



# Arquitectura de Computadoras

---

Espacio de direcciones de E/S

Curso 2020

Prof. Jorge Runco



# E/S aislado y mapeado en memoria

---

- Hay 2 métodos para hacer la interface del espacio de E/S :
- E/S aislado
- E/S mapeado en memoria



# E/S aislado (1)

---

- Técnica utilizada por sistemas basados en procesadores Intel.
- Aislado: las posiciones de E/S están “separadas” de la memoria del sistema, en un espacio distinto de direcciones.
- Las direcciones de E/S llamadas puertos están separadas de memoria.



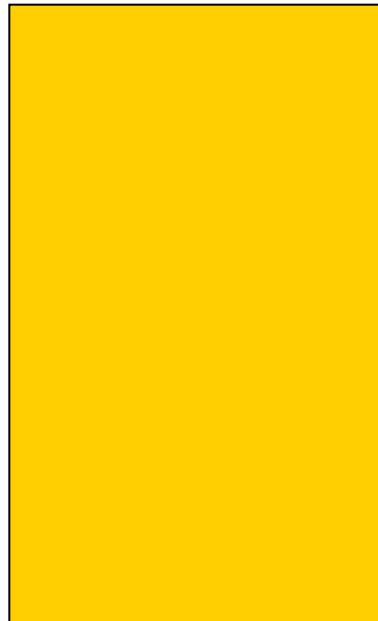
# E/S aislado (2)

---

## MEMORIA

FFFFFFH

00000H



## E/S

FFFFH

0000H





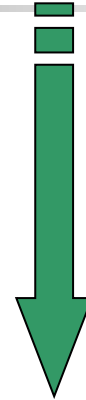
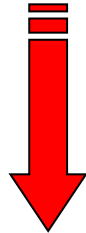
## E/S aislado (3)

---

- Ventaja : todo el espacio de memoria está ocupado por la misma.
- Desventaja : para transferir datos entre el  $\mu p$  y e/s tenemos que usar instrucciones especiales como in y out.

MOV [0378H], AL

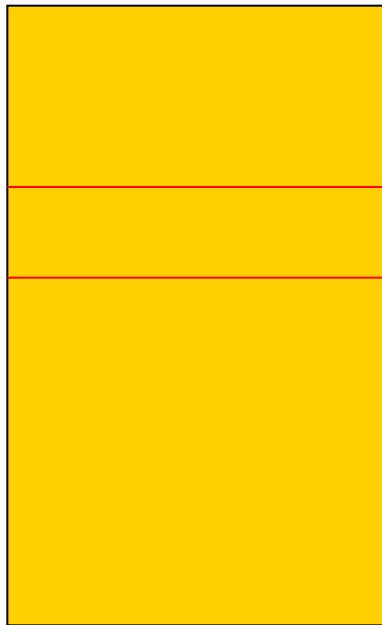
OUT 0378H, AL



FFFFFFH

0378H

00000H



MEMORIA

0000H

0378H

FFFFH



E/S



## E/S aislado (4)

---

- Recordemos, todas las instrucciones que vimos eran de la forma :
  - ❖ MOV mem, reg
  - ❖ MOV reg, mem
  - ❖ MOV reg, reg
  
- Son entre el procesador y memoria



## E/S aislado (5)

---

□ Ahora tenemos instrucciones :

- ✓ **IN AL, puerto** Lee un byte de la dirección de e/s puerto.
- ✓ **IN AL, DX** Lee un byte de la dirección de e/s almacenada en DX.
- ✓ **OUT puerto, AL** Escribe un byte de AL en la dirección de e/s puerto.
- ✓ **OUT DX, AL** Escribe un byte de AL en la dirección de e/s contenida en DX.





## E/S aislado (6)

---

- Cuando la UC decodifica OUT ó IN, activa las líneas del bus de control  
iow=input/output write  
ior=input/output read
- Cuando la UC decodifica MOV, activa las líneas del bus de control  
mwr=memory write mrd=memory read
- Se puede ver en el simulador.



