

Arquitectura de computadoras



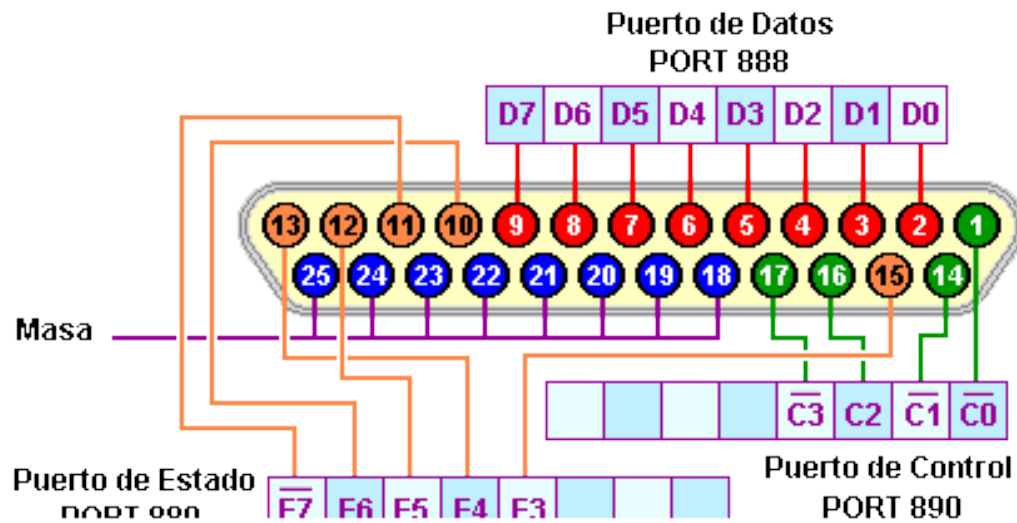
Explicación Práctica 3
Entrada / Salida

Entrada / Salida

- ▶ Gran variedad de periféricos con varios métodos de operación, pero todos **son más lentos que la CPU y la RAM**
- ▶ Necesidad de una **interface** entre el procesador, la memoria y los periféricos
- ▶ MSX88 utiliza lo que se llama **E/S aislada** → las direcciones de E/S , llamadas **puertos**, están separadas de la memoria
 - ▶ **Ventaja:** no existe memoria de E/S mapeada en memoria principal (no se desperdicia espacio)
 - ▶ **Desventaja:** para transferir datos se debe usar instrucciones especiales como **IN** y **OUT**.

Puerto paralelo (Centronics)

- ▶ Registro de **control** (bidireccional): 4 bits.
- ▶ Registro de **estado** (entrada): 5 bits.
- ▶ Registro de **datos** (bidireccional): 8 bits.



PIO (puerto paralelo E/S)

- ▶ Son 2 puertos de 8 bits: A y B.
- ▶ Se puede programar cada bit por separado como entrada ó salida.

Posee 4 registros 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	
Datos {	30H PA
	31H PB
Configuración {	32H CA
	33H CB

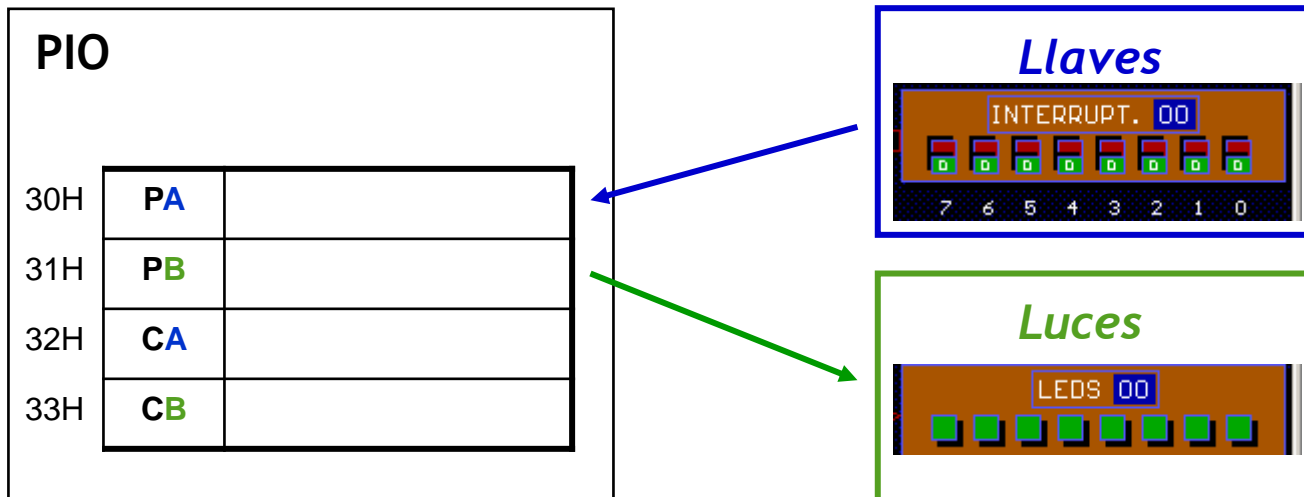
*Un bit en **0** en CA/CB selecciona como **salida** a la línea correspondiente en PA/PB*

