



Solución Interrogación 1

Pregunta 1

- a) Explique en detalle que pasaría en el computador básico si el registro PC cambia su funcionamiento de flanco de subida a flanco de bajada. **(1 pto.)**

Solución: Al cambiar la sincronización, el almacenamiento en los registros o memoria, no coincidiría necesariamente con lo que la instrucción espera almacenar. Dependiendo de la velocidad de ejecución de la instrucción, algunas instrucciones funcionarían de la misma manera, mientras que otras almacenarían resultados incorrectos.

- b) ¿Qué tipo de dato puede representar más valores distintos, **int** de 32 bits o **float** de 32 bits? **(1 pto.)**

Solución: El **int** puede almacenar 2^{32} posibles valores distintos, mientras que en **float**, al tener varias combinaciones de bits que representan lo mismo (NaN), tiene menos de 2^{32} combinaciones posibles disponibles.

- c) Diseñe un circuito que calcule el resultado de la multiplicación de dos números de 1 bit. **(1 pto.)**

Solución: La multiplicación de dos números de 1 bit corresponde a combinar las dos señales con una compuerta AND.

- d) Describa un mecanismo para determinar el endianness de un computador. **(1 pto.)**

Solución: Una posible opción es escribir en memoria un **int** con valor 1 a partir de la posición 0 y luego leer el contenido de esa posición. Si lo leído es 0, el computador utiliza **big endian**, mientras que si es 1, es **little endian**.

- e) Describa en detalle los pasos que debe realizar un **assembler** para procesar código assembly del computador básico. **(1 pto.)**

Solución: El primer paso es registrar todos los *labels* definidos en el segmento **DATA**. Luego, en el segmento **CODE**, sustituir las referencias a estos labels por el literal correspondiente (dir. de memoria). Finalmente, se debe transformar cada instrucción del segmento **CODE** a un opcode.

- f) Demuestre que el complemento a 2 del complemento a 2 de un número x es igual a x , *i.e.*, $x = C_2(C_2(x))$.

Hint: asuma que $C_2(x + y) = C_2(x) + C_2(y)$. **(1 pto.)**

Solución:

$$x + C_2(x) = 0 \quad / \quad C_2$$

$$C_2(x + C_2(x)) = 0 \quad / \quad \text{utilizamos el hint}$$

$$C_2(x) + C_2(C_2(x)) = 0$$

$$C_2(x) + C_2(C_2(x)) = x + C_2(x)$$

$$C_2(C_2(x)) = x$$

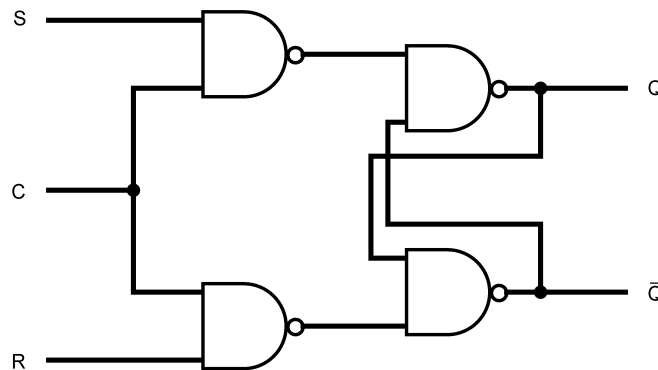
Pregunta 2

- a) Dados números naturales K, N, T , donde $K \geq 2 \wedge N > T > 0$, describa un algoritmo para transformar de manera **eficiente** números naturales codificados en base K^N a base K^T . No es válido utilizar la fórmula general de transformación de bases $\sum_{k=0}^{n-1} s_k \times b^k$. **(2 ptos.)**

Solución: Un algoritmo eficiente puede realizarse en dos pasos: i) Transformar cada cifra del número original en N cifras que representen el mismo número en base K , manteniendo el orden correlativo entre cifras. ii) Agrupar las cifras del número expandido en grupos de T y transformar cada grupo en un número en base K^T , nuevamente manteniendo el orden correlativo.

