

Principios, implementación y rendimiento

1. La historia de la computación se suele estudiar dividida en 4 ó 5 generaciones. Identifique los hitos *hardware* y *software* de cada una.
2. ¿Qué es un intérprete *software*? ¿Qué es un compilador? ¿En qué se diferencian? ¿Qué impacto cree que han tenido en la historia de la computación?
3. Se dice que el ENIAC, uno de los primeros computadores digitales, es una máquina Turing-completa. ¿Qué significa este término?
4. Se dice que casi todos los lenguajes de programación son Turing-completos. ¿Por qué cree que un lenguaje de programación se puede equiparar con una máquina de Turing?
5. La concurrencia es la capacidad de un computador de procesar varias tareas al mismo tiempo. ¿Cómo podemos modificar la máquina de Turing para que modele un computador concurrente?
6. ¿Qué orden de complejidad comporta la tarea de extraer el valor máximo de una lista de n números naturales?
7. ¿Qué tipo de problema computacional es el algoritmo de ordenación por el método de la burbuja? Determine la complejidad asintótica de este método de ordenación.
8. Determine la complejidad asintótica de la búsqueda binaria.
9. Un operador lógico AND de n bits está constituido por un array de n puertas AND de 2 entradas. ¿Qué tipo de complejidad tiene en cuanto a coste y retardo?
10. Suponga que un operador de suma consume un tiempo de 2 ns por cada bit de los operandos y que su retardo es lineal con el tamaño w de los mismos. Sabiendo que el operador es de tamaño palabra y que la suma se ha de completar en un periodo de reloj, determine la máxima frecuencia del mismo cuando la palabra es de 32 y de 64 bits.
11. El tiempo de acceso a una memoria es de 60 ns . Calcule cuántos ciclos representa dicho tiempo para una máquina que trabaja a una frecuencia de reloj de 100 MHz y para otra que lo hace a 3 GHz .
12. ¿Cuál es la síntesis de la función XOR basada en puertas universales NAND? Dibuje el esquema de la implementación en la familia RTL. ¿Cuántos transistores emplea? ¿Cuántas resistencias?
13. A partir de la implementación directa de la puerta XOR de 2 entradas en la familia CMOS, dibuje el esquema de una composición XOR de 3 entradas en cascada. ¿Cuántos elementos activos emplea?
14. Entendemos por *fan-in* el número de entradas que pueden atacar o estimular una puerta lógica sin distorsionar su funcionamiento. Asimismo, entendemos por *fan-out* el número de puertas a las que puede excitar una salida directamente sin menoscabo de la señal.
En muchas ocasiones, el resultado de una ecuación lógica impone el uso de puertas lógicas con un número de entradas n que excede su *fan-in* (f_{in}). La solución a este problema de implementación consiste en disponer las puertas lógicas en cascada. Evalúe la complejidad *hardware* tanto en coste como en retardo en función de n y f_{in} .
15. ¿Cómo es la síntesis de un MUX 2:1 con puertas universales NAND? ¿Cómo queda la implementación RTL en ese caso?
16. Sea un repertorio de 16 instrucciones en el que todas se ejecutan en 2 ciclos excepto cinco de ellas que lo hacen en 6 ciclos. Supongamos que un programa utiliza 90 millones de instrucciones rápidas y 10 millones de instrucciones lentas en la mezcla de código. Calcule el monto total de ciclos que se consumen en procesar dicho programa.

17. Si el programa del problema anterior pertenece a una rutina de servicio del sistema operativo que se ejecuta en una máquina que trabaja a 500 MHz, calcule el tiempo que consume e identifique de qué tipo es.
18. La medida de MIPS se puede dar como la frecuencia de reloj dividida entre el CPI y 10^6 . En definitiva, representa una cuantificación relativa a la tecnología (frecuencia de reloj) y a la arquitectura de la máquina (CPI). La tabla siguiente muestra una selección de microprocesadores con su medida de MIPS y la frecuencia de reloj. Calcule el CPI y discuta los resultados.

microprocesador	MIPS	frecuencia (MHz)	año
Intel 8080	0,64	2	1974
VAX-11/780	1	5	1977
Intel 8086	0,8	4,77	1979
Motorola 68000	1	8	1979
Intel i386DX	2,15	16	1985
Intel Pentium	100	60	1993
Alpha 21064	135	200	1993
MIPS R4400	85	150	1993
PowerPC 600s (G2)	35	33	1994
Zilog eZ80	80	50	1999
ARM Cortex A8	2.000	1.000	2005
AMD Athlon 64 Dual Core	18.500	2.200	2005
MIPS64	1.370	600	2007
Intel Core i7 920	82.300	2.930	2008

19. Sea un procesador que trabaja a una frecuencia de reloj de 200MHz. Mediante el uso del comando `time` de Linux sabemos los tiempos empleados por un proceso, a saber, el tiempo de pared o real ha sido de 0.040s, el tiempo de usuario o CPU de 0,020s y el tiempo de sistema 0,016s. Indique qué es cada uno de esos tiempos y expréselos en ciclos.
20. Asumimos que el CPI medio del procesador del problema anterior es 1,5 ciclos/instrucción. Bajo este supuesto determine el recuento del proceso mencionado en dicho problema.
21. Supongamos que gracias a una mejora en el algoritmo empleado para implementar ese proceso se ha reducido en un 5% el recuento. Estime el nuevo tiempo de CPU que podremos esperar al ejecutar el proceso sobre la misma máquina. Calcule el *speed-up* o aceleración.
22. Qué aceleración obtendremos si la frecuencia de reloj crece hasta los 250MHz para la máquina anterior y el mismo proceso con el algoritmo mejorado.
23. Sea un procesador que trabaja a una frecuencia de reloj de 300MHz. Su repertorio de instrucciones produce un *CPI* medio para un conjunto de programas de prueba de 1.2 ciclos/instrucción. Determine el *speed-up* que produciría un aumento de la frecuencia de reloj hasta alcanzar los 350MHz a expensas de que el *CPI* se sitúe en 1.4 ciclos/instrucción. ¿Compensa realmente el aumento de la frecuencia de reloj? Asumimos que el recuento r es constante.