



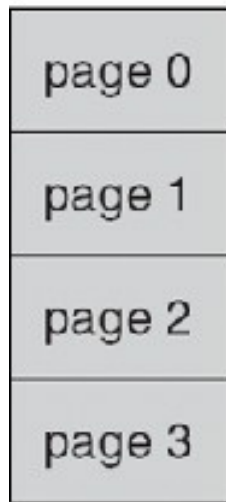
Sistemas Operativos 1

Paginación

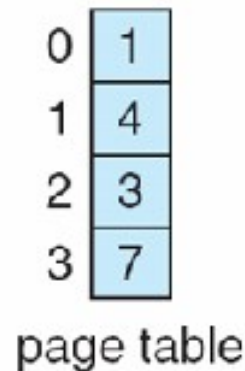
Paginación

- El espacio lógico de direcciones puede no ser continuo; se le asigna memoria a un proceso siempre que haya disponible
- Divide la memoria física en bloques de tamaño fijo llamados marcos (frames) (el tamaño es una potencia de 2, entre 512 bytes y 8192 bytes)
- Divide la memoria lógica en bloques de igual tamaño llamados páginas.
(**tamaño de pagina == tamaño del frame**).
- Mantiene registro de todos los marcos (frames) vacíos
- Para ejecutar un programa de n páginas, necesita encontrar n marcos libres y cargar el programa
- Inicializar una tabla de páginas que permita traducir una dirección lógica a una física
- Fragmentación interna

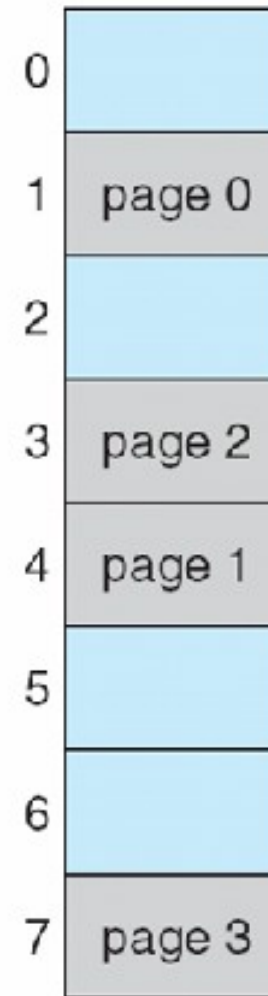
Paginación: Memoria Lógica y Física



logical
memory



frame
number

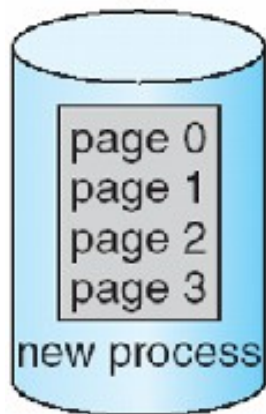


physical
memory

Marcos Libres

free-frame list

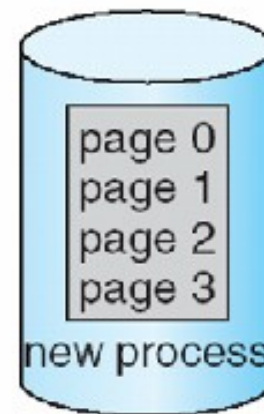
14
13
18
20
15



(a)

free-frame list

15



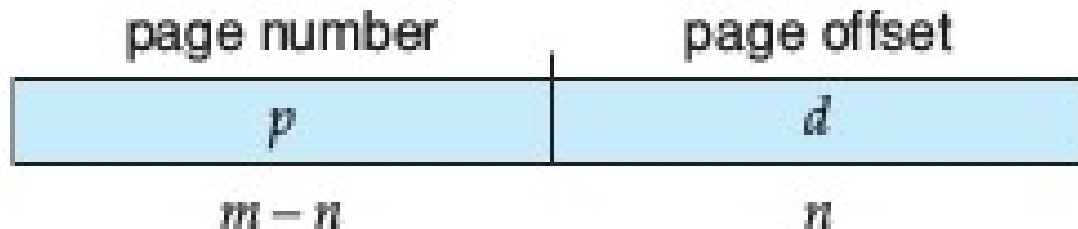
0	14
1	13
2	18
3	20

new-process page table

(b)