- b) i) Modifique un latch tipo RS agregando una señal de control C , tal que los cambios en el estado del latch sólo se realicen cuando $C = 1$. **(2 ptos.)**

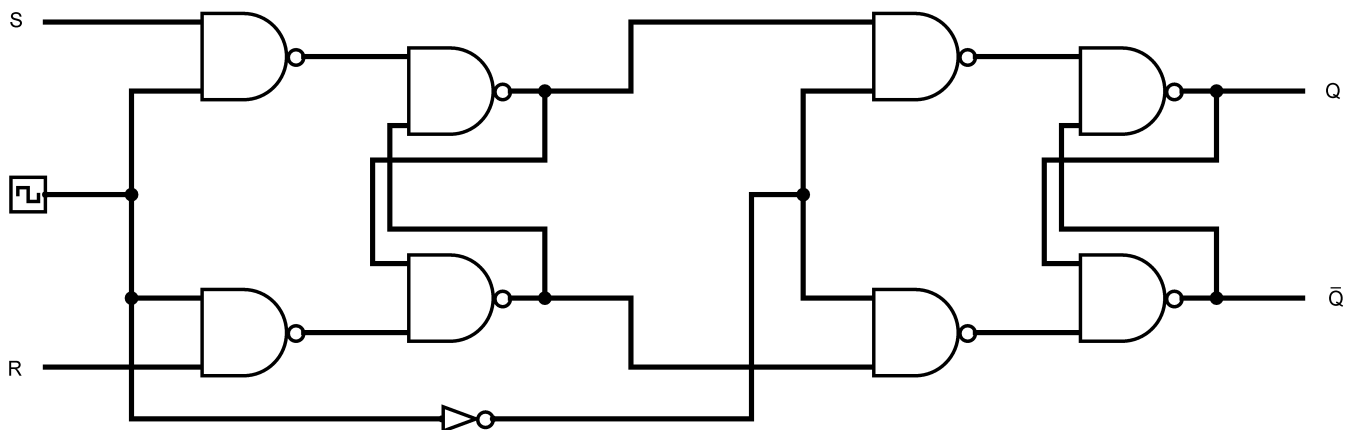
Solución:



- ii) Diseñe un flip-flop tipo RS, dado por la siguiente tabla de verdad: **(2 ptos.)**

S	R	C	Q	\bar{Q}
0	0	\uparrow	Q	\bar{Q}
0	1	\uparrow	0	1
1	0	\uparrow	1	0
1	1	\uparrow	?	?
X	X	0,1, \downarrow	Q	\bar{Q}

Solución:



Pregunta 3

En esta pregunta deberá diseñar un computador especializado en el manejo de matrices. El computador debe ser capaz de: i) copiar una matriz desde la memoria de datos a un registro y viceversa, ii) sumar 2 matrices almacenadas en registros distintos y almacenar el resultado en un registro.

- a) Haga el diagrama del computador, considerando que las matrices pueden tener como máximo $N \times N$ elementos, cada uno de 1 byte. **(3 ptos.)**

Solucion: Basta con utilizar el diagrama del computador básico, aumentando el tamaño de los registros, buses de datos, ALU y las palabras de memoria, a N^2 bytes.

- b) Diseñe el assembly del computador. Cada instrucción debe estar asociada a un opcode y estos a sus respectivas señales de control **(3 ptos.)**

Solucion: En base al diagrama anterior, basta utilizar el assembly del computador básico. La única diferencia se da en los literales, donde existen dos opciones: i) separar cada uno de los elementos de una matriz con coma (,) y ii) declarar la matriz completa como un único número de tamaño N^2 bytes.

- c) **Bonus:** Agregue tanto al hardware como al assembly, soporte para una instrucción que permita modificar el valor de un elemento arbitrario de una matriz almacenada en la memoria de datos. **(3 ptos.)**

Solución: Una posible solución es agregar unidades de shifting para aislar el elemento buscado y modificarlo y luego realizar operaciones lógicas para integrar este valor con la matriz completa.