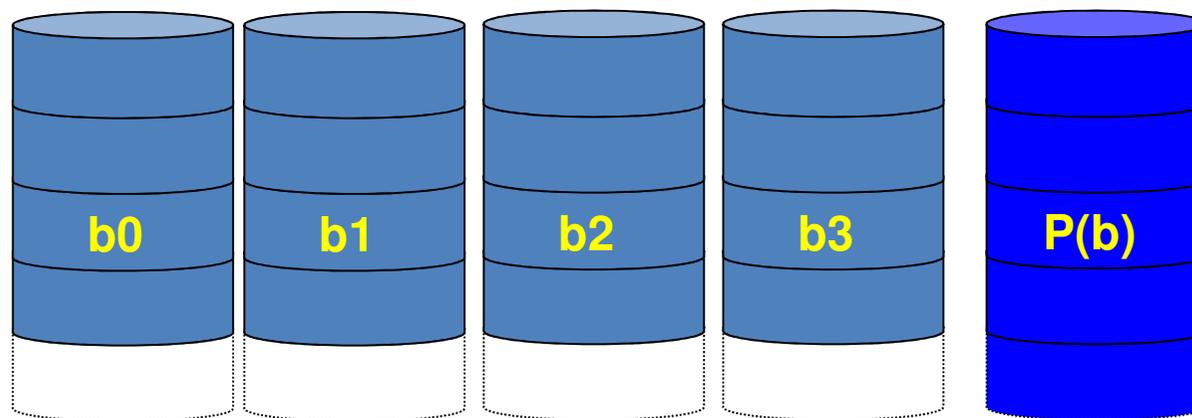
| | | |
|-----------------------------|---|--------------------------|
| TASA DE PETICION I/O | TASA DE TRANSFERENCIA DE DATOS I/O | APLICACION NORMAL |
| POBRE | EXCELENTE | SIN APLICACION COMERCIAL |



RAID 2 (REDUNDANCIA POR CODIGO HAMMING)

RAID 3 (ACCESO PARALELO)

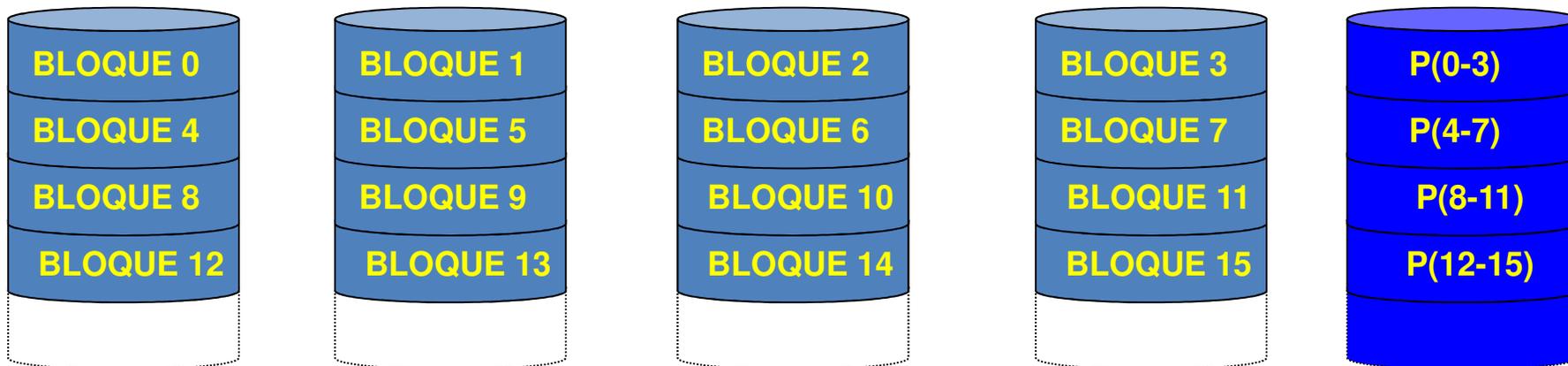
| TASA DE PETICION I/O | TASA DE TRANSFERENCIA DE DATOS I/O | APLICACION NORMAL |
|----------------------|------------------------------------|--|
| POBRE | EXCELENTE | SOLICITUD DE I/O DE GRAN TAMAÑO EN APLICACIONES TIPO CAD |