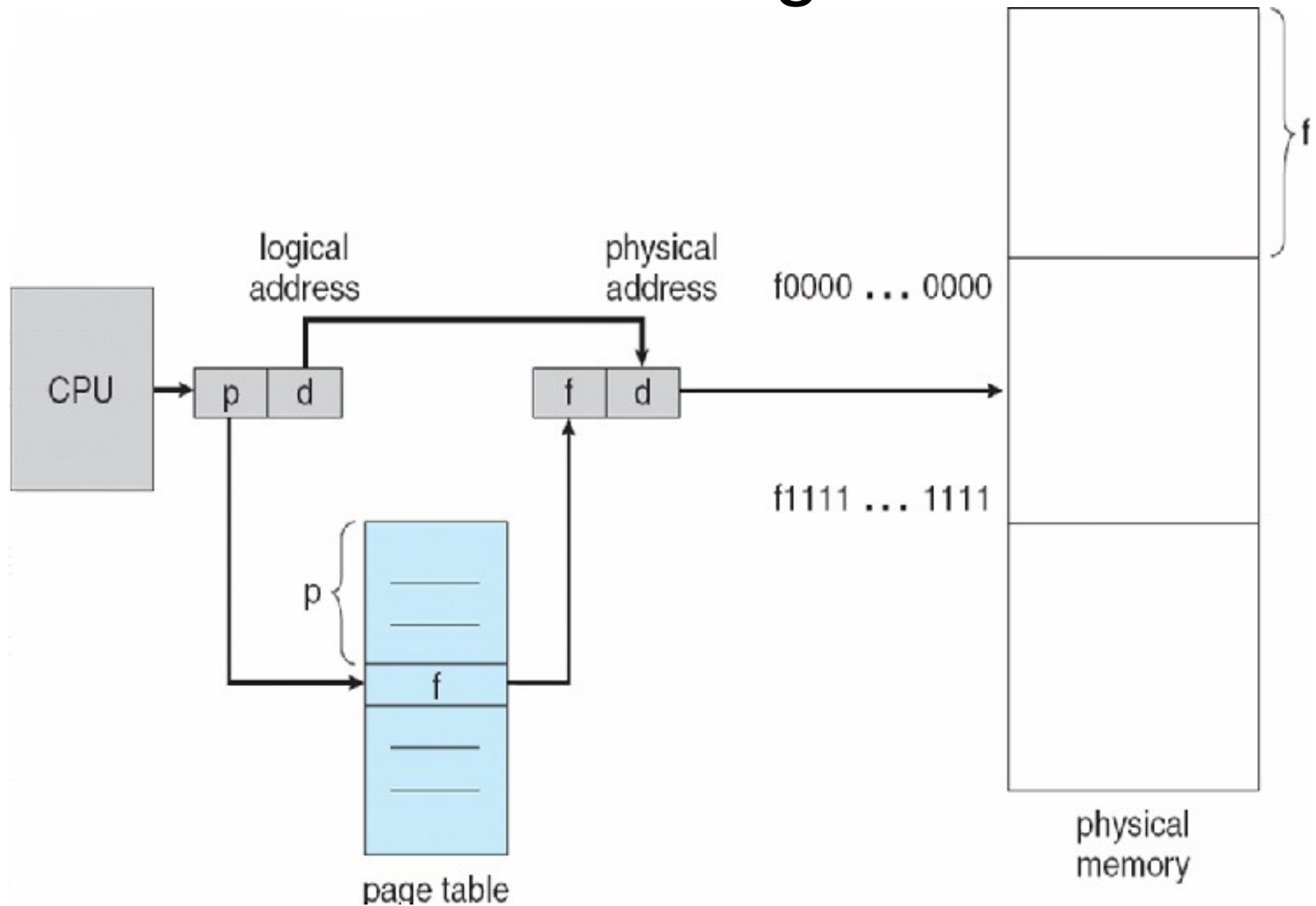
Esquema de Traducción

- Las direcciones generadas por la CPU son divididas en:
 - **Número de Página (p)** – usado para indexar dentro de la tabla de página que contiene la dirección base de cada marco
 - **Desplazamiento de Página u “offset” (d)** – combinado con la dirección base para definir la dirección física de memoria que se envía a la unidad de memoria



- Para un espacio de direcciones lógico 2^m y un tamaño de página de 2^n

Hardware de Paginación



Esquema de Traducción

Con memoria de 32-bytes con “frame size” = 4-bytes

Espacio de direcciones lógico: Con 5 bits puedo direccionar 32 bytes o “instrucciones”

- 00000b = 0d
- 11111b = 31d

m = 5

- **Tamaño de página:** Con 2 bits puedo direccionar 4 bytes o “instrucciones” (= Frame size)

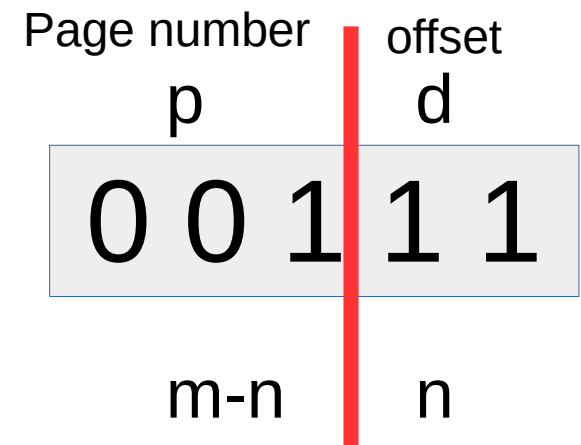
- 00b = 0d
- 11b = 3d

n = 2

Dirección Lógica a traducir: 7d

Page number: 001 = 1d

Offset: 11 = 3d



Paginación de memoria de 32-bytes con “frame size” = 4-bytes