# E/S mapeada en memoria (1)

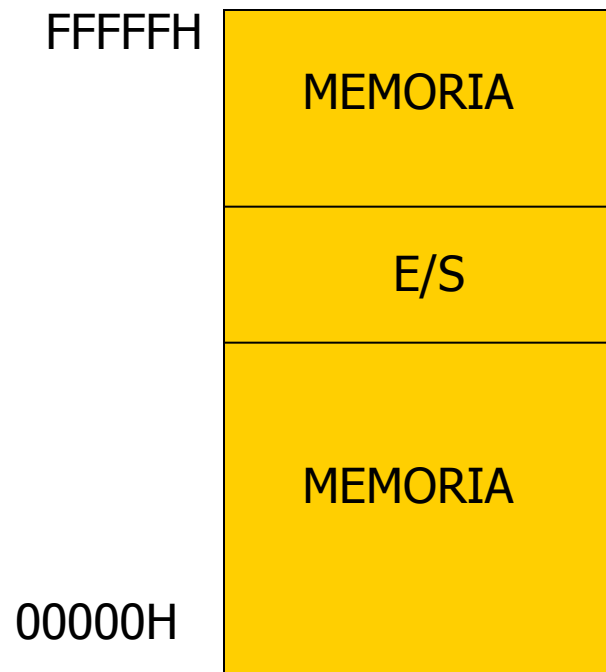
---

- ❑ Las direcciones de e/s están mapeadas en las direcciones de memoria.
- ❑ Las direcciones de e/s pertenecen al espacio de memoria.
- ❑ No se distingue una posición de e/s de una posición de memoria.



# E/S mapeada en memoria (2)

---





## E/S mapeada en memoria (3)

---

- Ventaja : puedo usar todo el conjunto de instrucciones del  $\mu p$ , porque todas las posiciones son tomadas como direcciones. No hay instrucciones como IN y OUT.
- Desventaja : ocupa espacio de memoria



# PIO (1)

---

- 2 puertos paralelos de 8 bits: A y B.
- Se puede programar c/bit por separado como entrada ó salida.
- 4 reg internos de 8 bits :
  - 2 de datos, PA y PB.
  - 2 de control CA y CB, para programar los bits de PA y PB.



## PIO (2)

---

- Las direcciones de los registros son :

○ PA          30H

○ PB          31H

○ CA          32H

○ CB          33H



## PIO (3)

---

- CA= un bit en 0 selecciona como salida a la línea correspondiente en PA.
- CA= un bit en 1 selecciona como entrada a la línea correspondiente en PA.
- CB= controla de la misma manera a PB.