RAID 3 (PARIDAD POR INTERCALACION DE BITS)

RAID 4 (ACCESO INDEPENDIENTE)

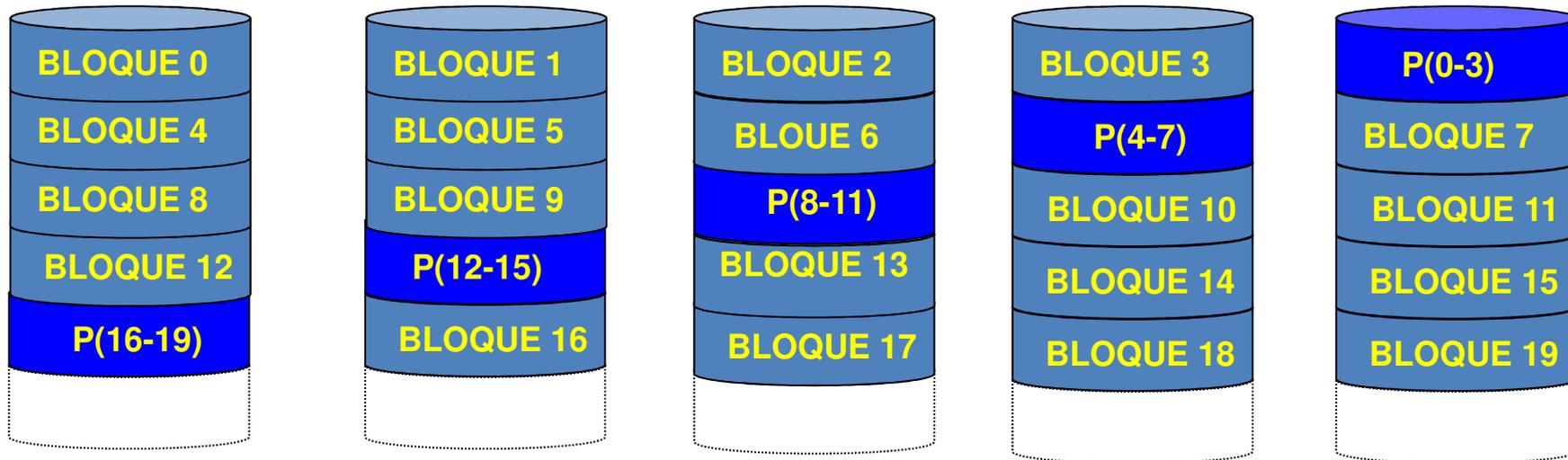
| TASA DE PETICION I/O | TASA DE TRANSFERENCIA DE DATOS I/O | APLICACION NORMAL |
|----------------------|------------------------------------|--------------------------|
| EXCELENTE/JUSTO | JUSTO/POBRE | SIN APLICACION COMERCIAL |



RAID 4 (PARIDAD POR INTERCALACION DE BLOQUES)

RAID 5 (ACCESO INDEPENDIENTE)

