

Ejercicio 1

Pags 4K → 12 bits offset

↑
10 bits DIR

↓
10 bits TP

Una PC con procesador intel de 32 bits, que ya está corriendo un mini sistema operativo tiene la siguiente distribución de memoria física.

Se aclara que:

- El Programa A está corriendo en este momento y ocupa 8 Kbytes contiguos.
- El Programa B ocupaba el mismo espacio y lugar físico en memoria que el Programa A pero fue bajado a disco por falta de espacio. El programa B todavía no terminó de correr.
- La tabla de páginas 1 se utiliza para el programa A.
- La tabla de páginas 2 se utiliza para el programa B.
- La primera dirección de memoria virtual del Programa A es: AFA22000h
- La primera dirección de memoria virtual del Programa B es: CAFE0000h

10 A = 1010

11 B = 1011

12 C = 1100

13 D = 1101

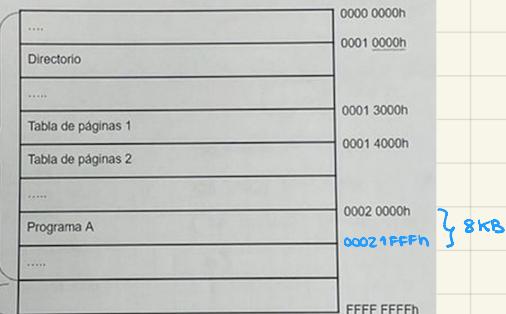
14 E = 1110

15 F = 1111

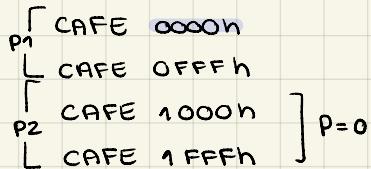
2GB de RAM

256 MB de ROM

Mapa de memoria física



programa B → 8 KB tambien ANTES estaba en el mismo lugar físico



Ejercicio 2

to add suffix
to add prefix

Con un procesador Intel de 32 bits se desea crear el siguiente sistema:

- a) Una memoria ROM de 256 KB x 8 a partir de la dirección 0x00000000 *me res
ui tecido*

b) Un dispositivo de reloj encapsulado en SOT-8 al cual se accede a través de 4 registradores de 8 bits.

c) Se desea también concretar el MAXIMO posible de memoria RAM para lo que se dispone de los siguientes integrados *como 16Mbit con 3 integraciones*
como un solo.

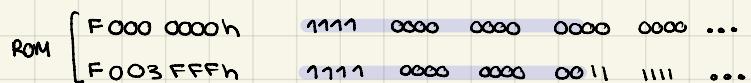
 - i) 2GB x 8 $\rightarrow 2^{31}$
 - ii) 1GB x 8 $\rightarrow 2^{30}$
 - iii) 256KB x 8
 - iv) 128KB x 8
 - v) 64KB x 8
 - vi) 16KB x 8 $\rightarrow 2^{29}$

*Sin programar
a secundaria*

Se pide lo siguiente

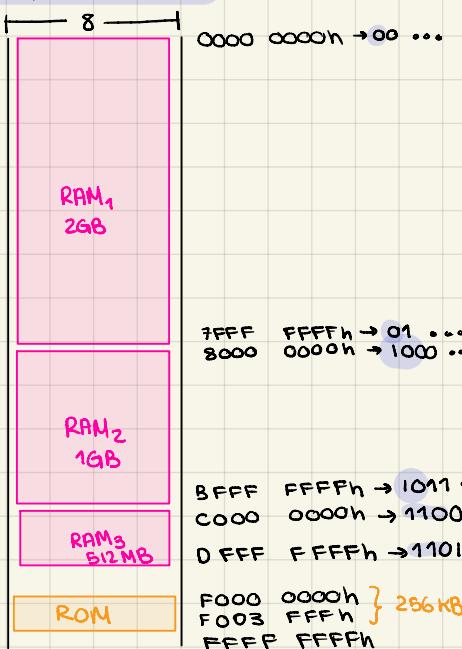
- 1) Represente el mapa virtual y real de memoria y de E/S resultante.
 - 2) Realice la decodificación completa de los nuevos componentes.
 - 3) Escriba un programa assembler que copie los 4 registros del teclado a partir de la dirección lineal ACADE123h.

