

Universidad Nacional Autónoma de México

Facultad de Ciencias

Organización y Arquitectura de Computadoras

Tarea 01: Sistemas Numéricos

Álvarez Ríos Metztitlalli | 423052523

Martínez Jiménez Maitreyi | 320099773

Preguntas

1. Expresar -31 y +31 en 8 bits en el sistema de complemento a 1.
Convertimos 31 en binario

$$31 \div 2 = 15 \quad \text{residuo } 1$$

$$15 \div 2 = 7 \quad \text{residuo } 1$$

$$7 \div 2 = 3 \quad \text{residuo } 1$$

$$3 \div 2 = 1 \quad \text{residuo } 1$$

$$1 \div 2 = 0 \quad \text{residuo } 1$$

El número binario es:

11111

Ajustamos a 8 bits

00011111

En complemento a 1 tomamos el valor binario de 31

$$C_1(00011111) = 11111111 - 00011111 = 11100000$$

Por lo tanto +31 en binario es

00011111

-31 en binario es

11100000

2. Expresar +13 y -13c en 8 bits en el sistema de complemento a 2.
Convertimos 13 en binario

$$13 \div 2 = 6 \quad \text{residuo } 1$$

$$6 \div 2 = 3 \quad \text{residuo } 0$$

$$3 \div 2 = 1 \quad \text{residuo } 1$$

$$1 \div 2 = 0 \quad \text{residuo } 1$$

El número binario es:

1101

Ajustamos a 8 bits

00001101

En complemento a 2 tomamos el valor binario de 13

$$C_2(00001101) = 11111111 - 00001101 = 11110010 + 1 = 11110011$$

Por lo tanto +13 en binario es

00001101

-13 en binario es

11110011

3. ¿Cuál es el rango de números representables en complemento a dos con 4 bits?

El rango de números representables en complementos a 2 van desde $-(2^2)$ hasta $+(2^{n-1})$.

Consideremos $n = 4$ bits.

Entonces van desde $-(2^{4-1}) = -(2^3) = -8$ hasta $+(2^{4-1} - 1) = +(2^3 - 1) = +(8 - 1) = +7$

Por lo tanto, el rango de bits va desde $\{-8, \dots, +7\}$

4. El número $(10110101)_2$ es un número de 8 bits incluyendo el bit en complemento a 2. Da su equivalent en número decimal.

Dado que el primer bit es 1 sabemos que el número es negativo. Invertimos el número

$$10110101 = 01001010 + 1 = 01001011$$

Ahora lo convertimos en base 10

$$0 * 2^7 + 1 * 2^6 + 0 * 2^5 + 0 * 2^4 + 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0$$

$$64 + 8 + 2 + 1$$

$$75$$

Y al tener que el bit más significativo es 1 = (-), $(10110101)_2 = -75$

5. El número $(00110111)_2$ es un número de 8 bits incluyendo el bit signo en complemento a 2. Da su equivalente en número decimal.

Observamos que el primer bit es 0, procedemos a convertirlo en base 10

$$0 * 2^7 + 0 * 2^6 + 1 * 2^5 + 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * 2^0$$

$$32 + 16 + 4 + 2 + 1$$

$$55$$

Entonces $(00110111)_2 = +55$

6. Menciona las cuatro unidades funcionales principales de una computadora y describe su funcionamiento.

- **Unidad de control:** Interpreta las instrucciones que son generadas por un programa (una por una) y después inicia las acciones que llevan a cabo las instrucciones.
- **Unidad lógica y aritmética (ALU):** Realiza la operación ordenada por la Unidad de control.
- **Memoria:** Lugar donde el ordenador almacena de forma temporal las instrucciones y recupera datos.
- **Dispositivos de entrada y salida:** Son componentes encargados de extraer e introducir información del ordenador/sistema.

