







1. Un procesador desea configurar uno de los canales del controlador DMA para copiar un conjunto de datos de la memoria a un dispositivo de salida. ¿Cuál es el papel del procesador en esta operación y qué información fundamental transfiere al DMA antes de que comience la copia?
 - a) El procesador se encarga de transferir el dispositivo de origen y el de destino, pero el DMAC infiere las direcciones y la cantidad total de datos a transferir.
 - b) El procesador transfiere el primer dato al DMAC y luego este asume la transferencia.
 - c) El procesador actúa como gestor del DMAC, escribiendo en sus registros de control la dirección de origen en memoria, la dirección o identificador del dispositivo de destino, y la cantidad total de datos a transferir.
 - d) El procesador ejecuta la transferencia completa y el DMAC solo gestiona la interrupción al finalizar.
2. La *productividad máxima* de un dispositivo de E/S, a menudo citada en información comercial, puede resultar engañosa para evaluar el rendimiento real de una operación de transferencia. ¿Cuál es la razón fundamental por la que la productividad máxima no es la métrica más representativa de una transacción?
 - a) La *productividad máxima* se calcula excluyendo el tiempo de latencia inicial y se basa en el mejor caso posible para el dispositivo, por lo que no tiene en cuenta el tiempo total que tarda la transacción completa.
 - b) La *productividad máxima* solo considera la transferencia de datos de salida, ignorando las operaciones de entrada.
 - c) La *productividad máxima* se calcula basándose en el peor escenario posible, mientras que la transaccional asume el mejor caso.
 - d) La *productividad máxima* se aplica solamente a transferencias realizadas por DMA, no a transferencias por programa.
3. ¿Qué par de dispositivos de E/S ejemplifica mejor el principio de que los requisitos de baja latencia y alta productividad no siempre se exigen conjuntamente en un sistema?
 - a) Un teclado para jugadores (baja productividad y baja latencia —respuesta casi instantánea—) y un disco duro (alta productividad requerida para la transferencia de archivos, pero mayor latencia admisible).
 - b) Un ratón y un teclado para jugadores, ya que ambos son de baja latencia —respuesta casi instantánea— y baja productividad.
 - c) Un disco duro y un sensor de temperatura, ya que ambos requieren alta productividad y baja latencia.
 - d) Un monitor TFT y una tarjeta de sonido, ya que ambos son exclusivamente dispositivos de salida.



4. Aunque el Acceso Directo a Memoria (DMA) es superior para la transferencia de grandes bloques de datos, la transferencia de datos por programa (*Programmed I/O*) sigue siendo una técnica adecuada para ciertos dispositivos. ¿Para qué tipo de dispositivos y por qué es ventajosa la transferencia por programa?
 - a) Dispositivos que tienen una productividad muy baja (p.e., teclado o ratón), ya que la sencillez de esta técnica consume muy poco tiempo del procesador para transferir los pocos bytes por segundo que estos generan.
 - b) Dispositivos que requieren transferencia de datos sin usar buffers de memoria intermedios, ya que de esta forma no los necesitan.
 - c) Dispositivos de alta productividad (como SSDs) porque la velocidad de E/S requiere que el procesador esté constantemente activo.
 - d) Dispositivos que requieren una latencia inicial muy alta, lo que hace más ventajoso utilizar la transferencia de datos por programa.
5. En un sistema operativo multitarea con muchos periféricos (p.e., un PC), la gestión de E/S mediante consulta de estado es inadecuada. ¿Qué desventaja es la más crítica en este contexto?
 - a) La consulta de estado no permite el uso de registros de estado, mientras que las interrupciones sí.
 - b) La consulta de estado siempre requiere transferencia por DMA, mientras que las interrupciones se realizan por programa.
 - c) La consulta de estado consume una cantidad de tiempo de CPU inútilmente alta al consultar repetidamente dispositivos que a menudo no requieren atención.
 - d) La consulta de estado tiene una latencia mínima constante, mientras que las interrupciones introducen una latencia variable.
6. Para almacenar un archivo de 4 MB en un determinado disco SSD, es necesario descomponer el archivo en 4 bloques de 1 MB cada uno, que se enviarán de forma secuencial al disco duro. La transferencia de cada bloque tarda $400\ \mu\text{s}$, y entre la finalización del envío de un bloque y el comienzo del siguiente es necesario esperar $50\ \mu\text{s}$. Además, la latencia inicial de una transacción de este tipo es $100\ \mu\text{s}$. ¿Cuál es la productividad de la transacción completa, expresada en Megabytes por segundo (MB/s)?
 - a) 4.062,16 MB/s.
 - b) 2.562,16 MB/s.
 - c) 2.262,16 MB/s.
 - d) 2.162,16 MB/s.
7. Una diferencia clave entre el retorno de una subrutina y el retorno de una Rutina de Tratamiento de Interrupción (RTI) en ARM es la recuperación automática del estado del procesador. ¿Qué mecanismo utiliza el procesador ARM para distinguir si el retorno («**mov** pc, lr» o «**pop** pc») proviene de una subrutina estándar o de una RTI?



