

Entrada/Salida

Organización del Computador I

Departamento de Computación - FCEyN
UBA

1^{er} Cuatrimestre 2018

Hasta ahora vimos...

- Representación de números enteros
- Circuitos combinatorios y secuenciales
- Arquitecturas
- Algo de programación en assembler de ORGA1

¿De qué sirve una computadora sin dispositivos de E/S?

Dispositivos de E/S

- **Entrada:** Teclado, Mouse, Micrófono, etc.
- **Salida:** Monitor, Parlantes, Impresora, etc.
- **Entrada/Salida:** Disco rígido, Módem, Placa de Red, etc.

¿Y cómo intercambio información con un dispositivo de E/S?

¿Qué es un dispositivo de E/S para la CPU?

Cada dispositivo de E/S tiene sus propios registros, donde la CPU puede leer o escribir datos.

Tipos de registro:

- Lectura.
- Escritura.
- Lectura/Escritura.

Métodos de acceso a los registros

¿Con qué instrucciones accedo a estos registros?

Hay dos formas de referirse a los registros de un dispositivo de E/S:

- E/S independiente (instrucciones especiales: IN y OUT).
Espacio de direcciones independiente
- E/S mapeado a direcciones de memoria.
Direcciones de memoria principal reservadas para E/S

ORGA1 reserva las direcciones de memoria 0xFFFF0 a 0xFFFF para E/S.

Esquemas de E/S

- **E/S por encuesta (Polling) o Programada**
- **E/S por interrupciones**
- **E/S por acceso directo a memoria (DMA)**

¿Como funciona cada uno?

Ejercicio 1

Luego de muchos años de discusiones, la FIFA accedió a incorporar tecnología a los fallos de los árbitros en los partidos de fútbol.

Para lograr esto la FIFA, con su gran presupuesto, decidió comprar una máquina ORGA1 junto con dispositivo de E/S.

Este dispositivo cuenta con un registro de estado mapeado a la dirección de E/S 0xFFFF0 de sólo lectura, donde se refleja el porcentaje de la pelota ingresado al arco, que varía entre 0 y 100, y comienza en el valor 0x0000. Se decide que cuando el porcentaje llega a 100 se considera gol. Luego, debe volver a cero antes de que se considere que la pelota está afuera y se pueda empezar a evaluar si hubo un segundo gol.

Escribir una rutina en ensamblador para satisfacer el sistema pedido, guardando la cantidad de goles en R0. Esta rutina se invoca al iniciar el partido.

Solucion 1)

```
comienzaElPartido: MOV R0, 0x0000
noEntro:           CMP [0xFFFF0], 0x0064
                   JNE noEntro
                   ADD R0, 0x0001
sigueAdentro:     CMP [0xFFFF0], 0x0000
                   JG  sigueAdentro
                   JMP noEntro
```


El procesador ORGA1I

El procesador ORGA1I es un procesador ORGA1 que ha sido extendido con la capacidad de atender una única interrupción (enmascarable) de un dispositivo de E/S.

- **Nuevas señales:**

- Entrada: INTR (Interrupción)
- Salida: INTA (Interrupción reconocida)

- **Nuevo flag:** I, que indica si el procesador puede ser interrumpido o no

- **Nuevo registro:** PSW, en donde se almacenan los flags

- **Nuevas instrucciones:**

- CLI y STI
- IRET
- PUSH Ri y POP Ri

- **Nueva dirección reservada:** 0x0000, donde se indica la dirección de la rutina de atención de la interrupción

¿Qué pasa cuando interrumpen al CPU?

En el caso de un procesador ORGA1I, si el dispositivo de E/S activa la señal de interrupción y el flag I vale 1, termina de ejecutar la instrucción en curso y realiza **atómicamente** los siguientes pasos:

- Coloca $[SP] = PSW$ y decrementa SP
- Coloca $[SP] = PC$ y decrementa SP
- Coloca $I=0$ para evitar que el procesador vuelva a interrumpirse
- Coloca $PC = [0x0000]$
- Activa la señal INTA para indicarle al dispositivo que atenderá su pedido
- Se ejecuta la rutina de atención de interrupción (RAI)
- La RAI termina con la ejecución de IRET

Ejercicio 2

Una computadora ORGA1I se está utilizando para monitorear el estado de una montaña rusa. Esta computadora se encarga de verificar ciertos parámetros de la montaña rusa y actuar de acuerdo a su estado. Para ello, cuenta con: dos dispositivos de E/S que actúan como sensores, uno que actúa como alarma y otro que efectúa una parada de emergencia. Cada sensor posee un registro de E/S de sólo lectura que reportan la siguiente información:

- Velocidad: Mide la velocidad del carrito de la montaña rusa (VEL_STATUS)
- Frenos: Mide el estado de los frenos (BR_STATUS)

Ejercicio 2 (continuación)

El dispositivo de alarma posee un registro de E/S de lectura/escritura (ALARMA) que permite activar dos alarmas distintas:

- El bit menos significativo representa la alarma de velocidad. Esta le indica al operador que el carrito está yendo muy rápido.
- El segundo bit representa el estado de los frenos comunes. Este indica que hay un problema con estos frenos y que se van a activar los frenos de emergencia.