7. Realiza la siguiente operación -3-6 de base decimal en base binario representando los números en 4 bits.

Convertimos 3 en binario

$$3 \div 2 = 1 \quad \text{residuo } 1$$

$$1 \div 2 = 0 \quad \text{residuo } 1$$

El número en binario es

$$11$$

Ajustamos a 4 bits

$$0011$$

Invertimos el número

$$0011 = 1100 + 1 = 1101$$

Por lo tanto -3 en base binario es 1101
 Ahora convertimos 6 en binario

$$\begin{array}{l} 6 \div 2 = 3 \quad \text{residuo } 0 \\ 3 \div 2 = 1 \quad \text{residuo } 1 \\ 1 \div 2 = 0 \quad \text{residuo } 1 \end{array}$$

El número en binario es

$$110$$

Ajustamos a 4 bits

$$0110$$

Invertimos el número

$$0110 = 1001 + 1 = 1010$$

Por lo tanto -6 en base binario es 1010
 Realizamos la suma en binario

$$\begin{array}{r} 1 \quad 1 \quad 0 \quad 1 \\ + \quad 1 \quad 0 \quad 1 \quad 0 \\ \hline 1 \quad 0 \quad 1 \quad 1 \quad 1 \end{array}$$

El acarreo es de 1.

Notemos que el resultado de la resta -9 no se puede representar en 4 bits, ya que solo podemos representar $\{-8, \dots, 7\}$

8. Realiza la siguiente operación $9+3$ de base decimal en base binario representando los números en 4 bits.
 Convertimos 9 en binario

$$\begin{array}{l} 9 \div 2 = 4 \quad \text{residuo } 1 \\ 4 \div 2 = 2 \quad \text{residuo } 0 \\ 2 \div 2 = 1 \quad \text{residuo } 0 \\ 1 \div 2 = 0 \quad \text{residuo } 1 \end{array}$$

El número en binario es

$$1001$$

Ahora convertimos a 3 en binario

$$\begin{array}{l} 3 \div 2 = 1 \quad \text{residuo } 1 \\ 1 \div 2 = 0 \quad \text{residuo } 1 \end{array}$$

Ajustamos a 4 bits

0011

Realizamos la suma en binario

$$\begin{array}{r} 1 \ 0 \ 0 \ 1 \\ + \ 0 \ 0 \ 1 \ 1 \\ \hline 1 \ 1 \ 0 \ 0 \end{array}$$

Pasamos 1100 a base 10, por lo que podemos ver que efectivamente es 12.

9. Realiza la siguiente suma de 2 bits.

A		B		Acarreo	Suma
0	+	0	=	0	0
0	+	1	=	0	1
1	+	0	=	0	1
1	+	1	=	1	0

10. Suma los siguientes dos números $(10011011)_2 + (11101100)_2$. Explica qué sucede con el acarreo.

Acarreo	1	1	1	1					
	1	0	0	1	1	0	1	1	
+	1	1	1	0	1	1	0	0	
1	1	0	0	0	0	1	1	1	

Si la suma de los bits más el acarreo es 2 o más lo llevamos al bit a la izquierda.

11. Representa el número 39,1 en base 2 usando el estándar IEEE 754. Convertimos 39 en binario

$$\begin{array}{l} 39 \div 2 = 19 \quad \text{residuo } 1 \\ 19 \div 2 = 9 \quad \text{residuo } 1 \\ 9 \div 2 = 4 \quad \text{residuo } 1 \\ 4 \div 2 = 2 \quad \text{residuo } 0 \\ 2 \div 2 = 1 \quad \text{residuo } 0 \end{array}$$

Por lo tanto el número en binario es

100111

Convetimos 0,1 en binario

$$0,1x2 = 0,2 \quad \text{parte entera } 0$$

$$0,2x2 = 0,4 \quad \text{parte entera } 0$$

$$0,4x2 = 0,8 \quad \text{parte entera } 0$$

$$0,8x2 = 1,6 \quad \text{parte entera } 1$$

$$0,6x2 = 1,2 \quad \text{parte entera } 1$$

Por lo tanto 0,1 en binario es aproximadamente:

$$00011$$

El resultado de 39,1 en binario es

$$100111,00011$$

Para el estándar IEEE 754 normalizamos

$$100111,00011 = 1,0011100011x2^5$$

Sesgar el exponente

$$127 + 5 = 132$$

Pasamos a binario

$$132 \div 2 = 66 \quad \text{residuo } 0$$

$$66 \div 2 = 33 \quad \text{residuo } 0$$

$$33 \div 2 = 16 \quad \text{residuo } 1$$

$$16 \div 2 = 8 \quad \text{residuo } 0$$

$$8 \div 2 = 4 \quad \text{residuo } 0$$

$$4 \div 2 = 2 \quad \text{residuo } 0$$

$$2 \div 2 = 1 \quad \text{residuo } 0$$

$$1 \div 2 = 0 \quad \text{residuo } 1$$

Por lo tanto 32 en binario es

$$10000100$$

Juntamos la mantisa normalizada con el cambio de base del exponente y agregamos bits hasta que sean los 23 bits, lo que nos queda

$$01000010000111000110000000000000$$

12. Representa el número 576,65 en base 2 usando el estándar IEEE 754.

$$\begin{aligned}576 \div 2 &= 288 && \text{residuo } 0 \\288 \div 2 &= 144 && \text{residuo } 0 \\144 \div 2 &= 72 && \text{residuo } 0 \\72 \div 2 &= 36 && \text{residuo } 0 \\36 \div 2 &= 18 && \text{residuo } 0 \\18 \div 2 &= 9 && \text{residuo } 0 \\9 \div 2 &= 4 && \text{residuo } 1 \\4 \div 2 &= 2 && \text{residuo } 0 \\2 \div 2 &= 1 && \text{residuo } 0\end{aligned}$$

Por lo tanto el número en binario es

$$1001000000$$

$$\begin{aligned}0,65 \times 2 &= 1,3 && \text{parte entera } 1 \\0,3 \times 2 &= 0,6 && \text{parte entera } 0 \\0,6 \times 2 &= 1,2 && \text{parte entera } 1 \\0,2 \times 2 &= 0,4 && \text{parte entera } 0 \\0,4 \times 2 &= 0,8 && \text{parte entera } 0 \\0,8 \times 2 &= 1,6 && \text{parte entera } 1\end{aligned}$$

$$1001000000.101001$$

Normalizamos la mantiza:

$$1001000000.101001$$

$$1,0010000001010012^9$$

Sesgar el exponente

$$127 + 9 = 136$$

Pasamos a binario

$$\begin{aligned}136 \div 2 &= 68 && \text{residuo } 0 \\68 \div 2 &= 34 && \text{residuo } 0 \\34 \div 2 &= 17 && \text{residuo } 0 \\17 \div 2 &= 8 && \text{residuo } 1 \\8 \div 2 &= 4 && \text{residuo } 0 \\4 \div 2 &= 2 && \text{residuo } 0 \\2 \div 2 &= 1 && \text{residuo } 0\end{aligned}$$

Por lo tanto 32 en binario es

10001000

Juntamos el signo, es decir, 0, con el exponente sesgado (10001000) y con la mantiza normalizada (001000000101001 000000000)

01000100000100000010100100000000

13. ¿Qué ventajas y desventajas puedes encontrar en el modelo de la arquitectura de Von Neumann? Argumenta tu respuesta.

Ventajas:

- La estructura de memoria y el bus de sistema único hacen que la arquitectura sea fácil de entender e implementar.
- El procesamiento de las instrucciones es secuencial.
- Los datos e instrucciones comparten el mismo espacio en la memoria.
- Los componentes están interconectados a través de una vía de comunicación central.

Desventajas:

- Utiliza un único pozo de RAM en donde se almacenan las instrucciones y datos, estos mismos se capturan de manera secuencial desde la memoria el mismo tiempo. Se le llama cuello de botella.
 - Tiene limitación en la longitud de las instrucciones debido al bus de datos, hace que el microprocesador realice múltiples accesos a memoria para buscar instrucciones complejas.
 - Se hace lenta al usar más ciclos para realizar una tarea.
 - La ejecución lineal de las instrucciones puede limitar el paralelismo.
14. La Arquitectura Von Neuman fue descrita por el matemático y físico John Von Neumann y otros, en *el primero borrador de un informe sobre el ED-VAC*. Pero la computación de 1945 a la actualidad ha dado pasos gigantados, aumentando la complejidad de la arquitectura inicial, la base de su funcionamiento es la misma. ¿Qué cambios aprecias hoy en día en tu computador que no se ven descritos por el diagrama dado en 1945? Argumenta tu respuesta.