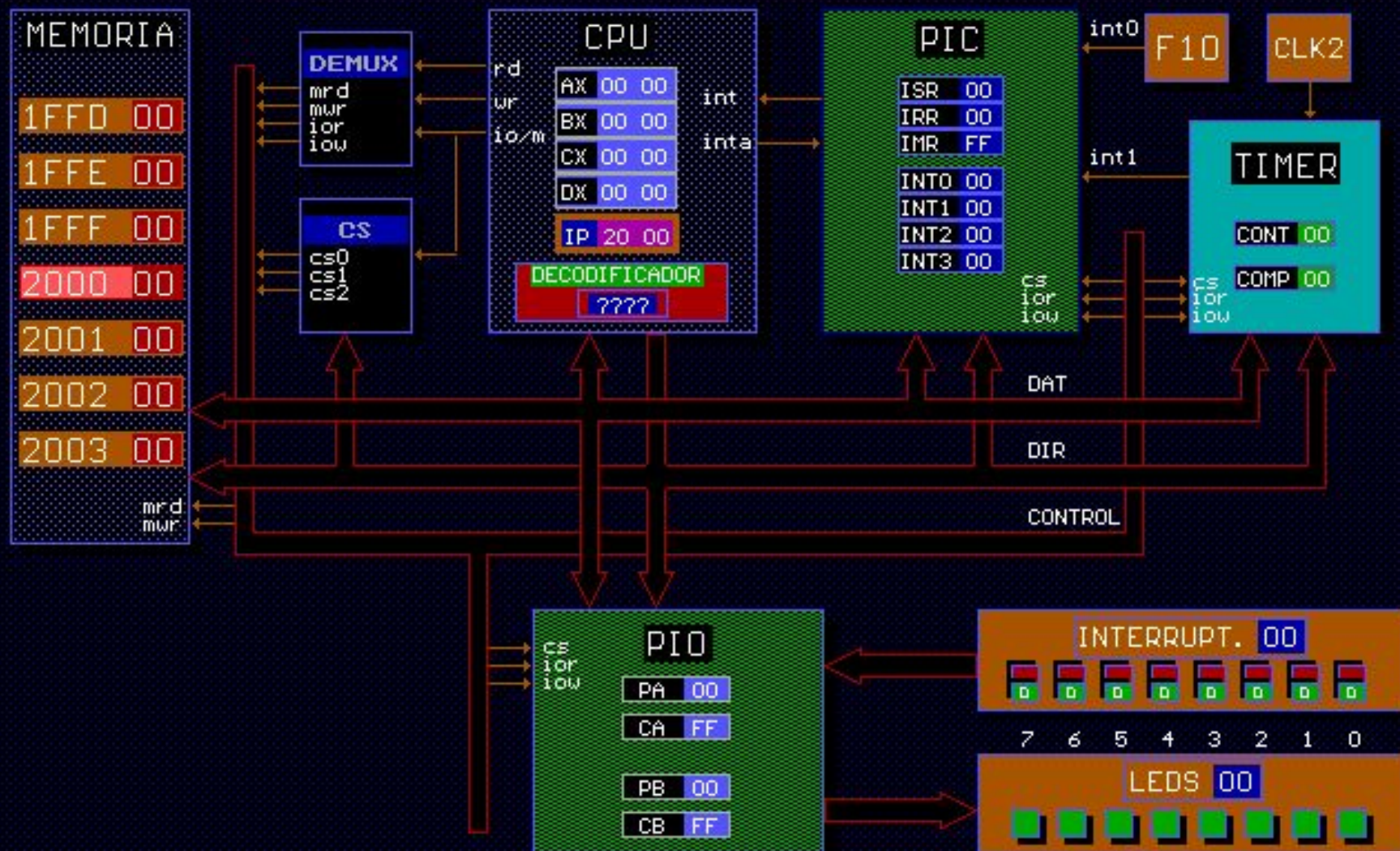


# Conexión con el sistema (1)

---

- Configuración 0 ( C0 )
- PA= entrada de los microinterruptores  
 $PA_7 \dots PA_0$
- CA= todos en 1, pues todas las líneas de PA son entradas.
- Los interruptores se controlan con las teclas 0...7



