Exercise 4

Given a computer with paged virtual memory and a cache memory, with the following characteristics:

- 4GB Virtual Memory system.
- 64MB Physical Memory with 8MB pages, which uses a FIFO replacement policy.
- 256KB direct-mapped Physical Data Cache, with a block size of 64KB.

The disk stores 20 photo albums (Alb1 … Alb20). It is executed a program which randomly selects one of the albums, and shows in the screen random photos one after the other. Each album occupies 2MB and each photo 64KB. Alb1 starts in virtual address **0x02800000, and the other albums are stored sequentially after Alb1.**

1)Obtain the virtual and physical address formats (both for the memory and for the cache.

2)Obtain the albums which are stored in Main Memory, given the following contents of Page Table, TLB and $.

Tabla Páginas

Nº fila	V	FIFO	Nº pag. física
.	0	.	.
.	.	.	.
.	0	.	.
5	1	1	6
6	1	0	0
.	0	.	.
.	0	.	.
9	1	2	1
.	0	.	.
.	.	.	.
.	0	.	.

TLB

FIFO	V	Nº pag. virtual	Nº pag. física
0	1	5	6
1	1	9	1

Caché

Etiqueta	Bloque
0xD0	0
0xD5	1
0xD0	2
0x27	3

All data is expressed in decimal, except the Cache Tag. V is the valid bit. The page to replace corresponds to the one with a FIFO bit equal to 0.

1)What are the initial physical addresses for the albums of the page that was brought to Main Memory in the last place?

2)The photo at virtual address 0x03400000 is accessed. Say the order of accesses, whether there is a miss or a hit, and the final content of the tables.

Exercise 4

Given a computer with paged virtual memory and a cache memory, with the following characteristics:

- 4GB Virtual Memory system.
- 64MB Physical Memory with 8MB pages, which uses a FIFO replacement policy.
- 256KB direct-mapped Physical Data Cache, with a block size of 64KB.

The disk stores 20 photo albums (Alb1 … Alb20). It is executed a program which randomly selects one of the albums, and shows in the screen random photos one after the other. Each album occupies 2MB and each photo 64KB. Alb1 starts in virtual address **0x02800000, and the other albums are stored sequentially after Alb1.**

1)Obtain the virtual and physical address formats (both for the memory and for the cache.

1) Obtain the virtual and physical address formats (both for the memory and for the cache).

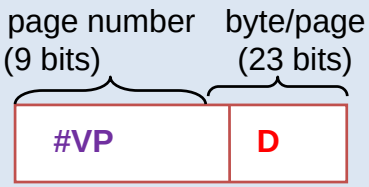
Assume a computer with 4GB Virtual Memory system.
64MB Physical Memory with 8MB pages, which uses a FIFO replacement policy.
256KB direct-mapped Physical Data Cache, with a block size of 64KB.

Address space: 4 GB = 2^{32} bytes

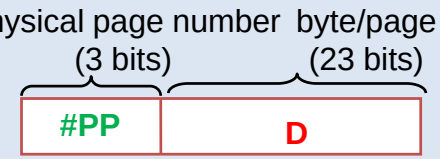
Page size: 8 MB = 2^{23} bytes

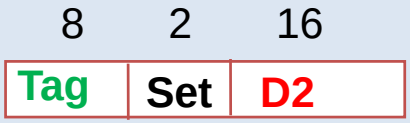
Memory: 64 MB = 2^{26} bytes

Fields of the virtual address



Fields of the physical address





$$\frac{256\ KB}{64\ KB/block} = 4\ blocks$$

Exercise 4

Given a computer with paged virtual memory and a cache memory, with the following characteristics:

- 4GB Virtual Memory system.
- 64MB Physical Memory with 8MB pages, which uses a FIFO replacement policy.
- 256KB direct-mapped Physical Data Cache, with a block size of 64KB.

The disk stores 20 photo albums (Alb1 … Alb20). It is executed a program which randomly selects one of the albums, and shows in the screen random photos one after the other. Each album occupies 2MB and each photo 64KB. Alb1 starts in virtual address **0x02800000, and the other albums are stored sequentially after Alb1.**

1)Obtain the virtual and physical address formats (both for the memory and for the cache.

2)Obtain the albums which are stored in Main Memory, given the following contents of Page Table, TLB and $.

Tabla Páginas

Nº fila	V	FIFO	Nº pag. física
. . .	0 . 0	. . .	. . .
5	1	1	6
6	1	0	0
. .	0 0	. .	. .
9	1	2	1
. . .	0 . 0	. . .	. . .