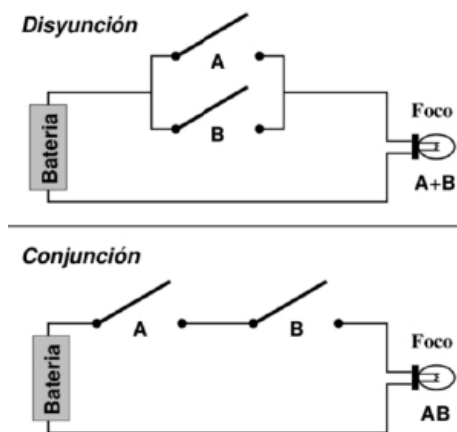
El diseño de Von Neumann se basa en el CPU (unidad central de procesamiento), la memoria y los dispositivos de entrada y salida (como un teclado y pantalla), pero ha habido varios cambios grandes tanto para el software como para el hardware, como las tarjetas gráficas y la RAM. así como los procesadores, ya que para el diseño de 1945 solo se consideraba un procesador para ejecutar las instrucciones de forma secuencial.

Otro de los grandes cambios y avances es el almacenamiento, para 1945 el almacenamiento no era masivo como en la actualidad, ahora contamos con discos duros y almacenamiento en la nube.

Las computadoras actuales ya son más compactas y menos costosas que en ese entonces.

Y también las computadoras en la actualidad tienen varios núcleos de procesamiento CPU y GPU para que las tareas se puedan hacer de forma simultánea y es un gran contraste con el diseño de Von Neumann de 1945, que solo consideraba un solo procesador para realizar todas.

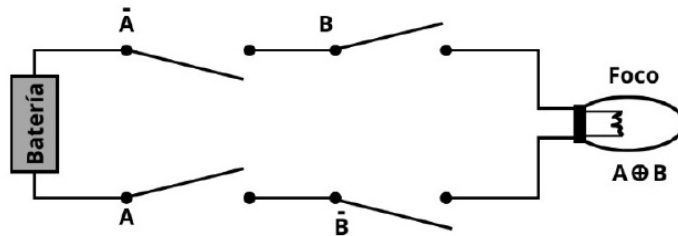
15. En la siguiente imagen, se nos muestra la disyunción y la conjunción proposicional usando interruptores.



Usando ese mismo modelo ¿cómo sería un xor usando interruptores?
Empezamos por la tabla de verdad

\bar{A}	\cdot	B	+	A	\cdot	\bar{B}
1	0	0	0	0	0	1
1	1	1	1	0	0	0
0	0	1	0	1	0	0
0	0	1	0	1	0	0

Y el XOR usando interruptores queda de la siguiente forma



Referencias

- [1] Alonso, Rodrigo. (2024). Arquitectura Von Neuman: qué es, cómo funciona y para qué sirve. Recuperado de: <https://compilandoconocimiento.com/2017/01/29/arquitecturasvon-newmanvsharvard/>
- [2] Rosas, O. (2017). Arquitecturas Von-Newman VS Harvard. Recuperado de: <https://hardzone.es/tutoriales/rendimiento/von-neumann-limitaciones/>
- [3] StudySmarter. (SF). Arquitectura de Von Neumann. Recuperado de: <https://www.studysmarter.es/resumenes/ciencias-de-la-computacion/organizacion-y-arquitectura-de-computadoras/arquitectura-de-von-neumann/>
- [4] KeepCoding. (2024). ¿Qué es el modelo de von Neumann y cómo cambió la programación?. Recuperado de: <https://keepcoding.io/blog/que-es-el-modelo-de-von-neumann/>
- [5] Cardona, R. (2023). John von Neumann, el matemático que diseñó los ordenadores modernos. Recuperado de: <https://elpais.com/ciencia/cafe-y-teoremas/2023-02-23/john-von-neumann-el-matematico-que-diseno-los-ordenadores-modernos.html>