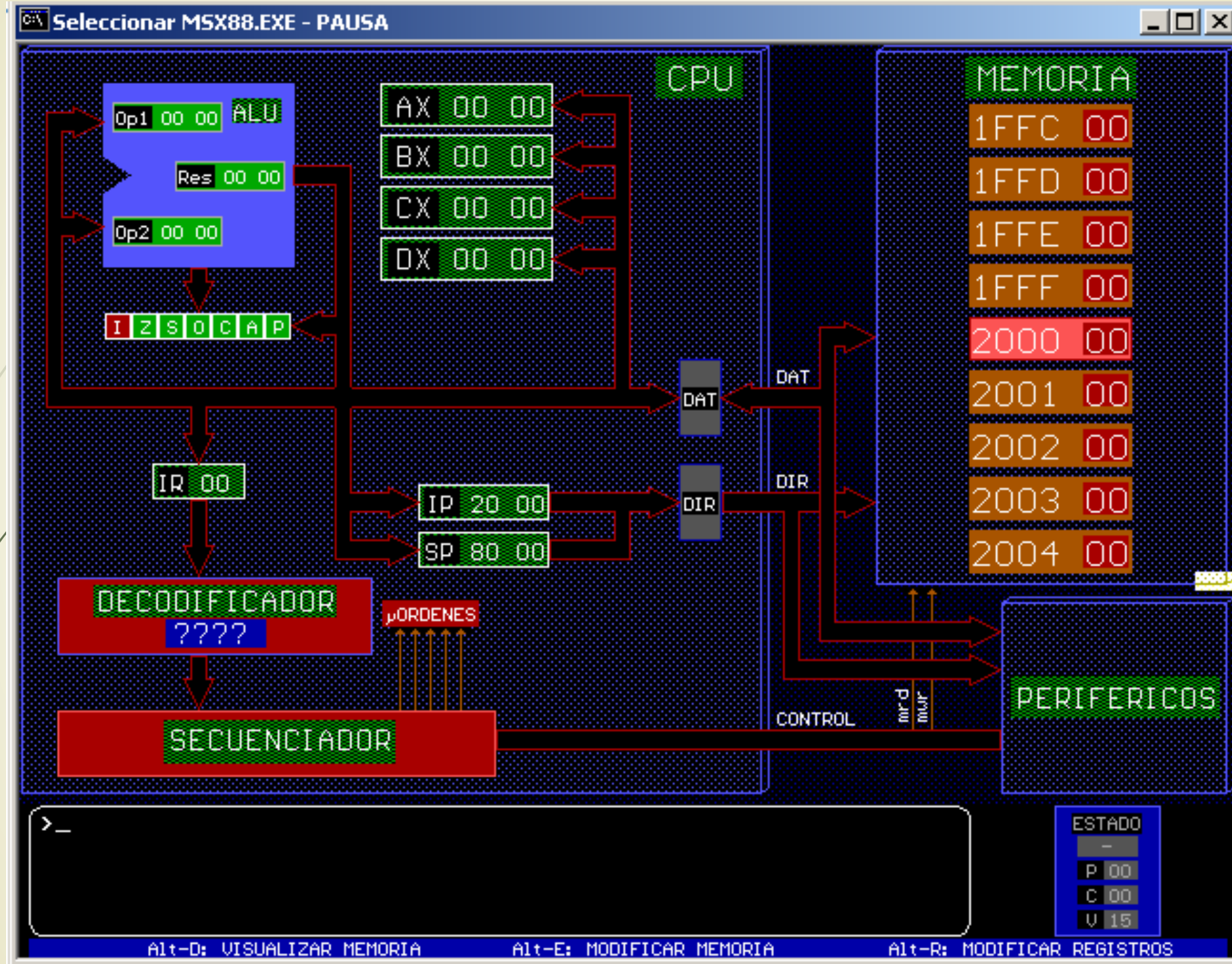


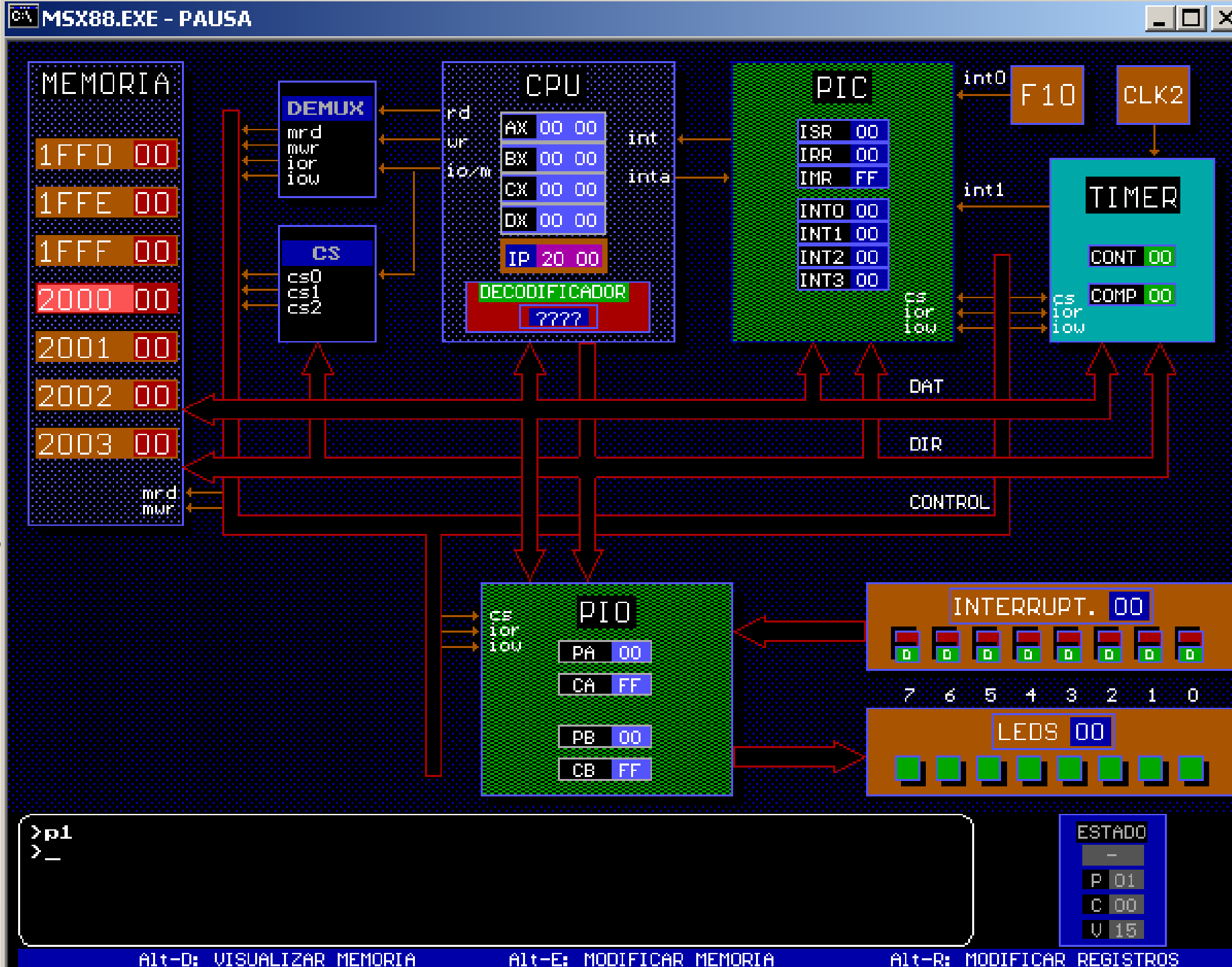
Arquitectura de Computadoras

Curso 2020 – Prof. Jorge Runco
Interrupciones - Programación

Simulador

- Veamos interrupciones (software y hardware) en el simulador.