TLB

FIFO	V	Nº pag. virtual	Nº pag. física
0	1	5	6
1	1	9	1

Caché

Etiqueta	Bloque
0xD0	0
0xD5	1
0xD0	2
0x27	3

All data is expressed in decimal, except the Cache Tag. V is the valid bit. The page to replace corresponds to the one with a FIFO bit equal to 0.

2) Obtain the albums which are stored in Main Memory, given the following contents of Page Table, TLB and $.

Tabla Páginas

Nº fila	V	FIFO	Nº pag. física
.	0	.	.
.	.	.	.
.	0	.	.
5	1	1	6
6	1	0	0
.	0	.	.
.	0	.	.
9	1	2	1
.	0	.	.
.	.	.	.
.	0	.	.

TLB

FIFO	V	Nº pag. virtual	Nº pag. física
0	1	5	6
1	1	9	1

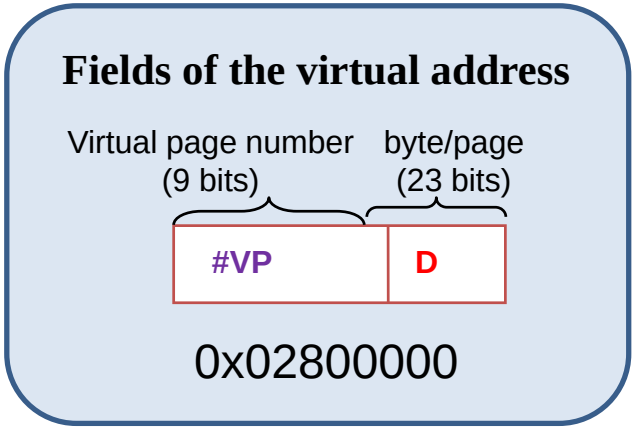
Caché

Etiqueta	Bloque
0xD0	0
0xD5	1
0xD0	2
0x27	3

Pages are in main memory if the valid bit in the page table is 1.
So, pages 5,6 and 9 are in main memory.

2) Obtain the albums which are stored in Main Memory, given the following contents of Page Table, TLB and $.

Each album occupies 2MB and each photo 64KB. Alb1 starts in virtual address **0x02800000, and the other albums are stored sequentially after Alb1.**



Page size: 8 MB = 2^{23} bytes

$$\frac{8\ MB/page}{2\ MB/album} = 4\ albums/page$$

All 20 albums take up 5 pages.

0000 0010 1000 0000 0000 0000 0000 0000

Alb1 starts in the initial address of virtual page 5

Pages 5,6 and 9 are in main memory.

	Virtual page
Alb1-4	0x05
Alb5-8	0x06
Alb9-12	0x07
Alb13-16	0x08
Alb17-20	0x09

Exercise 4

Given a computer with paged virtual memory and a cache memory, with the following characteristics:

- 4GB Virtual Memory system.
- 64MB Physical Memory with 8MB pages, which uses a FIFO replacement policy.
- 256KB direct-mapped Physical Data Cache, with a block size of 64KB.

The disk stores 20 photo albums (Alb1 … Alb20). It is executed a program which randomly selects one of the albums, and shows in the screen random photos one after the other. Each album occupies 2MB and each photo 64KB. Alb1 starts in virtual address **0x02800000, and the other albums are stored sequentially after Alb1.**

1)Obtain the virtual and physical address formats (both for the memory and for the cache.

2)Obtain the albums which are stored in Main Memory, given the following contents of Page Table, TLB and $.

Tabla Páginas

Nº fila	V	FIFO	Nº pag. física
.	0	.	.
.	.	.	.
.	0	.	.
5	1	1	6
6	1	0	0
.	0	.	.
.	0	.	.
9	1	2	1
.	0	.	.
.	.	.	.
.	0	.	.

TLB

FIFO	V	Nº pag. virtual	Nº pag. física
0	1	5	6
1	1	9	1

Caché

Etiqueta	Bloque
0xD0	0
0xD5	1
0xD0	2
0x27	3

All data is expressed in decimal, except the Cache Tag. V is the valid bit. The page to replace corresponds to the one with a FIFO bit equal to 0.

3) What are the initial physical addresses for the albums of the page that was brought to Main Memory in the last place?

3) What are the initial physical addresses for the albums of the page that was brought to Main Memory in the last place?

Tabla Páginas

Nº fila	V	FIFO	Nº pag. física
.	0	.	.
.	.	.	.
.	0	.	.
5	1	1	6
6	1	0	0
.	0	.	.
.	0	.	.
9	1	2	1
.	0		.
.	.	.	.
.	0	.	.