$$\text{ROM } 256\text{ KB} \times 8 \rightarrow 2^{18} = 262144$$



teclado → 4 registro x 8 → 0000 0000h

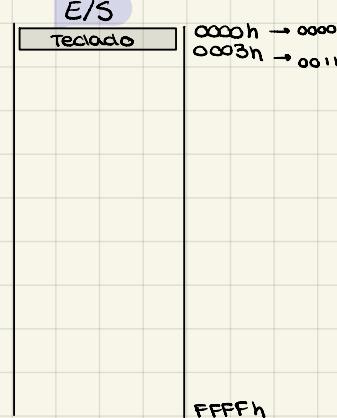
Mapa de Memoria



RAM → 3 integrados maximo espacio

elijo 1GB 2GB y 512 MB

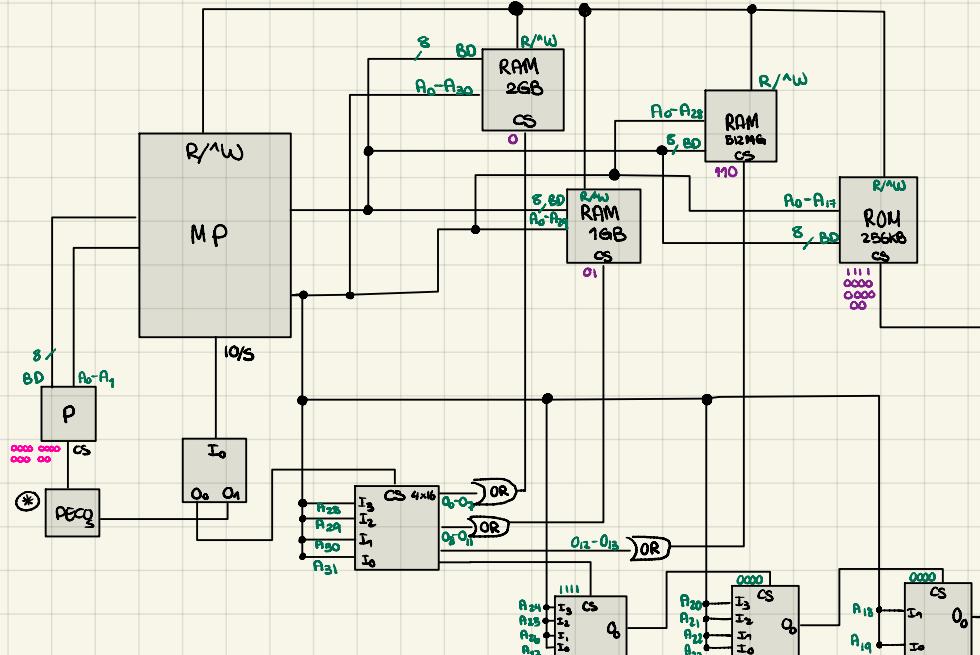
$$2^{30} + 2^{31} + 2^{29} = 2^{29}(2 + 2^2 + 1) = 3584 \text{ MB}$$



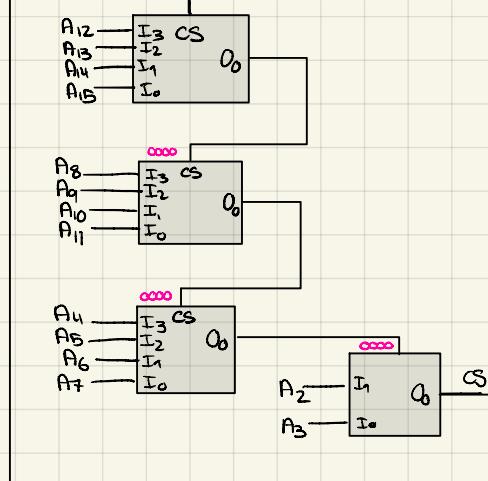
64KB tamaño de mapa
de entrada y salida.

$$2^{16} = \text{direcciones}$$

9



* 4 decós
periférico



(3) ASM periferico copie registros en ACAD E123h MV.

```
mov EBX, ACAD E123h ; destino.  
; AL para ver registros.  
mov DX, 0000h ; puntero a periferico mapa E/S.
```

• copy_registers:

```
cmp DX, 0004h  
je .end  
in AL, DX ; 1er registro.  
mov [EBX], AL ; guardo copia  
  
inc EBX ; aumento puntero. para proximo registro.  
inc DX ; prox registro.  
jmp .copy_registers.
```

Ejercicio 1 - (4 puntos)

En el presente ejercicio se le pide para todo el sistema descripto:

- Dibujar el mapa de memoria física.
- Realizar la decodificación completa según especificaciones brindadas a continuación.

Se tiene un procesador Intel de 32 bits cuyo bus de direcciones tiene 26 líneas. El bus de datos es de 16 líneas.

El valor inicial del registro IP al arrancar el sistema es IP = 0x FFFF FFF0.

La memoria Flash donde se encontrará el firmware/BIOS que va a correr al inicio es de 512K x 16.

