TAREA 3

Microprocesadores – ingeniería informática ubp

baigorria constanza / gomez martina / sposetti leonel

2023

## ACTIVIDAD 1

Represente tanto en signo-magnitud como en complemento a dos, con 16 bits, los siguientes números decimales: +512; -29.

512(10) = 0000001000000000(2)

Complemento a dos: 1111111000000000(2)

-29(10) = 1000000000011101(2)

Complemento a dos: 0111111111100011(2)

## ACTIVIDAD 2

Represente en decimal los siguientes valores en complemento a dos: 1101011; 0101101.

21(10) = 00010101(2)

Complemento a dos: 11101011(2)

19(10) = 00010011(2)

Complemento a dos: 11101101(2)

## ACTIVIDAD 3

Considere los números representados en complemento a dos con ocho bits y realice los siguientes cálculos: (a) 6 + 13 (b) -6 + 13 (c) 6 - 13 (d) -6 – 13.

1. 00000110(2) + 01101101(2) = 01110011(2)
2. 11111010 (2) + 00001101(2) = 10000111(2)
3. 00000110 (2) + 11110011 (2) = 11111001(2)
4. 11111010 (2) + 11110011 (2) = 111101101(2)

## ACTIVIDAD 4

Divida 145 entre 13 en notación binaria de complemento a dos utilizando palabras de 12 bits.

145(10) = 000010010001(2)

Complemento a dos: 111101101111(2)

13(10) = 000000001101(2)

Complemento a dos: 111111110011(2)

111101101111(2) / 111111110011(2) = 000000010000(2)

## ACTIVIDAD 5

Use el algoritmo de Booth para multiplicar 23 (multiplicando) por 29 (multiplicador), donde cada número está representado con 7 bits.

23(10) = 0010111(2)

29(10) = 0011101(2)

A = 0

M = 0010111

Q = 0011101

Q – 1 = 0

ITERACIÓN 1:

SI LOS DOS ÚLTIMOS BITS DE Q Y Q1 SON 01 O 10, DESPLAZAMIENTO ARITMÉTICO A LA DERECHA Y SUMA M AL ACUMULADOR.

* Q: 0101111
* Q1: 0

LOS DOS ÚLTIMOS BITS DE Q Y Q1 SON 11, POR LO QUE DESPLAZAMOS A LA DERECHA Y SUMAMOS M.

* Q: 0010111 (DESPLAZADO A LA DERECHA)
* Q1: 1 (GUARDAMOS EL ÚLTIMO BIT DE Q ANTES DE DESPLAZAR)
* A: 0000000
* M: 1000011
* A = A + M: 1000011 (SUMA M AL ACUMULADOR)

ITERACIÓN 2:

DESPLAZAMIENTO ARITMÉTICO A LA DERECHA.

* Q: 0010111
* Q1: 1

DESPLAZAMIENTO A LA DERECHA.

* Q: 0001011 (DESPLAZADO A LA DERECHA)
* Q1: 0 (GUARDAMOS EL ÚLTIMO BIT DE Q ANTES DE DESPLAZAR)
* A: 1000011

ITERACIÓN 3:

SI LOS DOS ÚLTIMOS BITS DE Q Y Q1 SON 01 O 10, DESPLAZAMIENTO ARITMÉTICO A LA DERECHA Y SUMA M AL ACUMULADOR.

* Q: 0001011
* Q1: 0

LOS DOS ÚLTIMOS BITS DE Q Y Q1 SON 01, POR LO QUE DESPLAZAMOS A LA DERECHA Y SUMAMOS M.

* Q: 0000101 (DESPLAZADO A LA DERECHA)
* Q1: 1 (GUARDAMOS EL ÚLTIMO BIT DE Q ANTES DE DESPLAZAR)
* A: 1000011
* A = A + M: 1110110 (SUMA M AL ACUMULADOR)

ITERACIÓN 4:

DESPLAZAMIENTO ARITMÉTICO A LA DERECHA.

* Q: 0000101
* Q1: 1

DESPLAZAMIENTO A LA DERECHA.

* Q: 0000010 (DESPLAZADO A LA DERECHA)
* Q1: 0 (GUARDAMOS EL ÚLTIMO BIT DE Q ANTES DE DESPLAZAR)
* A: 1110110

ITERACIÓN 5:

SI LOS DOS ÚLTIMOS BITS DE Q Y Q1 SON 01 O 10, DESPLAZAMIENTO ARITMÉTICO A LA DERECHA Y SUMA M AL ACUMULADOR.

* Q: 0000010
* Q1: 0

LOS DOS ÚLTIMOS BITS DE Q Y Q1 SON 10, POR LO QUE DESPLAZAMOS A LA DERECHA Y SUMAMOS M.

* Q: 0000001 (DESPLAZADO A LA DERECHA)
* Q1: 0 (GUARDAMOS EL ÚLTIMO BIT DE Q ANTES DE DESPLAZAR)
* A: 1110110
* A = A + M: 0110011 (SUMA M AL ACUMULADOR)

ITERACIÓN 6:

DESPLAZAMIENTO ARITMÉTICO A LA DERECHA.

* Q: 0000001
* Q1: 0

DESPLAZAMIENTO A LA DERECHA.

* Q: 0000000 (DESPLAZADO A LA DERECHA)
* Q1: 1 (GUARDAMOS EL ÚLTIMO BIT DE Q ANTES DE DESPLAZAR)
* A: 0110011

ITERACIÓN 7:

DESPLAZAMIENTO ARITMÉTICO A LA DERECHA.

* Q: 0000000
* Q1: 1

DESPLAZAMIENTO A LA DERECHA.

* Q: 0000000 (DESPLAZADO A LA DERECHA)
* Q1: 0 (GUARDAMOS EL ÚLTIMO BIT DE Q ANTES DE DESPLAZAR)
* A: 0110011

PRODUCTO FINAL:

667(10) = 0000001010011011(2)