TLB

FIFO	V	Nº pag. virtual	Nº pag. física
0	1	5	6
1	1	9	1

Caché

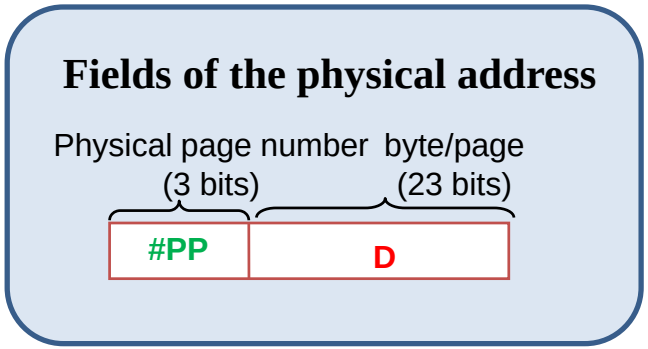
Etiqueta	Bloque
0xD0	0
0xD5	1
0xD0	2
0x27	3

The page to replace corresponds to the one with a FIFO bit equal to 0.

To know the order in which pages were brought to memory we have to read the FIFO bits: 0 for the oldest, 2 for the newest.

3) What are the initial physical addresses for the albums of the page that was brought to Main Memory in the last place?

Albums in page 9: Alb17, Alb18, Alb19, Alb20.
Number of physical page: 1
Size of each album: 2MB= 2^{21} B



	Initial physical address
Alb17	001 000 0000 0000 0000 0000 0000 = 0x0800000
Alb18	001 010 0000 0000 0000 0000 0000 = 0x0A00000
Alb19	001 100 0000 0000 0000 0000 0000 = 0x0C00000
Alb20	001 110 0000 0000 0000 0000 0000 = 0x0E00000

Exercise 4

Tabla Páginas

Nº fila	V	FIFO	Nº pag. física
.	0	.	.
.	.	.	.
.	0	.	.
5	1	1	6
6	1	0	0
.	0	.	.
.	0	.	.
9	1	2	1
.	0	.	.
.	.	.	.
.	0	.	.