- a) La instrucción de retorno de una RTI («**mov pc, lr**») es diferente a la de una subrutina.
 - b) El bit C del registro de estado se activa solo al salir de una RTI, indicando al procesador que debe restaurar los registros apilados.
 - c) La RTI debe leer un registro de estado especial (NVIC) para saber si debe restaurar el contexto.
 - d) El procesador almacena el valor especial EXC_RETURN en el registro LR antes de llamar a la RTI.
8. Para implementar la consulta de estado de un pulsador conectado al PIOB₂₇ en un ATSAM3X8E, es crucial leer el valor de la entrada real del pin después de pasar por la lógica de filtrado de rebotes. ¿Cuál de los siguientes registros del controlador del PIOB proporciona esta información?
- a) PIO_ISR (*Interrupt Status Register*).
 - b) PIO_SODR (*Set Output Data Register*).
 - c) PIO_PDSR (*Pin Data Status Register*).
 - d) PIO_ODSR (*Output Data Status Register*).
9. Un microcontrolador se utiliza para regular la temperatura de una caldera. El sistema recibe datos de un sensor de temperatura cada 50 ms y, basándose en esta lectura, ajusta un relé de potencia (actuador). ¿Cuál es la caracterización temporal y cualitativa más precisa para este sensor de temperatura?
- a) Comportamiento: Bidireccional; Interlocutor: Humano; Productividad: Alta; Latencia: Mínima.
 - b) Comportamiento: Entrada; Interlocutor: Medio; Productividad: Baja (debido a la frecuencia de muestreo de 50 ms); Latencia: Media (aceptable en un rango de decenas de ms).
 - c) Comportamiento: Salida; Interlocutor: Máquina; Productividad: Alta; Latencia: Baja.
 - d) Comportamiento: Entrada; Interlocutor: Medio; Productividad: Media (debido a la frecuencia de muestreo de 50 ms); Latencia: Alta (mayor a 1 segundo).
10. En un sistema ARM que utiliza E/S mapeada en memoria, ¿cuál es el mecanismo por el cual el procesador utiliza un registro de datos de un controlador de E/S para una operación de salida, y cómo se relaciona esto con el repertorio de instrucciones del procesador?
- a) El procesador ejecuta una instrucción especial de E/S (**out**) dirigida al mapa de direcciones aisladas, que transfiere el dato al registro de datos del controlador.
 - b) El controlador de E/S escribe en el registro de datos (**str**) y luego genera una interrupción para avisar al procesador que el dato está disponible.
 - c) El procesador ejecuta una instrucción de carga (**ldr**) que trae el dato desde memoria al registro de datos del dispositivo.



- d) El procesador ejecuta una instrucción de almacenamiento (`str`) dirigida a la dirección de memoria asignada al registro de datos del controlador.