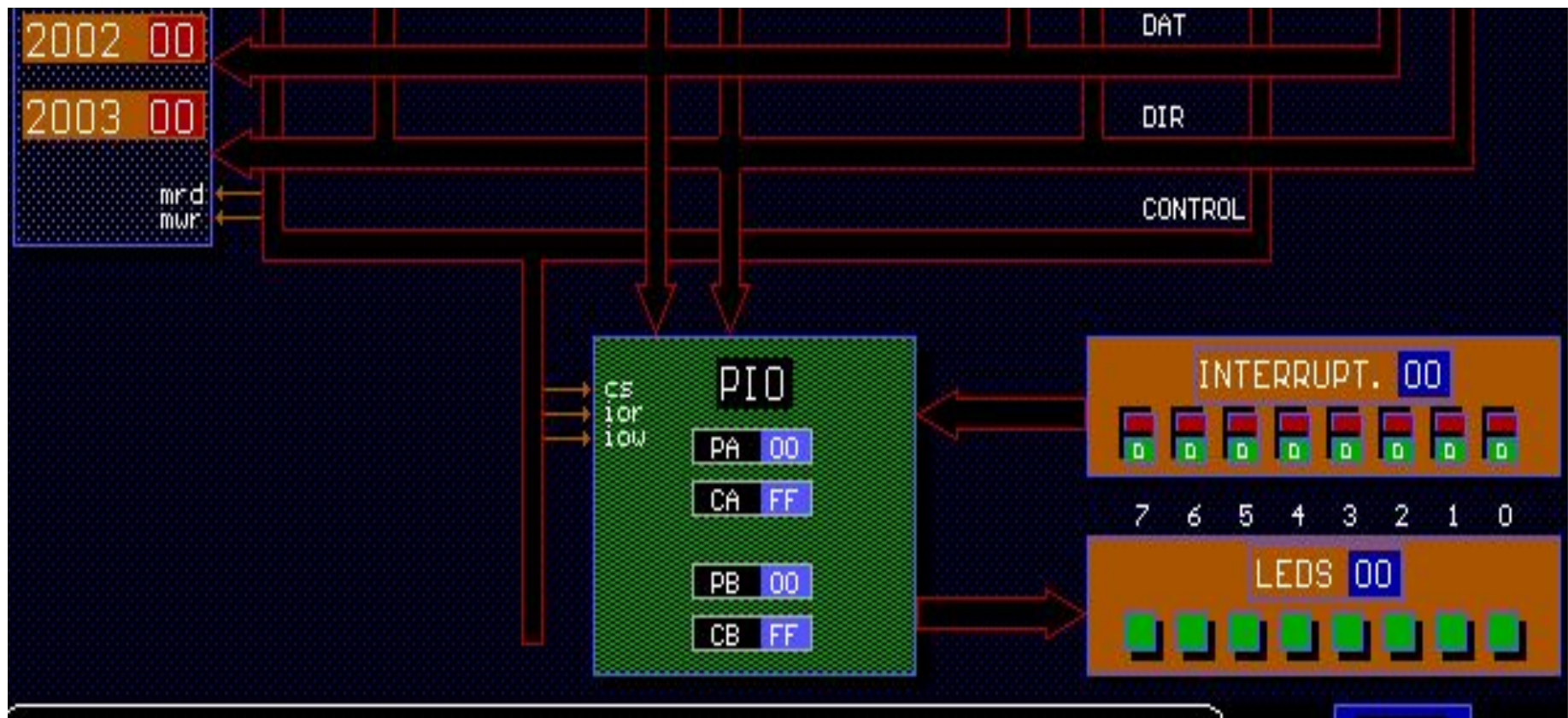


>p1  
>c1  
>c0  
>0\_

ESTADO

-  
P 01  
C 00  
V 15

# Ampliada





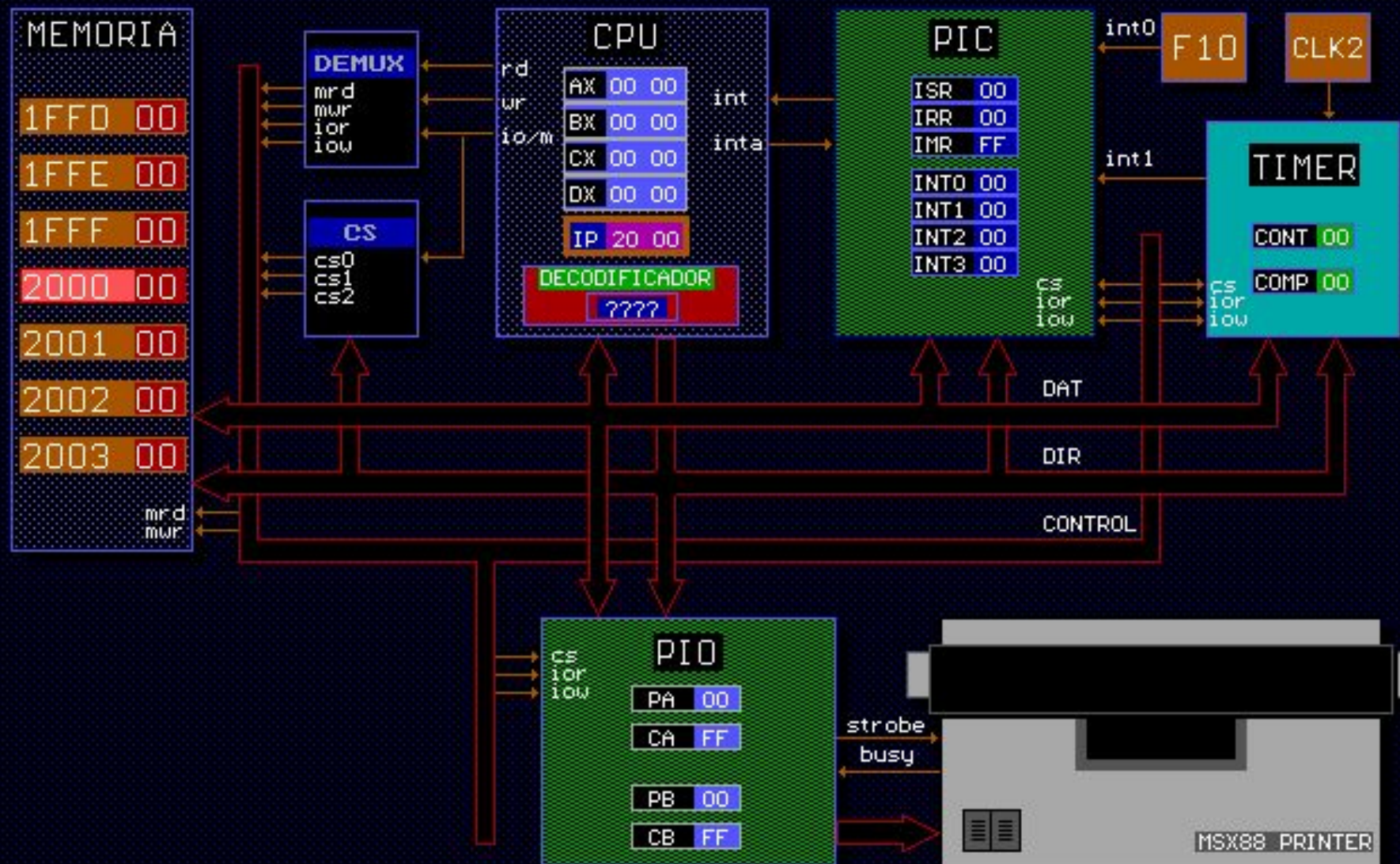
## Conexión con el sistema (2)