Page 0	0	a
	1	b
	2	c
	3	d
Page 1	4	e
	5	f
	6	g
	7	h
Page 2	8	i
	9	j
	10	k
	11	l
Page 3	12	m
	13	n
	14	o
	15	p

logical memory

0	5
1	6
2	1
3	2

page table

0		Frame 0
4	i j k l	Frame 1
8	m n o p	Frame 2
12		Frame 3
16		Frame 4
20	a b c d	Frame 5
24	e f g h	Frame 6
28		Frame 7

physical memory

Paginación de memoria de 32-bytes con “frame size” = 4-bytes

00000	a
00001	b
00010	c
00011	d
00100	e
00101	f
00110	g
00111	h
01000	i
01001	j
01010	k
01011	l
01100	m
01101	n
01110	o
01111	p

Memoria lógica

000	101
001	110
010	001
011	010

Page Table

00000	
00001	
00010	
00011	
00100	i
00101	j
00110	k
00111	l
01000	m
01001	n
01010	o
01011	p
01100	
01101	
01110	
01111	
10000	
10001	
10010	
10011	
10100	a
10101	b
10110	c
10111	d
11000	e
11001	f
11010	g
11011	h
11100	
11101	
11110	
11111	

Memoria física

Direccionamiento

Dirección
lógica

0 0 1 1 1

00000	a
00001	b
00010	c
00011	d
00100	e
00101	f
00110	g
00111	h
01000	i
01001	j
01010	k
01011	l
01100	m
01101	n
01110	o
01111	p

