

1 [3 puntos] A una CPU de 32 bits de ancho de palabra y 1.000 MIPS de capacidad de ejecución se desea conectar un periférico del que se sabe que su velocidad de transferencia es 40 MB/s (40×10^6 bytes/s) y su buffer es de 4 registros de datos de 32 bits.

Suponga que opera mediante interrupciones. La secuencia de reconocimiento de interrupción, SRI, tiene una duración de 10 ns y en su rutina de interrupción se ejecutan 50 instrucciones. Calcule:

- a) El tiempo entre dos solicitudes de interrupción consecutivas, período de las interrupciones, T_{Int}
- b) La duración máxima posible de su rutina de interrupción, $t_{Rut-Int}$ o t_{RTI}
- c) La frecuencia de las solicitudes de interrupción del periférico, $Frec_{Int}$
- d) El consumo de CPU equivalente debido a las interrupciones expresado en MIPS. En función de este valor calcule el número máximo de unidades de este tipo de periférico que podrían operar simultáneamente.

Suponga ahora que se sustituye el módulo de E/S por otro cuyo buffer es, en lugar de 4, de 16 registros de 32 bits. Indique cómo afectaría este cambio a:

- e) La velocidad de transferencia del periférico.
- f) La frecuencia de las solicitudes de interrupción, $Frec_{Int}$
- g) El consumo de CPU.

SOLUCIÓN

- a) El tiempo entre interrupciones se determina considerando que hay una solicitud por cada buffer de 4 palabras de 8 bytes:

$$T_{Int} = \frac{(4 \times 8B) / Int}{40 \cdot 10^6 B/s} = 400 \text{ ns} / Int$$

- b) De este tiempo, la Secuencia de Reconocimiento de Interrupción consume 10 ns, por lo que quedarían un máximo teórico de 390 ns para la rutina de tratamiento de interrupción de un único periférico, $t_{Rut-Int-MAX}$ o $t_{RTI-MAX}$.

- c) La frecuencia de interrupciones es la inversa del tiempo entre interrupciones:

$$Frec_{Int} = 1Int / 400 \text{ ns} = 2,5 \times 10^6 Int/s$$

- d) A partir de la frecuencia de solicitud de interrupción antes calculada y del consumo de CPU de cada una, t_{Int} , podremos calcular el consumo máximo de cada unidad del periférico. De este consumo por unidad se obtendrá cómo se reparte el total de MIPS de la CPU.

Cada interrupción consume:

$$t_{Int} = t_{SRI} + t_{Rut-Int} = 10 \text{ ns} + 50 \text{ ns} = 60 \text{ ns} / Int$$

y por lo tanto, el consumo de CPU por cada unidad de periférico será:

$$Consumo_{CPU/unidad} = 2,5 \times 10^6 Int/s \times 60 \text{ ns} / Int \times 1 \text{ inst/ns} = 150 \text{ MIPS/unidad}$$

por lo que los 1.000 MIPS de la CPU podrían permitir el funcionamiento simultáneo de:

$$\frac{1.000 \text{ MIPS}}{150 \text{ MIPS/unidad}} = 6,6 \text{ unidades} \rightarrow 6 \text{ unidades}$$

Cuestiones supuesto un nuevo buffer de 16 palabras:

- e) La velocidad de transferencia del periférico es una característica propia suya e independiente del tamaño del buffer de su módulo de E/S.
- f) Al cuadruplicarse el tamaño del buffer, las solicitudes de interrupción se distanciarán a cuatro veces el período original, es decir, lo que es lo mismo, su frecuencia será 1/4 de la calculada anteriormente.
- g) De acuerdo con lo anterior, el consumo de CPU también se verá reducido a la cuarta parte, aunque sólo en el supuesto de que la rutina de interrupción siga consumiendo 50 ns. En la vida real la nueva rutina suele tener una duración mayor para gestionar el nuevo tamaño del buffer.

2 [7 puntos] A una CPU igual a la del ejercicio anterior se conectan ahora dos periféricos, P1 y P2, que tienen las siguientes características:

- P1: *Red*
 - Velocidad de transferencia: 100 Mbits/s (100×10^6 bits/s)
 - Tamaño de los bloques: 1.024 B
 - Buffer: un registro de datos de 32 bits
- P2: *Disco duro*
 - Velocidad de transferencia: 20 MB/s (20×10^6 bytes/s)
 - Tamaño de los bloques: 4.096 B
 - Buffer: 128 registros de datos de 32 bits
 - Tiempo medio de acceso: 5 ms

Las rutinas de programación de las operaciones de E/S de los dos periféricos ejecutan 50 instrucciones y en la rutina de finalización se ejecutan, también en ambos casos, 100 instrucciones.