Vectores de Interrupción (1)

- 00 → 255 - 00h → FFh hay 256 entradas en la tabla.
- Cada entrada en la tabla son 4 bytes, que tienen la dirección del servicio que atiende a la interrupción.
- Los 16 bits más altos están siempre en cero.
- Los restantes 16 bits tienen la dirección lógica/física. Las direcciones el simulador son de 16 bits. Ej. 0000XXXXh

Vectores de Interrupción (2)

- De las 256 posibles interrupciones : 4 están usadas por interrupciones de soft:
- Tipo 0 (INT 0): finaliza la ejecución de un programa.
- Tipo 3 (INT 3): puntos de parada en el programa.
- Tipo 6 (INT 6): espera un carácter del teclado y lo almacena en la dirección apuntada por BX.(Entrada estándar).

Vectores de Interrupción (3)

- Tipo 7 (INT 7): escribe en pantalla.
BX=dirección de comienzo del bloque de datos. AL=No de datos. (Salida estándar).
- El resto de las interrupciones (entradas en la tabla) están libres para ser utilizadas.

Periféricos internos

- PIO : puertos paralelos de e/s
- HAND-SHAKE :
- PIC : controlador de interrupciones
- TIMER : contador de eventos
- CDMA : controlador de acceso directo a memoria

Periféricos externos

- Barra de led's.
- Barra de interruptores (microswitchs).
- Impresora.

Controlador de interrupciones (PIC)

- ▶ Puede manejar hasta 8 interrupciones.
- ▶ INT0 → INT7, por hardware (sin espacios)
- ▶ Esquema de prioridades :
- ▶ 0 = más alta prioridad
- ▶ 7 = más baja prioridad

PIC: registros internos

- Todos son de 8 bits
- ISR: registro de interrupción en servicio. Indica cual es la interrupción que está siendo atendida. Ej. Si se atiende INT0 se pone en 1 el bit 0; si se atiende INT7 se pone en 1 bit 7.
- IRR: registro de petición de interrupción. El bit asociado se pone en 1 cuando hay un pedido de int y va a 0 cuando es atendida.

PIC: registros internos

- IMR: registro de máscara de interrupción. Permite enmascarar cada una de las entradas, bit 0 INT0, si se pone en 1 se enmascara INT0 (no se atiende).
- INT0...INT7: contiene el valor del vector. Este número multiplicado por 4 = entrada en la tabla
- EOI : se envía el comando de final de interrupción (end of interrupt) 20h = comando al final de la rutina de hard.

Acceso a los registros del PIC

Se acceden a partir de la dirección 20h.