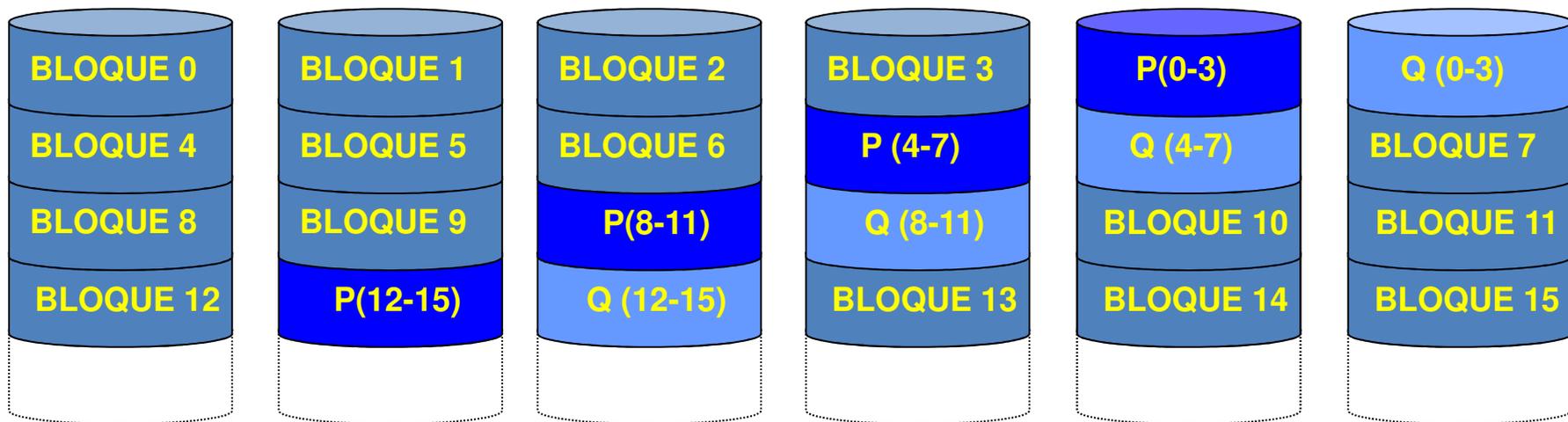
| TASA DE PETICION I/O | TASA DE TRANSFERENCIA DE DATOS I/O | APLICACION NORMAL |
|----------------------|------------------------------------|--|
| EXCELENTE/JUSTO | JUSTO/POBRE | ALTA TASA DE SOLICITUDES, LECTURA INTENSIVA, BUSQUEDA DE DATOS |



RAID 5 (PARIDAD POR INTERCALACION DISTRIBUIDA DE BLOQUES)

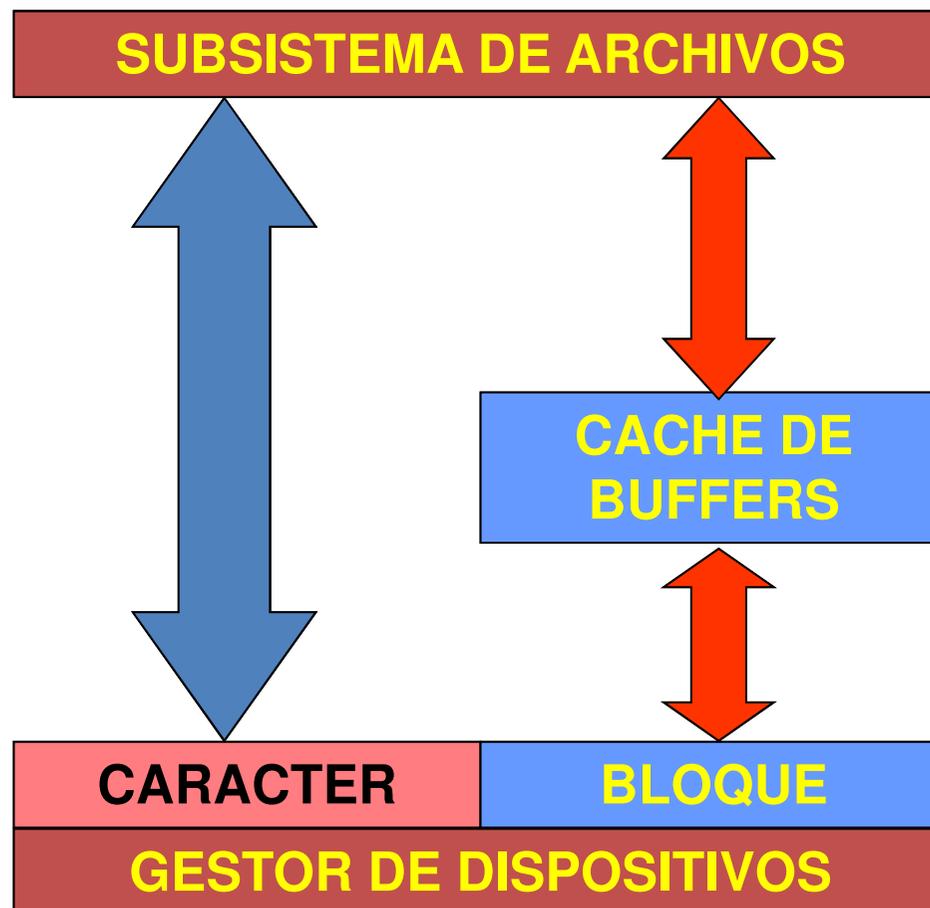
RAID 6 (ACCESO INDEPENDIENTE)

| TASA DE PETICION I/O | TASA DE TRANSFERENCIA DE DATOS I/O | APLICACION NORMAL |
|----------------------|------------------------------------|--|
| EXCELENTE/POBRE | JUSTO/POBRE | APLICACIONES QUE TIENEN DISPONIBILIDAD EXTREMADAMENTE ALTA |