*Un bit en **1** en CA/CB selecciona como **entrada** a la línea correspondiente en PA/PB*

PIO – Luces y Llaves

Encendido/apagado de las **luces** (periférico de salida) mediante la barra de microconmutadores o **llaves** (periférico de entrada), ambos comunicados con el microprocesador a través de los puertos paralelos de la PIO.

Implementar un programa que configure la PIO para leer el estado de los microconmutadores y escribirlo en la barra de luces. El programa se debe ejecutar bajo la configuración **P1 C0** del simulador. Las llaves se manejan con las teclas 0-7.



PIO – Luces y llaves

PA EQU 30H
PB EQU 31H
CA EQU 32H
CB EQU 33H

ORG 2000H

MOV AL, 0FFH

OUT CA, AL

MOV AL, 0

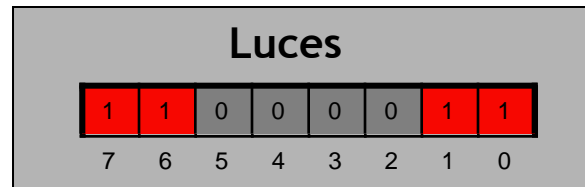
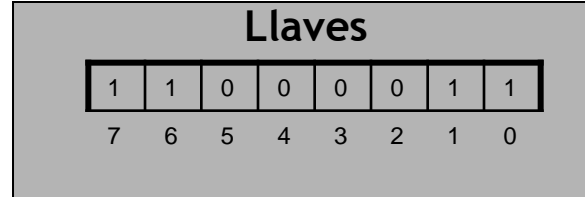
OUT CB, AL