EOI = 20h INT0 = 24h INT4 = 28h

IMR = 21h INT1 = 25h INT5 = 29h

IRR = 22h INT2 = 26h INT6 = 2Ah

ISR = 23h INT3 = 27h INT7 = 2Bh

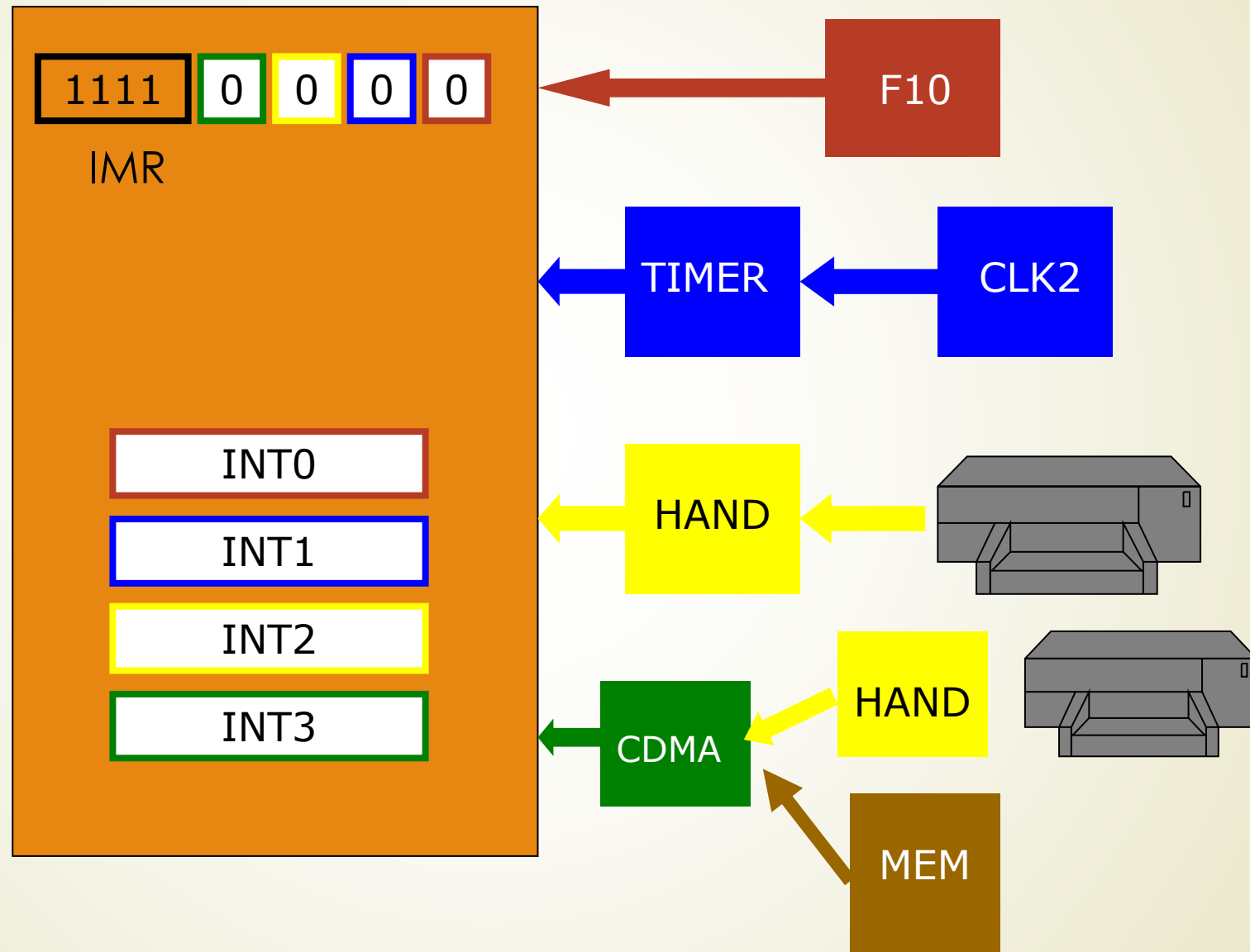
Interrupciones por hardware

- int0 : conectada a la tecla F10, que produce una interrupción al ser pulsada.
- int1 : se conecta a la línea OUT del TIMER, se produce una interrupción con cada pulso del mismo.
- int2 : se conecta a la línea INT de HAND
- int3 : se conecta al controlador de acceso directo a memoria.

Conexión del PIC con los periféricos externos

- Tecla F10
- Timer
- Hand : para controlar la impresora
- CDMA : controlador de acceso directo a memoria. Controla transferencias mem-mem ó mem-impresora.

PIC



Registros del PIC

IMR

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

INT0 e INT2
habilitadas, bit0 y
bit2 en cero

ISR


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

Está en servicio
INT2, pues bit2
vale 1

IRR

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

Hay un pedido de
INT0 pendiente
pues bit0=1



ORG 4 x Número de vector
dw Nombre

ORG 2000H

Deshabilitar la atención de interrupciones (CPU)

Habilitar la interrupción en el IMR (PIC)

Escribir el vector seleccionado en el reg (INT0, INT1...)

Inicialización dependiendo del periférico

Habilitar la atención de interrupciones (CPU)

Esperar la finalización del programa

ORG 3000H

Nombre: Rutina de interrupción

RET

Interrupción por software

```
ORG 1000H
MSJ  DB "ARQUITECTURA DE COMPUTADORAS"
      DB "FACULTAD DE INFORMATICA"
      DB "UNLP"
FIN  DB ?
```

```
ORG 2000H
Parámetros { MOV BX, OFFSET MSJ ; Puntero al comienzo del mensaje
              MOV AL, OFFSET FIN-OFFSET MSJ ; Cantidad de
; caracteres en el mensaje
INT 7
INT 0
END
```

Lectura de datos desde el teclado

MSJ
FIN
ORG 1000H
DB "INGRESE UN NUMERO :"
DB ?