000	101
001	110
010	001
011	010

Page Table

Memoria lógica

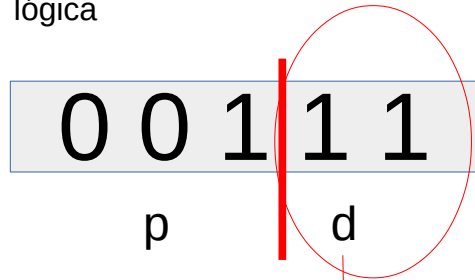
Dirección
física

01000	m
01001	n
01010	o
01011	p
01100	
01101	
01110	
01111	
10000	
10001	
10010	
10011	
10100i	a
10101	b
10110	c
10111	d
11000i	e
11001	f
11010	g
11011	h
11100	
11101	
11110	
11111	

Memoria
física

Direccionamiento

Dirección
lógica



00000	a
00001	b
00010	c
00011	d
00100	e
00101	f
00110	g
00111	h
01000	i
01001	j
01010	k
01011	l
01100	m
01101	n
01110	o
01111	p

000	101
001	110
010	001
011	010

Page Table

Memoria lógica

Dirección
física

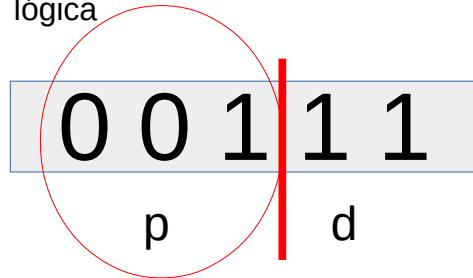


01000	m
01001	n
01010	o
01011	p
01100	
01101	
01110	
01111	
10000	
10001	
10010	
10011	
10100i	a
10101	b
10110	c
10111	d
11000i	e
11001	f
11010	g
11011	h
11100	
11101	
11110	
11111	

Memoria
física

Direccionamiento

Dirección
lógica



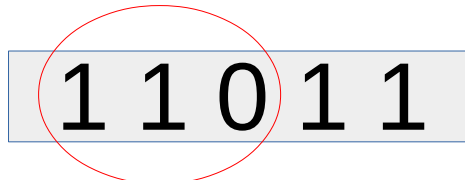
00000	a
00001	b
00010	c
00011	d
00100	e
00101	f
00110	g
00111	h
01000	i
01001	j
01010	k
01011	l
01100	m
01101	n
01110	o
01111	p

Memoria lógica

000	101
001	110
010	001
011	010

Page Table

Dirección
física



01000	m
01001	n
01010	o
01011	p
01100	
01101	
01110	
01111	
10000	
10001	
10010	
10011	
10100i	a
10101	b
10110	c
10111	d
11000i	e
11001	f
11010	g
11011	h
11100	
11101	
11110	
11111	

Memoria
física

Direccionamiento

Dirección
lógica

0	0	1	1	1
p			d	

00000	a
00001	b
00010	c
00011	d
00100	e
00101	f
00110	g
00111	h
01000	i
01001	j
01010	k
01011	l
01100	m
01101	n
01110	o
01111	p