POLL: IN AL, PA

OUT PB, AL

JMP POLL

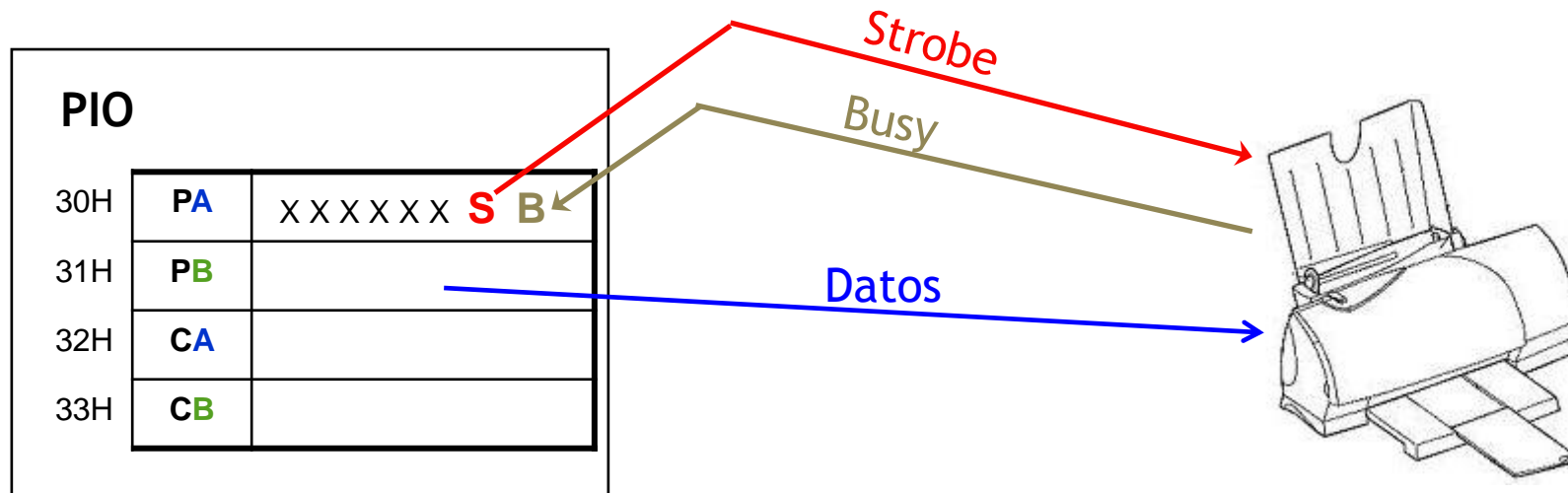
END



PIO		
30H	PA	
31H	PB	
32H	CA	
33H	CB	

PIO – Impresora

PA_0 : línea **BUSY** (*entrada*)
 PA_1 : línea **STROBE** (*salida*)
 $PB_0...PB_7$: líneas de **DATOS** (*salidas*)



PIO – Impresora

Uso de la **impresora** a través de la **PIO**

Escribir un programa que envíe datos a la impresora a través de la PIO. La PIO debe cumplir las funciones de temporización que requiere la impresora para la comunicación.

Ejecutar en configuración **P1 C1** del simulador y presionar **F5** para mostrar la salida en papel. El papel se puede blanquear ingresando el comando **BI**.

PIO – Impresora

```
PIO EQU 30H
ORG 1000H
MSJ DB "ARQUITECTURA DE "
      DB "COMPUTADORAS"
FIN  DB ?
```

ORG 2000H

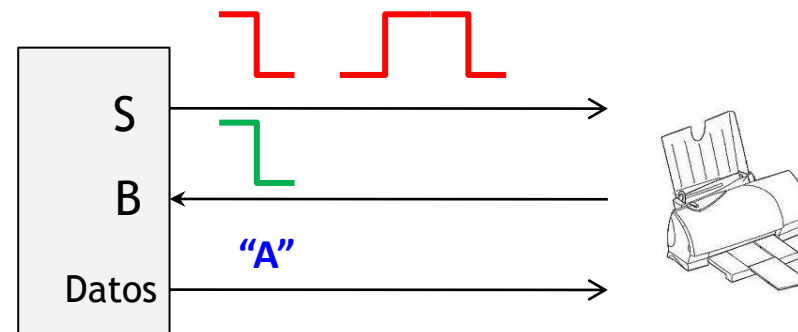
```
MOV AL, 0FDH
OUT PIO+2, AL
MOV AL, 0
OUT PIO+3, AL
IN AL, PIO
AND AL, 0FDH
OUT PIO, AL
MOV BX, OFFSET MSJ
MOV CL, OFFSET FIN-OFFSET MSJ
```

CA = 1111 1101 → B= Entrada S=Salida
CB = 0000 0000 → PB=Salida
Fuerzo Strobe a 0
Inicializo BX y CL

```
POLL: IN AL, PIO
      AND AL, 1
      JNZ POLL
      MOV AL, [BX]
      OUT PIO+1, AL
      IN AL, PIO
      OR AL, 02H
      OUT PIO, AL
      IN AL, PIO
      AND AL, 0FDH
      OUT PIO, AL
      INC BX
      DEC CL
      JNZ POLL
      INT 0
```

Leo estado de la línea **BUSY** y no avanzo hasta que no este libre (0)
Coloco el carácter a imprimir en el puerto de datos (PB)
Fuerzo Strobe a 1
Fuerzo Strobe a 0
BX apunto al siguiente caracter