NUM
ORG 1500H
DB ?

ORG 2000H
MOV BX, OFFSET MSJ ; Puntero a MSJ
MOV AL, OFFSET FIN-OFFSET MSJ ; Cantidad de caracteres
INT 7
MOV BX, OFFSET NUM ; Puntero a NUM
INT 6
MOV AL, 1; BX quedó cargado apuntando a NUM
INT 7
MOV CL, NUM
INT 0
END

PIC EQU 20H
 EOI EQU 20H
 N_F10 EQU 10

ORG 40
 IP_F10 DW RUT_F10

ORG 2000H
 CLI
 MOV AL, 0FEH
 OUT PIC+1, AL
 MOV AL, N_F10
 OUT PIC+4, AL
 MOV DX, 0
 STI
 LAZO: JMP LAZO

Interrupción con tecla F10

ORG 3000H
 RUT_F10: PUSH AX
 INC DX
 MOV AL, EOI
 OUT EOI, AL
 POP AX
 IRET

 END

ORG 40

IP_F10 DW RUT_F10

- ORG 40 : porque vamos a instalar la interrupción en el lugar 10 de la tabla de vectores. Como cada entrada ocupa 4 bytes, la dirección es 4×10 . Aquí va la dirección de la primera instrucción del servicio que atiende a la interrupción. Esta dirección tiene una etiqueta RUT_F10, pero vemos que es 3000h.

MOV AL, 0FEH
OUT PIC+1, AL

- Estas dos instrucciones cargan en el registro IMR el valor FEh, poniendo el bit 0 en 0 y los restantes bits en 1, enmascarando todas las interrupciones menos la INT0 que corresponde a la tecla F10.
- 0FEh= lleva un 0 al principio para indicar que el resto de las letras corresponden a un número y no al nombre de una variable



MOV AL, N_F10
OUT PIC+4, AL


- Estas dos instrucciones escriben en el registro INT0 del PIC, el valor de la posición en la tabla de vectores, en este registro se buscará dicha posición para la interrupción producida por F10. Recordemos que se accede al registro en la dirección 24h (PIC+4).

MOV DX, 0

- En DX vamos a contar cuantas veces fue presionada la tecla F10. Acá inicializamos la cuenta en 0.

STI y CLI

- ▶ Las instrucciones STI y CLI activan y desactivan, respectivamente, interrupciones. Cuando se activa la bandera I permite que por el terminal INTR del procesador, se produzca una interrupción.
- ▶ Cuando se desactiva el bit impide que se produzca la interrupción por el terminal INTR.
- ▶ INTR es un terminal del procesador.