El periférico P1 opera mediante interrupciones y P2 por DMA. En la rutina de tratamiento de interrupción de P1 se ejecutan 20 instrucciones. Se recuerda además que en este procesador la secuencia de reconocimiento de interrupción, SRI, dura 10 ns.

En el funcionamiento mediante DMA la concesión y devolución de los buses consume 2 ns en total y la transferencia de una palabra 40 ns.

- a) Calcule la duración de una operación completa de E/S para P1 y para P2.
- b) Calcule el tiempo total que se consume de CPU debido a una operación de E/S de P1 y de P2.
- c) Calcule el tiempo medio de CPU que se consume por cada byte transferido durante una operación de E/S, otra vez para ambos periféricos, P1 y P2.
- d) Suponga que se quiere transferir un archivo de 32 KB almacenado en el disco a través de la red.
 - d.1) Calcule cuántas operaciones de E/S de P1 y de P2 serían necesarias para transferir el archivo.
 - d.2) Indique la duración total de esta operación de transferencia del archivo. Haga las suposiciones que estime necesarias y escríbalas en su solución.
 - d.3) Indique el consumo total del CPU durante la operación de transferencia del archivo, y de nuevo escriba las suposiciones que haya creído necesarias.

SOLUCIÓN

a) La duración total de una operación será la suma de los tiempos de inicio o programación más el tiempo de acceso –nulo en el caso de la red–, más el de transferencia y el correspondiente más lo que tarde en ejecutarse el código para la finalizar el mandato. A estos tiempos pueden añadirse el correspondiente al último DMA o la última interrupción, para P2 y P1, respectivamente, y siempre bajo los supuestos expuestos en clase. Nótese que en este caso los accesos por DMA tienen una duración mayor que la que aparece en otros ejercicios, al tratarse de un buffer de 128 palabras y de un tiempo de Mp de 40 ns, un total de 5.122 ns. Aun así se trata de un tiempo poco significativo frente a la duración total de la operación.

De los 1.000 MIPS, cada instrucción durará 1 ns/inst, por lo que la SRI tarda el equivalente a la ejecución de 10 instrucciones.

P1:

$$\begin{aligned}
 t_{prog} &= 50 \text{ inst} \times 1 \text{ ns/inst} = 50 \text{ ns} \\
 t_{transf} &= \frac{1.024 \text{ bytes}}{100 \times 10^6 \text{ bits/s} \times 1 \text{ byte/8 bits}} = 81,92 \times 10^3 \text{ ns} \\
 t_{Int} &= t_{SRI} + t_{RTI} = (10 \text{ inst} + 20 \text{ inst}) \times 1 \text{ ns/inst} = 30 \text{ ns} \\
 t_{Fin} &= 100 \text{ inst} \times 1 \text{ ns/inst} = 100 \text{ ns} \\
 t_{operacion_P1} &= t_{prog} + t_{transf} + t_{Int} + t_{Fin} = \\
 &= 50 \text{ ns} + 81,92 \times 10^3 \text{ ns} + 30 \text{ ns} + 100 \text{ ns} = 82,100 \times 10^3 \text{ ns}
 \end{aligned}$$

P2:

$$\begin{aligned}
 t_{Prog} &= 50 \text{ inst} \times 1 \text{ ns/inst} = 50 \text{ ns} \\
 t_{acc} &= 5 \times 10^6 \text{ ns} \\
 t_{transf} &= \frac{4.096 \text{ bytes}}{(20 \times 10^6 \text{ byte/s})} = 204,8 \times 10^3 \text{ ns} \\
 t_{DMA} &= (2 \text{ ns} + 128 \times 40 \text{ ns}) = 5.122 \text{ ns} \\
 t_{Fin} &= 100 \text{ inst} \times 1 \text{ ns/inst} = 100 \text{ ns} \\
 t_{operacion_P2} &= t_{prog} + t_{acc} + t_{transf} + t_{DMA} + t_{Fin} = \\
 &= 50 \text{ ns} + 5 \times 10^6 \text{ ns} + 204 \times 10^3 \text{ ns} + 5.122 \text{ ns} + 100 \text{ ns} = 5,210072 \times 10^6 \text{ ns}
 \end{aligned}$$

b) En el caso de las interrupciones, cada una de ellas consume el correspondiente a la RTI más el equivalente en tiempo del SRI. En el caso de P2, aparte de las instrucciones que se ejecutan para programar o iniciar la operación, se supone que pierde de ejecutar el tiempo de los DMAs, siempre en los supuestos presentados en clase.