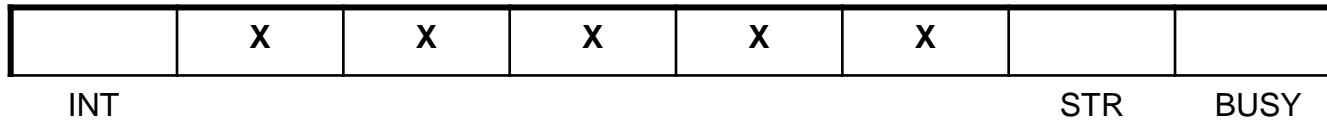
END



Handshake – Impresora

► **DATO (40H):** Usado para poner el **byte a imprimir**

► **ESTADO (41H):**



BIT 0 en 0: línea **BUSY** desactivada
en 1: línea **BUSY** activada

BIT 1 en 0: línea **STROBE** desactivada
en 1: línea **STROBE** activada (generación del pulso **automática**)

BIT 2..6 → **No se usan**

BIT 7 en 0: modo **POLLING** (no se activa la línea INT)
en 1: modo **INT** (se activa la línea INT cuando BUSY = 0)

HS – Impresora – Polling

Uso de la impresora a través del dispositivo de **hand-shaking** por **consulta de estado (POLLING)**.

Escribir un programa que envíe datos a la impresora a través del HAND-SHAKE. La comunicación se debe establecer por consulta de estado.

Ejecutar en configuración **P1 C2**.

HS – Impresora – Polling

HAND EQU 40H

ORG 1000H

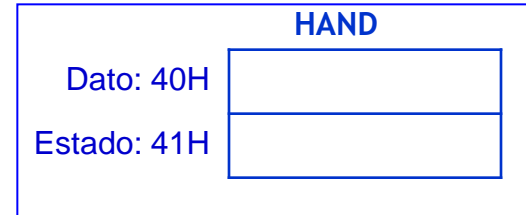
```
MSJ DB      "FACULTAD DE "  
      DB      "INFORMATICA"  
FIN   D B    ?
```

ORG 2000H

```
IN AL, HAND+1  
AND AL, 7FH  
OUT HAND+1, AL  
MOV BX, OFFSET MSJ  
MOV CL, OFFSET FIN-OFFSET MSJ
```

```
POLL: IN AL, HAND+1  
      AND AL, 1  
      JNZ POLL  
      MOV AL, [BX]  
      OUT HAND, AL  
      INC BX  
      DEC CL  
      JNZ POLL  
      INT 0
```

END



} Se configura el HAND para polling
(7FH = 0111 1111)

} Leo estado de la linea **BUSY** y no
avanzo hasta que no este libre (0)

} Envío el carácter a imprimir → **STROBE AUTOMATICO**

HS – Impresora – INT

Uso de la impresora a través del dispositivo de **hand-shaking** por **interrupción**.

Escribir un programa que envíe datos a la impresora a través del HAND-SHAKE. La comunicación se debe establecer por interrupciones emitidas desde el HAND-SHAKE cada vez que la impresora se desocupa.

Ejecutar en configuración **P1 C2**.

HS – Impresora – INT

```
PIC      EQU 20H
HAND     EQU 40H
N_HND    EQU 10
```

```
ORG 40
IP_HND   DW   RUT_HND
```

```
ORG 1000H
MSJ DB   "UNIVERSIDAD "
      DB   "NACIONAL DE LA PLATA"
FIN  DB   ?
```

```
ORG 3000H
RUT_HND: PUSH AX
          MOV AL, [BX] } Envía el carácter a
          OUT HAND, AL } imprimir
          INC BX
          DEC CL
          MOV AL, 20H } Fin de atención a
          OUT PIC, AL  } la interrupción
          POP AX
          IRET
```

ORG 2000H

```
MOV BX, OFFSET MSJ
MOV CL, OFFSET FIN-OFFSET MSJ
CLI
Configuración del } MOV AL, 0FBH } IMR = 11111011 → INT2 (HAND)
PIC para atender } OUT PIC+1, AL
al HAND           } MOV AL, N_HND } Asigno a INT2 el tipo 10
                  } OUT PIC+6, AL
                  } MOV AL, 80H } Configuro el HAND para
                  } OUT HAND+1, AL } interrupción
                  } STI
LAZO:             } CMP CL, 0 } Mientras tenga caracteres para
                  } JNZ LAZO } imprimir se queda en lazo
                  } IN AL, HAND+1
                  } AND AL, 7FH } Deshabilita las
                  } OUT HAND+1, AL } interrupciones
                  } INT 0
                  } END
```