Memoria lógica

000	101
001	110
010	001
011	010

Page Table

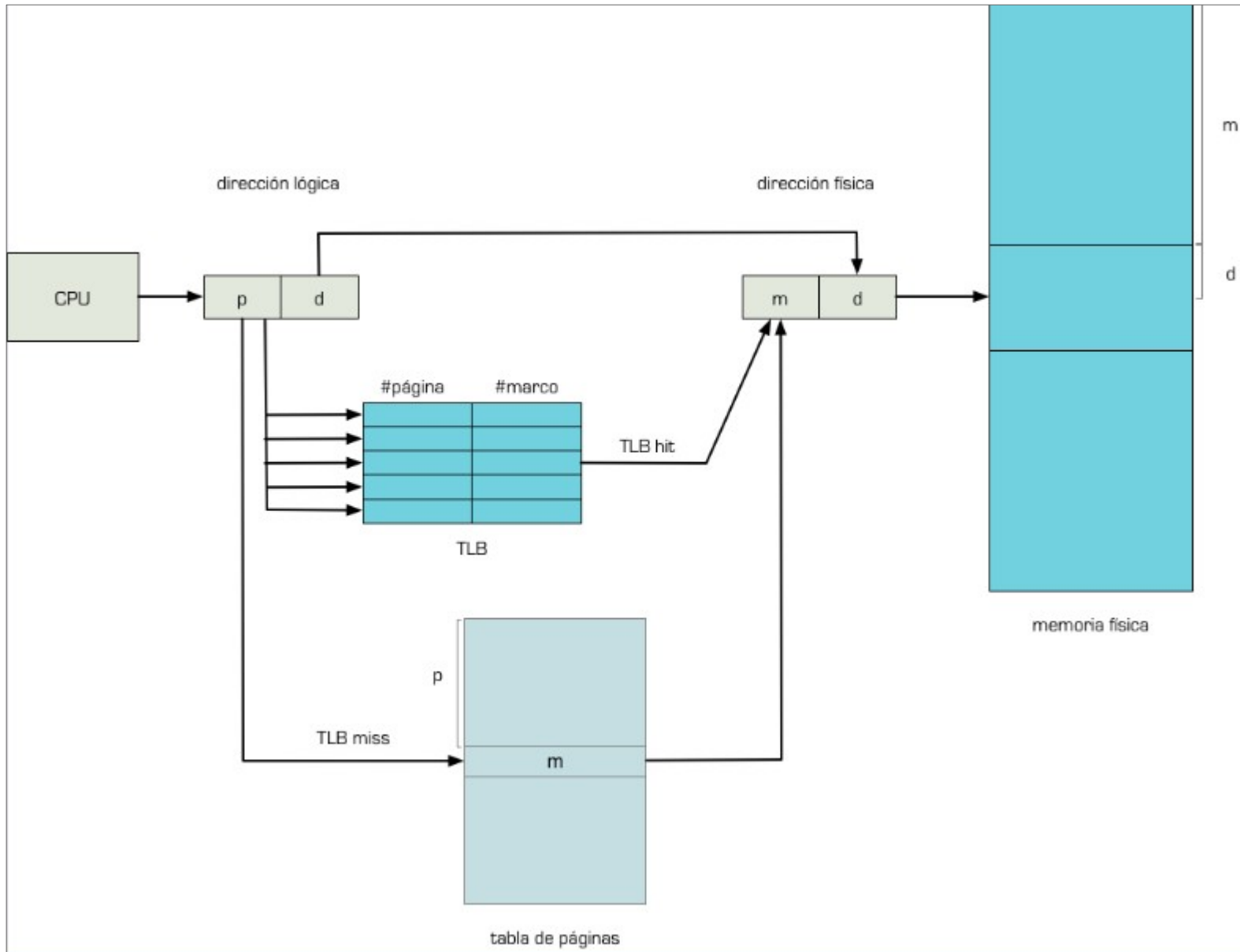
Dirección
física

1	1	0	1	1
---	---	---	---	---

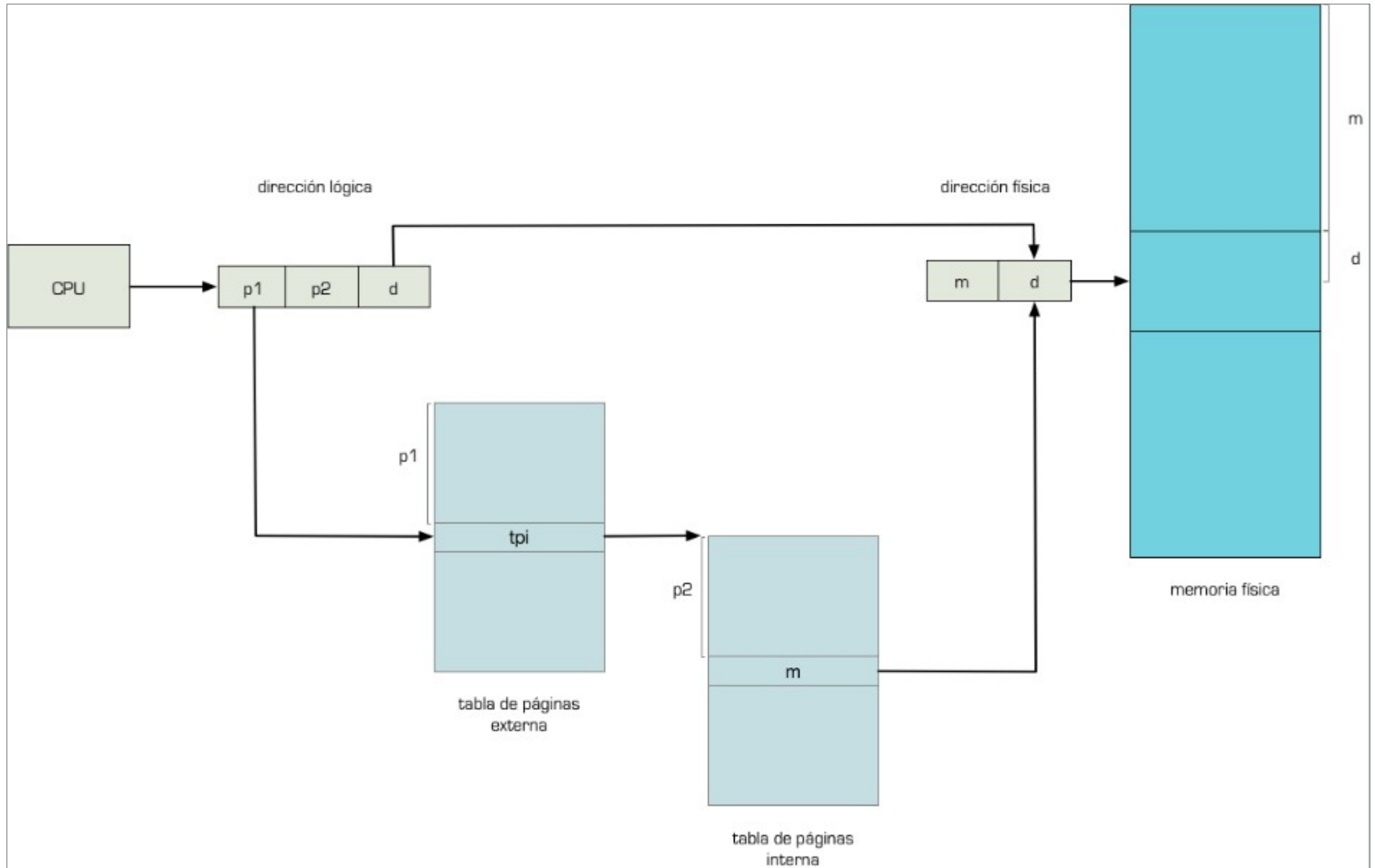
01000	m
01001	n
01010	o
01011	p
01100	
01101	
01110	
01111	
10000	
10001	
10010	
10011	
10100i	a
10101	b
10110	c
10111	d
11000i	e
11001	f
11010	g
11011	h
11100	
11101	
11110	
11111	

Memoria
física

TLB (Translation lookaside buffer)



ESQUEMA DE TRADUCCIÓN 2-NIVELES



ESQUEMA DE 3-NIVELES

número de página externo		número de página interno	desplazamiento
p1		p2	d
42		10	12
número de página externo 1-nivel	número de página interno 2-nivel	número de página interno	desplazamiento
p1	p2	p2	d
32	10	10	12

Paginación

- + Soporta **múltiples programas** ejecutando a la vez
- + Los programas de usuario tienen un **espacio de memoria** definido (protección)
- Requiere traducción **Dirección Lógica → Dirección Física**
 - + MMU Hardware
- Page Table: Requiere 2 accesos a memoria
 - + **Optimización TLB**
- + No Genera **Fragmentación Externa**
- Genera **Fragmentación Interna (muy poca)**