Si un bit está en 1, la alarma correspondiente se enciende. Las alarmas se apagan de manera externa al sistema.

Para determinar si se exceden los parámetros de velocidad o de estado de frenos se tienen las etiquetas MAX_SPEED y MIN_BRAKES (constantes de 16 bits).

El dispositivo de frenos de emergencia posee un registro de E/S de escritura (FRENOS_EM) que los activa en caso que se detecte un problema con los frenos comunes, poniendo todo los bits en 1.

Ejercicio 2

- a) Mapear los registros de E/S a direcciones de E/S de ORGA1.
- b) Realizar el código para sensor y activar las alarmas correspondientes.
- c) Suponiendo que el ciclo de instrucción de cada instrucción del programa tarda 3 ms y los valores máximos nunca se alcanzan ¿Cuál es la frecuencia (en Hz) de muestreo (lectura) de los sensores? ¿Y si todos los sensores sobrepasan los máximos?

Solución 2.a)

Mapeo de registros.

- VEL_STATUS \mapsto 0xFFFF0
- BR_STATUS \mapsto 0xFFFF1
- ALARMA \mapsto 0xFFFF2
- FRENOS_EM \mapsto 0xFFFF3

Solución 2.b)

```
sensaVel:    CMP [0xFFF0], MAX_SPEED    ;alcanzo velocidad maxima?
             JL  sensaFrenos
             OR  [0xFFF2], 0x0001
sensaFrenos: CMP [0xFFF1], MIN_BRAKES   ;problema con los frenos?
             JG  sensaVel
             MOV [0xFFF3], 0xFFFF
             OR  [0xFFF2], 0x0002
             JMP sensaVel
```

Solución 2.c)

Si no se alcanzan los valores máximos, cada iteración ejecuta 4 instrucciones. Por lo tanto, podemos concluir que cada iteración tarda $4 * 3$ ms. Como podemos realizar una única lectura por iteración, se lee cada señal cada $4 * 3$ ms. En conclusión (y teniendo en cuenta que $1000\text{ms} = 1\text{s}$), la frecuencia de muestreo es $\frac{1000}{4*3} = 83,33$ Hz.

En cambio, si se sobrepasan los valores máximos, cada iteración ejecuta 8 instrucciones, tardando $8 * 3$ ms. Cada señal se lee cada $8 * 3$ ms, dando una frecuencia de $\frac{1000}{8*3} = 41,66$ Hz.

Ejercicio 3

El dueño de la montaña rusa invirtió algo de dinero y compró un nuevo sensor para los frenos. Este nuevo sensor solicita una interrupción si se detecta un inconveniente con los frenos comunes.

- a) Modificar el programa presentado para aprovechar esta característica de modo que la frecuencia de muestreo sea mayor.
- b) Escribir la rutina de atención de la interrupción del sensor de frenos.
- c) Calcular la nueva frecuencia de muestreo para el sensor de velocidad.
- d) ¿Cómo quedaría la memoria si se pide además, que la rutina principal se cargue a partir de la posición 0xAAAA y la rutina de atención de interrupciones 20 posiciones antes?

Solución 3)

Modificación

```
sensaVel:  CMP [0xFFF0], MAX_SPEED ;alcanzó velocidad máxima?  
           JL  sensaVel  
           OR  [0xFFF2], 0x0001 ; prendo la alarma de velocidad  
           JMP sensaVel
```

Rutina de atención de la interrupción

```
rut_at_int: MOV [0xFFF3], 0xFFFF ; activo los frenos de emergencia  
           OR  [0xFFF2], 0x0002 ; prendo la alarma de frenos  
           IRET
```

Muestreo

Si no se alcanza el valor máximo, cada iteración ejecuta 2 instrucciones. Dando una frecuencia de muestreo de $\frac{1000}{2*3} = 166,66$ Hz. Si, en cambio, se sobrepasa el valor máximo, cada iteración ejecuta 4 instrucciones, dando una frecuencia de $\frac{1000}{4*3} = 83,33$ Hz.

Solución 3.d)

- A partir de la posición de memoria 0xAAAA se carga la rutina principal.
- A partir de la posición de memoria 0xAA96 se carga la rutina de atención de interrupciones.
- En la posición 0x0000 cargamos 0xAA96, que es la posición de memoria donde comienza la RAI.

¿Preguntas?