P1:

$$\begin{aligned}
 t_{Prog} &= 50 \text{ inst} \times 1 \text{ ns/inst} = 50 \text{ ns} \\
 t_{Int} &= t_{SRI} + t_{RTI} = (10 \text{ inst} + 20 \text{ inst}) \times 1 \text{ ns/inst} = 30 \text{ ns} \\
 \#Int &= 1.024 \text{ bytes} \times 1 \text{ Int/4 bytes} = 256 \text{ Int} \\
 t_{Fin} &= 100 \text{ inst} \times 1 \text{ ns/inst} = 100 \text{ ns} \\
 t_{CPU_P1} &= t_{Prog} + \#Int \times t_{Int} + t_{Fin} = \\
 &= 50 \text{ ns} + 256 \text{ Int} \times 30 \text{ ns/Int} + 100 \text{ ns} = 7.830 \text{ ns}
 \end{aligned}$$

P2:

$$\begin{aligned}
 t_{Prog} &= 50 \text{ inst} \times 1 \text{ ns/inst} = 50 \text{ ns} \\
 t_{Int} &= t_{SRI} + t_{RTI} = (6 \text{ inst} + 150 \text{ inst}) \times 1 \text{ ns/inst} = 156 \text{ ns} \\
 \#DMA &= 4.096 \text{ bytes} \times 1 \text{ DMA}/(128 \times 4 \text{ bytes}) = 8 \text{ DMA} \\
 t_{Fin} &= 100 \text{ inst} \times 1 \text{ ns/inst} = 100 \text{ ns} \\
 t_{CPU_P2} &= t_{Prog} + \#DMA \times t_{DMA} + t_{Fin} = \\
 &= 50 \text{ ns} + 8 \text{ DMA} \times 5.122 \text{ ns/DMA} + 100 \text{ ns} = 41.126 \text{ ns}
 \end{aligned}$$

c) El consumo medio por byte se calcula, simplemente, dividiendo el tiempo de CPU para cada operación por el tamaño del bloque correspondiente, 7,65 ns/byte y 10,05 ns/byte, respectivamente.

d) Operación de transferencias de 32 KB desde P1 a P2. **d.1)** Se necesitarán 8 operaciones (32 KB a 4 KB por operación) de P2 y 32 operaciones de P1 (1 KB por cada operación) para realizar la transferencia que se solicita .

Supuesto un almacenamiento temporal en memoria suficiente, es fácil comprobar que es viable el funcionamiento simultáneo de ambos periféricos, ésta es la opción que emplearemos, aunque consideramos igualmente válida la opción del funcionamiento secuencial. En nuestro supuesto, cada vez que se dispone de un bloque de 4 KB leído desde el disco se puede enviar a través de la red, a la vez que el disco sigue funcionando en la siguiente operación. Por tanto, el tiempo total será de 8 operaciones de P2 más el de las 4 últimas de la red. Bastará con utilizar los valores calculados en los apartados anteriores para los dos subapartados siguientes.

d.2) La duración total según el supuesto anterior:

$$\begin{aligned}
 t_{total_transf32KB P2 \rightarrow P1} &= 8 \times t_{operacion_P2} + 4 \times t_{operacion_P1} \\
 &= 8 \times 5.210.072 \text{ ns} + 4 \times 82,100 \times 10^3 \text{ ns}
 \end{aligned}$$

d.3) En el caso del consumo, tanto por interrupciones como por DMA, habrá que considerar el de las 8 operaciones de P2 y el de las 32 de P1:

$$\begin{aligned}t_{CPU32KB P2 \rightarrow P1} &= 8 \times t_{CPU_P2} + 32 \times t_{CPU_P1} \\ &= 8 \times 41.126 \text{ ns} + 32 \times 7.830 \text{ ns}\end{aligned}$$