RAID 6 (REDUNDANCIA DUAL)

ESTRUCTURA DE I/O EN UNIX/LINUX



ESTRUCTURA DE I/O EN WINDOWS 2000

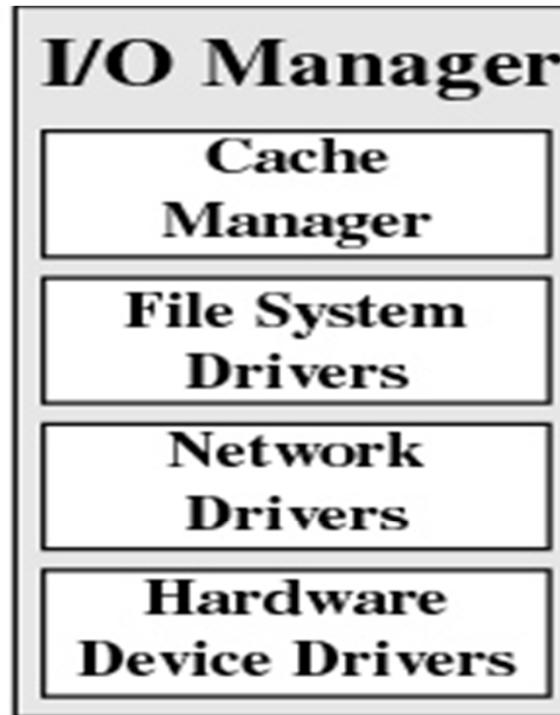


Figure 11.16 Windows 2000 I/O Manager

Ejercicios de práctica

1. Describa en forma sintética, cómo es la organización física de un disco, puede utilizar gráficos para mayor claridad.

2. La velocidad promedio para la obtención de datos de un disco esta dada por la suma de los siguientes tiempos:

Seek Time.

Latency Time.

Transfer Time.

De una definición para estos tres tiempos.

3. Supongamos que tenemos un programa de 40K words almacenado en disco con un tiempo promedio de *Seek + Latency* de 4 mseg. y una velocidad de transferencia de 1.000.000 de palabras por segundo. ¿Cuánto tiempo toma la transferencia del programa a memoria?

4. Considérese un disco típico con un tiempo de posicionamiento medio especificado de 20mseg., una velocidad de transferencia de 1 Mbyte/seg., y sectores de 512 bytes con 32 sectores por pista. El disco gira a 3600 RPM. Supóngase que queremos leer un fichero que consta de 256 sectores con un total de 128 Kbytes. Estimar el tiempo total de transferencia en los siguientes casos:

a. El archivo está almacenado secuencialmente.

b. El archivo está almacenado al azar.

BIBLIOGRAFIA DE REFERENCIA

1. Programación en Linux, con ejemplos. Kurt Wall. QUE, Prentice Hall. Madrid. 2000.
2. Sistemas Operativos. 5ta Ed. William Stalling. Pearson Prentice Hall. Madrid. 2006
3. Sistemas Operativos. 7ma Ed. William Stalling. Pearson Prentice Hall. Madrid. 2012
4. Sistemas Operativos Modernos. Andrew. S. Tanenbaum. Prentice-Hall. Interamericana S.A. Madrid, 2009.
5. Unix, Sistema V Versión 4. Rosen,Rozinsky y Farber.McGraw Hill. NY 2000.
6. Lunix, Edición especial. Jack Tackett, David Guntery Lance Brown. Ed. Prentice Hall. 1998.
7. El Libro de Linux. Syed M. Sarwar, Robert Koretsky y Syed. A. Sarwar. Ed. Addison Wesley. 2007. España.



Técnico en Programación de Computadoras

FIN UNIDAD 5

GESTION DE ENTRADA / SALIDA