Este firmware/BIOS particular, ya se encuentra escrito y no se modificará: se utilizará tal cual se encuentra. Sin embargo, en dicho código se hace uso de un stack de 256K de tamaño máximo. Le indican que el tope del stack se encuentra en la dirección 0x 0200 0000. Para resolver el requerimiento del stack, le indican que dispondrá de una RAM de 256K x 16.

El resto del sistema se compone de 2 bancos de RAM para permitir selección y ampliación de la memoria del sistema:

- 1 banco de 4M conformado por 2 RAM de 4M x 8 cada uno.
- 1 banco de 8M x 16

Estos pueden ser posicionados en cualquier orden, pero contiguos y localizados a partir de la dirección inicial del mapa de memoria física.

Le sugieren:

- Utilizar sólo decodificadores 4 a 16.
- Utilizar un máximo de 6 unidades.
- No utilizar compuertas lógicas absolutamente para nada.

Otras indicaciones que le proporcionan:

- En caso de considerar que requiere usar otros componentes / cantidades / alterar el mapa de memoria solicitado, puede desvirtuarse de la especificación, teniendo en cuenta que dicha desviación tendrá peso en la evaluación del circuito decodificador presentado. En tal caso, adicionalmente, justificar el cambio.
- Se quiere evitar la existencia de imágenes en el mapa de memoria.
- Se pide dibujar todas las señales y buses requeridos para la decodificación. Ejemplo: puede especificar los buses mediante indicación del rango de señales que abarca. Tenga en cuenta que la omisión de elementos en el diseño presentado, será tenida en cuenta en función de la importancia que se le asigne.

CAMBIO 26 líneas de bus de address ya que el IP = 0x FFFF FFF0
no entra en el CPU entonces lo modifiqué ya que busco evitar imágenes necesito que la memoria física sea continua

$$\text{RAM}_1 \rightarrow 256\text{K} \times 16 \quad \text{tope del stack } 0x 0200 0000$$

$$\downarrow \\ 2^{18} = 262144 = 0004\ 0000h$$

$$0x 0200 0000 - 0x 0203 FFFFh$$

26 linea

$$\downarrow \\ 0000\ 0000h = 0 3FF\ FFFFh$$

$$2\text{RAM}_2 \rightarrow 4M \times 8 \quad \leftarrow \text{comparte BD}$$

$$\downarrow \\ 2^{22} = 4194304 = 0040\ 0000h$$

$$0020\ 4000h - 0060\ 3FFFh$$

Mapeo de memoria

0000\ 0000h

$$\text{RAM}_3 \rightarrow 8M \times 16$$

$$\downarrow \\ 2^{23} = 8388608 = 0080\ 0000h$$

$$0060\ 4000h - 00E0\ 3FFF$$

16

$$\text{ROM} \rightarrow 512\text{K} \times 16 \rightarrow 2^{19} = 524288d = 8\ 0000h$$

$$00E0\ 4000h - 00E83FFFh$$

$$\text{IP} = 0x 00E0 4000h$$

03FF\ FFFFh

Ejercicio 2 - (4 puntos)

Se dispone de un sistema basado en un procesador de 32 bits al cual se la decodificado 256M de memoria RAM a partir de la dirección 0x40000000h, 128 MB de memoria ROM a partir de la dirección 0xF8000000h y una memoria de video de 16 M a partir de la dirección 0x80000000h

El procesador se inicializa en modo flat y se ubica la IDT a partir de la dirección 0x00000000h y se desea ejecutar el siguiente programa para actualizar la memoria de video

```

MOV EAX,A0000000H
MOV EBX,B0000000H
MOV ECX,01000000 H
SIGO: MOV DL,[EAX]
      MOV byte [EBX],DL
      INC EAX
      INC EBX
      DEC ECX
      CMP ECX,0
      JNE SIGO
  
```

Desarrolle el mapa virtual y real con todos los elementos , entradas de directorio , tablas de páginas y valores de registros para el correcto funcionamiento de este programa.

Ejercicio 3 - (2 puntos)

Dado el siguiente código de ASM (con representación hexa agregada) en ARM, explique 3 características significativas que diferencien a los procesadores ARM con los Intel en cuanto a instrucciones se refiere.

	0000 0000	ROM $\rightarrow 2^{27} = 134217728 \rightarrow 0800\ 0000h$
RAM	4000 0000h 4FFF FFFFh	RAM $\rightarrow 2^{28} = 268435456 \rightarrow 1000\ 0000h$
video	8000 0000h 80FF FFFFh	MV $\rightarrow 16M = 2^{24} = 16777216 \rightarrow 0100\ 0000h$
ROM	F800 0000h FFFF FFFFh	