



Universidad Autónoma de Baja California
Facultad de Ingeniería Arquitectura y Diseño



Ingeniería en Software y Tecnologías Emergentes

Materia: Organización de computadoras

Alumno: Jesus Eduardo Rodríguez Ramírez

Profesor: Jonatan Crespo Ragland

Grupo 932

Trabajo: Taller 4

Ensenada, B.C; a 17 de septiembre del 2024

A. Investiga para responder lo siguiente en tu cuaderno:

1. ¿Cómo se realiza la operación de desplazamientos en binario (shift left y shift right), y para qué se utilizan?

Shift left: Mueve los bits hacia la izquierda y rellena con ceros a la derecha. Equivale a multiplicar por potencias de dos.

Shift right: Mueve los bits hacia la derecha y rellena con ceros a la izquierda. Equivale a dividir por potencias de dos.

Son utilizados para optimización de algoritmos, procesamiento de señales y manipulación de bits.

2. ¿Por qué es importante la aritmética binaria en el diseño y funcionamiento de los sistemas informáticos? Facilita el diseño de circuitos electrónicos con solo dos estados (0 y 1).

Menos susceptible a interferencias y ruidos. Base para codificar y procesar toda la información. Permite realizar operaciones fundamentales de manera eficiente. Esencial para la seguridad y privacidad de la información.

3. ¿Qué es un "overflow" en la aritmética binaria y cuándo ocurre?

Ocurre cuando el resultado de una operación aritmética excede el rango de valores representables con los bits disponibles.

Por ejemplo: Cuando una suma o resta resulta un número fuera del rango representable.

16/sep/24.

4. ¿Cuál es la diferencia entre la multiplicación binaria y la multiplicación en el sistema decimal? Tienen algunas diferencias clave debido a las bases numéricas en las que operan.

En binario se usa base 2 y en decimal base 10. Además en binario el proceso es multiplicación y suma de productos parciales desplazados.

5. ¿Cómo se puede realizar la división binaria y en qué se parece o difiere de la división decimal? Para la división binaria primero se escribe el dividendo y divisor en formato de división larga, después se compara el divisor con los bits del dividendo, de izquierda a derecha.

Si el divisor es menor o igual al segmento del dividendo, escribir 1 en el cociente y restar el divisor del segmento del dividendo. Bajar el siguiente bit del dividendo y repetir el proceso hasta completar la división. Comparando la división binaria con la decimal podemos decir que se parecen en que ambos procesos implican comparar, dividir, restar y desplazar. Además ambos utilizan la división larga como método principal.

6. ¿Qué es el complemento a dos y cómo se utiliza en la aritmética binaria? El complemento a dos es un método utilizado en aritmética binaria para representar números enteros negativos y realizar operaciones con ellos, como la suma y la resta. Primero, escribe la representación binaria del número positivo. Cambia cada 1 por un 0 y cada 0 por un 1. Esto se conoce como el complemento a uno. Al resultado del paso anterior, se le suma 1. El valor resultante es la representación en complemento a dos del número negativo original.

7. ¿Cuál es la diferencia entre el complemento a uno y el complemento a dos? La principal diferencia entre el complemento a uno y el complemento a dos radica en la forma en la que se calculan y en cómo se representan los números negativos.

8. ¿Por qué el complemento a dos es el método preferido para representar números negativos en los sistemas digitales? El complemento a dos es el preferido porque ofrece una única representación del cero, simplifica la aritmética binaria al unificar la suma y la resta, permite un rango simétrico de números y simplifica el diseño de los circuitos dentro de las computadoras. Esto lo convierte en el método más eficiente y directo para manejar números negativos en sistemas digitales.

9. ¿Cómo afecta el tamaño del registro (cantidad de bits) al rango de números representables en complemento a dos? El tamaño del registro, es decir, la cantidad de bits que tenga, determina el rango de números que se pueden representar en complemento a dos. Entonces, cuanto más grande sea el registro, mayor será el rango de números que se pueden representar.

10. ¿Qué ventajas tiene el complemento a dos frente a la representación con signo y magnitud? El complemento a dos es más eficiente para realizar operaciones aritméticas, ocupa menos espacio al representar el cero y facilita el manejo de números negativos en comparación con la representación con signo y magnitud.

11. ¿Por qué la representación de signo y magnitud no es la más comúnmente utilizada en los sistemas informáticos? La representación de signo y magnitud no es común en sistemas informáticos porque:

1. Tiene dos representaciones para el cero ($+0$ y -0).
2. Las operaciones aritméticas son más complejas.
3. El manejo del desbordamiento es complicado.
4. Es menos eficiente que el complemento a dos.

12. ¿Qué diferencia existe entre la representación de signo y magnitud y la representación sin signo? La representación de signo y magnitud permite representar números negativos y positivos, pero con ciertas complicaciones, mientras que la representación sin signo es más simple y solo puede representar números positivos.

13. Describe una metodología a seguir para realizar la resta de números binarios, incluyendo el caso para cuando los números a restar tienen una longitud diferente.

1. Alinear los números: Igualar las longitudes agregando ceros a la izquierda del número más corto.
2. Restar bit a bit: Aplicar las reglas de la resta binaria, manejando los préstamos cuando sea necesario.
3. Eliminar ceros a la izquierda: Simplificar el resultado eliminando ceros innecesarios.

17/sep/24.

B. Convierte los siguientes números en decimal a binario y realiza la operación correspondiente, con procedimiento. Para el caso de las restas, usa la metodología que encontraste para el inciso B (Recuerda que tanto para la suma o resta, puedes inferir el resultado utilizando una resta de los números decimales):

14. $30 + 65$

$$\begin{array}{r} 11110 \\ + 1000001 \\ \hline 1011111 = 95 \end{array}$$

18. $101 + 102$

$$\begin{array}{r} 1100101 \\ + 1100110 \\ \hline 11001011 = 203 \end{array}$$

15. $64 + 102$

$$\begin{array}{r} 1000000 \\ + 01100110 \\ \hline 10100110 = 166 \end{array}$$

19. $100 - 21$

$$\begin{array}{r} 1100100 \\ - 10101 \\ \hline 1001111 = 79 \end{array}$$

16. $121 + 45$

$$\begin{array}{r} 1111001 \\ + 101101 \\ \hline 10000110 = 166 \end{array}$$

20. $99 - 32$

$$\begin{array}{r} 1100011 \\ - 100000 \\ \hline 1000011 = 67 \end{array}$$

17. $44 + 230$

$$\begin{array}{r} 101100 \\ + 11100110 \\ \hline 100010010 = 274 \end{array}$$

21. $78 - 21$

$$\begin{array}{r} 1001110 \\ - 10101 \\ \hline 111001 = 57 \end{array}$$

22. $51 - 45$

$$\begin{array}{r} 110011 \\ - 101101 \\ \hline 000110 = 6 \end{array}$$

23. $50 - 32$

$$\begin{array}{r} 110010 \\ - 100000 \\ \hline 010010 = 18 \end{array}$$

24. 32 en su complemento 2

$32 = 00100000$

$\text{comp. en } 2 = 11100000$

25. 21 en su complemento a 2

$21 = 00010101$

$\text{comp. en } 2 = 11101011$

26. 12 en su complemento a 2

$12 = 00001100$

$\text{comp. en } 2 = 11110100$