MOV AL, EOI { OUT 20H,20H
OUT EOI, AL }

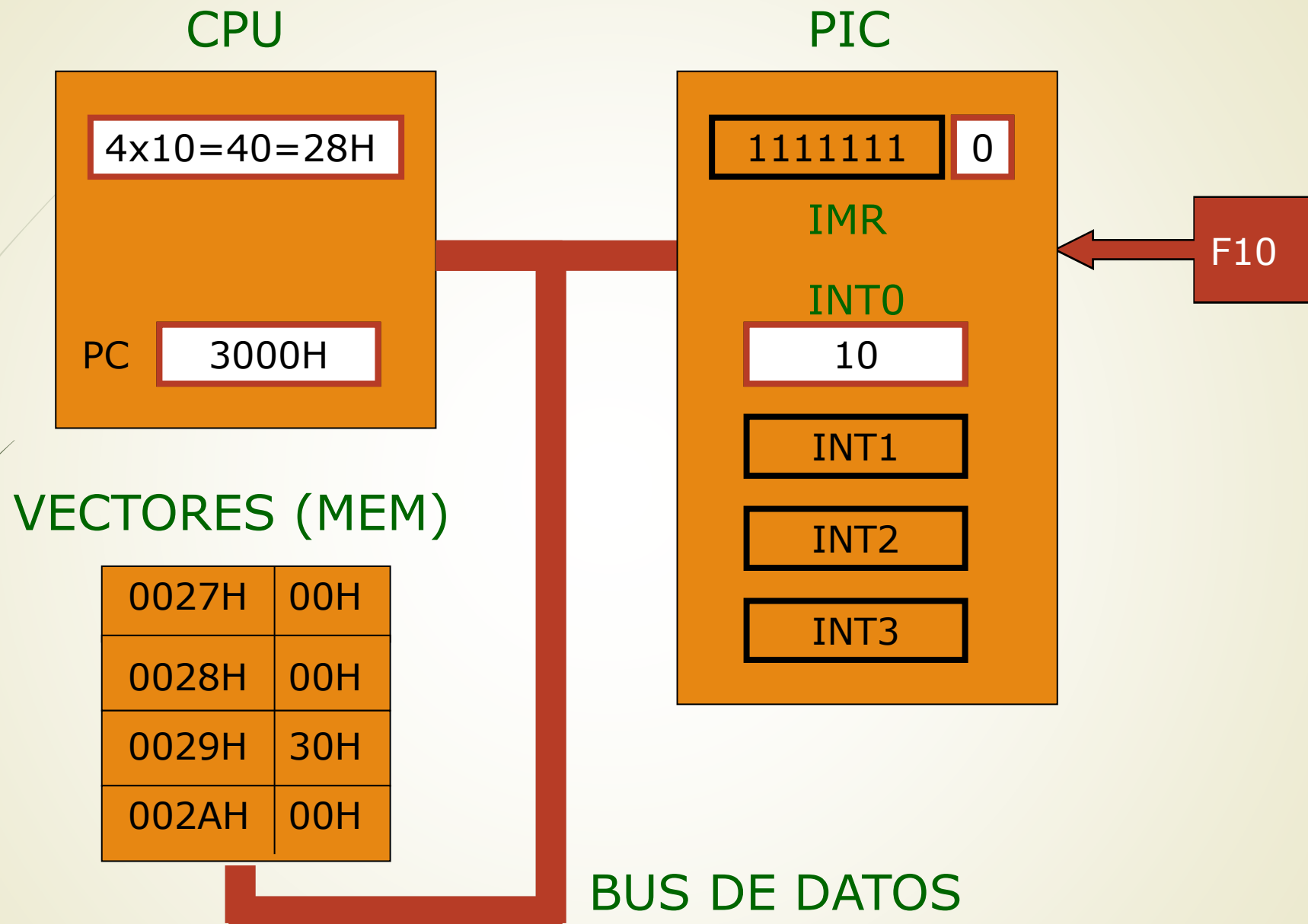
- La CPU debe indicarle al controlador cuando ha completado el servicio de cada **interrupción de hardware** . Por lo tanto al final de la rutina de servicio de la interrupción habrá que escribir en el registro de comandos, un número (comando) que indique el final de la interrupción.
- La dirección del registro coincide con el valor a escribir.

IRET

- La instrucción IRET es una instrucción especial para retorno y se utiliza para retornar de las interrupciones de software y hardware.
- La instrucción IRET es como una instrucción RET, porque recupera de la pila la dirección de retorno.
- Es diferente porque también recupera una copia del registro de banderas de la pila.
- La instrucción IRET extrae 6 bytes de la pila : 4 para la dirección de retorno y 2 para el registro de banderas.

TIMER

- Posee dos registros de 8 bits.
- COMP : registro de comparación que determina el módulo de la cuenta del timer.
- CONT : registro contador, muestra la cuenta de los pulsos de la señal aplicada a la entrada INT del periférico. Cuando coincide su valor con el reg COMP provoca una salida del TIMER.
- Direcciones de registros : 10h y 11h
- Frecuencia : 1 Hz



Funcionamiento del TIMER