---

✓ Configuración 1 (C1) : interfaz con la impresora

- ❖ PA0 : línea BUSY
- ❖ PA1 : línea STROBE
- ❖ PB0....PB7 : línea de datos







```
>p1
>c1
>|
```

ESTADO

```
-
P 01
C 01
U 15
```

# HAND – Configuración 2

- Interfaz con la impresora : cumple temporización necesaria Centronics.
- Dos registros de 8 bits:
- DATO : D0.....D7       (40H)
- EST : registro de estado  (41H)

INT	X	X	X	X	X	STR	BUSY
-----	---	---	---	---	---	-----	------



# HAND (2)

---

## □ En lectura :

- BIT 0    0: línea BUSY desactivada  
            1: línea BUSY activada
- BIT 1    0: línea STROBE desactivada  
            1: línea STROBE activada
- BIT 2..6    No se usan
- BIT 7    0: no se activará la línea INT  
            1: se activará la línea INT cuando  
              BUSY no esté activa



## HAND (3)

---

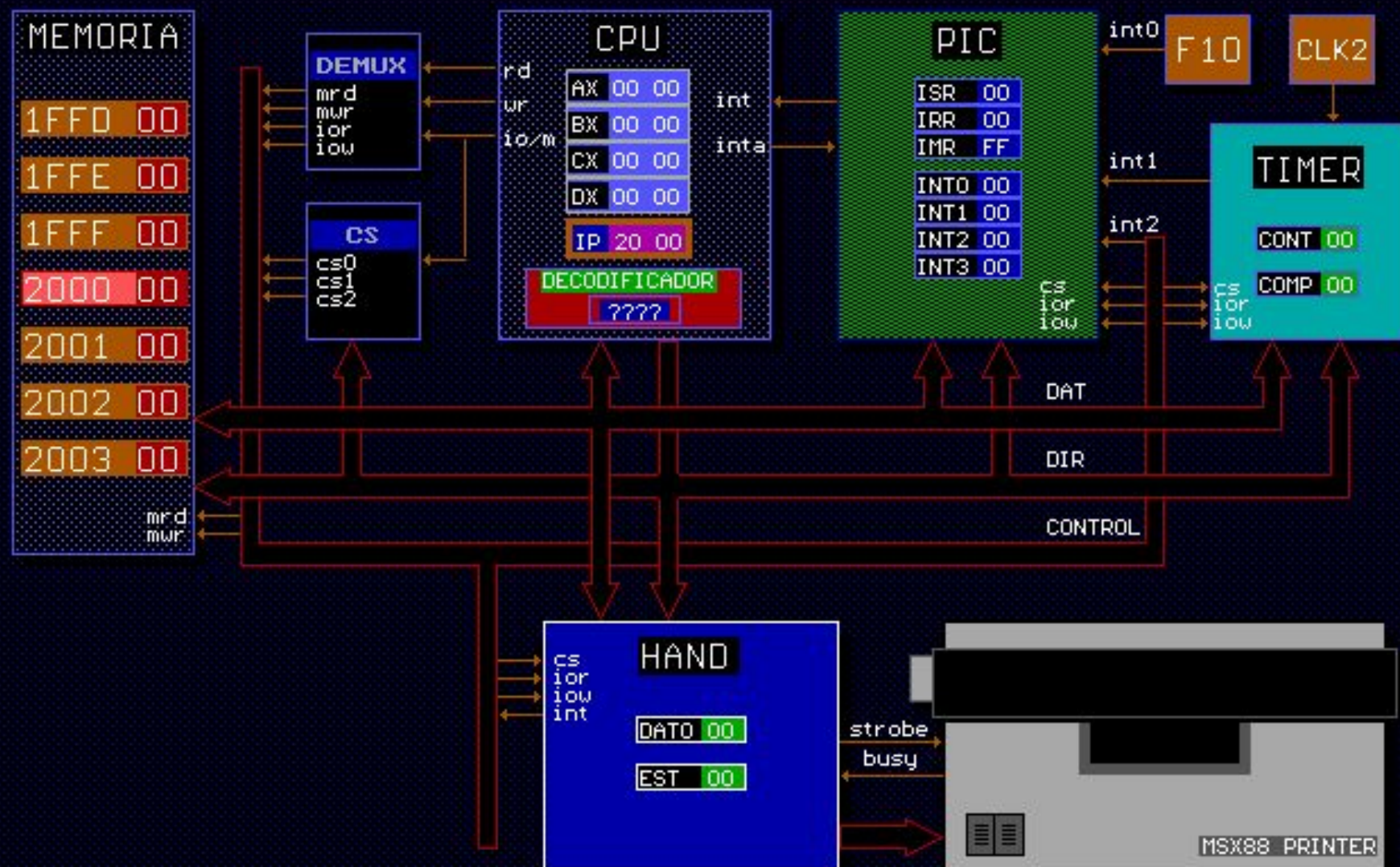
✓ En escritura :