TLB

FIFO	V	Nº pag. virtual	Nº pag. física
0	1	5	6
1	1	9	1

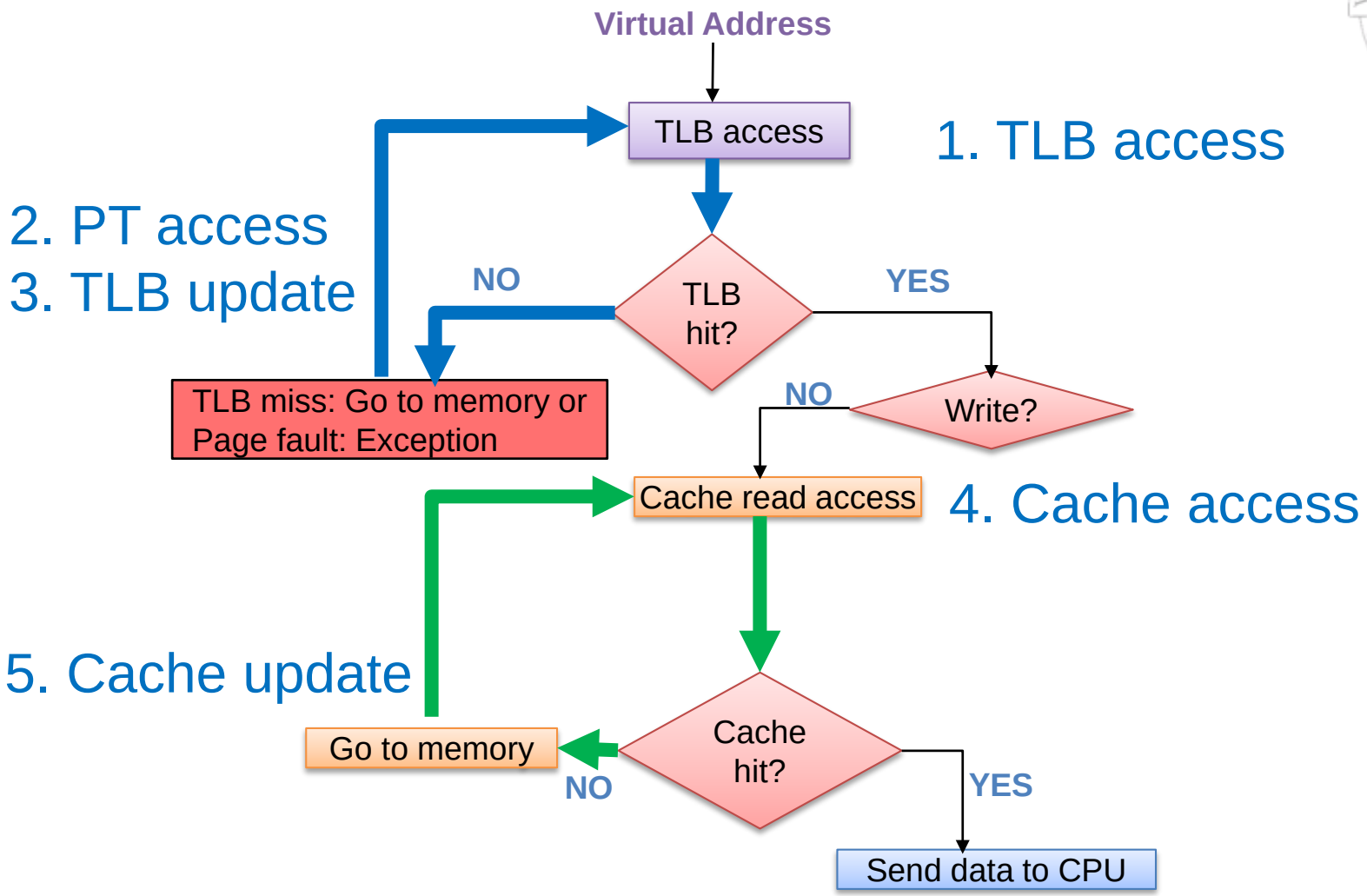
Caché

Etiqueta	Bloque
0xD0	0
0xD5	1
0xD0	2
0x27	3

All data is expressed in decimal, except the Cache Tag. V is the valid bit. The page to replace corresponds to the one with a FIFO bit equal to 0.

3) What are the initial physical addresses for the albums of the page that was brought to Main Memory in the last place?
4) The photo at virtual address 0x03400000 is accessed. Say the order of accesses, whether there is a miss or a hit, and the final content of the tables.

Physical cache



4) Say the order of accesses, whether there is a miss or a hit, and the final content of the tables

First: look for address translation in TLB

Page size: 8 MB = 2^{23} bytes

Virtual address: 0x03400000

0000 0011 0 100 0000 0000 0000 0000 0000

Virtual page number: 0x06

FIFO	V	Virtual page	Physical page
0	1	5	6
1	1	9	1

TLB miss: virtual page 6 is not in the buffer.

We need to look for the translation in the Page Table.

4) Say the order of accesses, whether there is a miss or a hit, and the final content of the tables

Second: look for address translation in the Page Table

Tabla Páginas

Nº fila	V	FIFO	Nº pag. física
.	0	.	.
.	.	.	.
.	0	.	.
5	1	1	6
6	1	0	0
.	0	.	.
.	0	.	.
9	1	2	1
.	0	.	.
.	.	.	.
.	0	.	.

Virtual address: 0x03400000

0000 0011 0 100 0000 0000 0000 0000 0000

Virtual page number: 0x06

00 0 100 0000 0000 0000 0000 0000

Physical addr.: 0x0400000

PT hit: virtual page 6 is in memory in physical page 0.

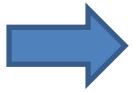
We need to update TLB and look for the data in the cache

4) Say the order of accesses, whether there is a miss or a hit, and the final content of the tables

Third: TLB update

Virtual page number: 0x06 Physical page number: 0

FIFO	V	Virtual page	Physical page
0	1	5	6
1	1	9	1



FIFO	V	Virtual page	Physical page
1	1	6	0
0	1	9	1

Replacement in TLB: the first entry changes

4) Say the order of accesses, whether there is a miss or a hit, and the final content of the tables

Fourth: look for address the data in the cache

Physical addr.: 0x040 0000

Tag	Set
0xD0	0
0xD5	1
0xD0	2
0x27	3

Cache miss: block 0x10 is not stored in set 0.

We need to look for the data in memory and bring it to the cache

4) Say the order of accesses, whether there is a miss or a hit, and the final content of the tables

Fifth: bring block from memory to the cache

Physical addr.: 0x040 0000
Size: 64 KB

Tag: 0x10 Set: 0

Tag	Set
0xD0	0
0xD5	1
0xD0	2
0x27	3

Tag	Set
0x10	0
0xD5	1
0xD0	2
0x27	3

Replacement in direct mapped cache: the block in set 0 is replaced