✓ BIT 0....6 : no se usan

✓ BIT 7    0 : inhibe la activación de la  
línea INT

1 : se activa INT





>p1  
>c2  
>0

ESTADO

-

P 01

C 02

V 15



PA EQU 30H

PB EQU 31H

CA EQU 32H

CB EQU 33H

EJ1P4

ORG 2000H

MOV AL, 0FFH; PA TODAS ENTRADAS

OUT CA, AL

MOV AL, 0 ; PB TODAS SALIDAS


OUT CB, AL

POLL: IN AL, PA : LEE EL ESTADO DE LOS SWITCH

OUT PB, AL ;ENCIENDE LOS LEDS

JMP POLL

END



PIC	EQU	20H
TIMER	EQU	10H
PIO	EQU	30H
N_CLK	EQU	10

## EJ2P4 PROGRAMA PRINCIPAL

ORG 40 ; 4X10 DIRECCIÓN DEL SERVICIO

IP\_CLK DW RUT\_CLK

ORG 1000H

INICIO DB 0

CLI

MOV AL, 0FDH

OUT PIC+1, AL ; HABILITA INT1

MOV AL, N\_CLK

OUT PIC+5, AL ; REGISTRO DE PUNTERO INT1 ( VALOR 10 )

MOV AL, 1 ; CUENTA HASTA 1 CON CADA PULSO DE CK2

OUT TIMER+1, AL

MOV AL, 0

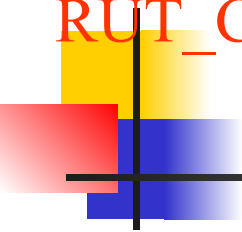
OUT PIO+3, AL ; TODAS SALIDAS EN PB

OUT PIO+1, AL ; TODOS LOS LEDS APAGADOS

OUT TIMER, AL

STI

LAZO: JMP LAZO



ORG 3000H

RUT\_CLK: INC INICIO ; INCREMENTA C/PULSO CLK2

CMP INICIO. 0FFH

~~JNZ LUCES; SE HACE HASTA FF~~

MOV INICIO, 0

LUCES: MOV AL, INICIO ; NUMERO CONTADO EN

OUT PIO+1, AL ; INICIO A LOS LEDS

MOV AL, 0

OUT TIMER, AL ; CUENTA DEL TIMER 0

MOV AL, 20H

OUT PIC, AL ; FIN DE INTERRUPCIÓN

IRET

END

SERVICIO DE INTERRUPCION