

CAPITULO SEGUNDO.

REPRESENTACION DE LA INFORMACION

Estructura de Ordenadores.

Departamento de
Automática

Juana M^a López

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 1

INTRODUCCION

■ Clasificación de las representaciones Numéricas:

- **Sistemas posicionales:** A cada bit le corresponde un peso dependiendo de su posición.
 - » Coma fija
 - » Coma flotante
- **Sistemas de Residuos:**
 - » Basados en representación de residuos resultantes de dividir un número entre otro tomado como divisor general.

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| P7 | P6 | P5 | P4 | P3 | P2 | P1 | P0 |
|----|----|----|----|----|----|----|----|

A cada posición corresponde un peso
(Sistemas posicionales)

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 2

COMA FIJA

- Representan números enteros.
 - Sin signo o binario puro.
 - Con signo:
 - » Signo-magnitud.
 - » Complemento a dos.
 - » Complemento a uno.
 - » Exceso a M.
 - BCD o decimal codificado como binario.



Dpto. Automática. Estructura de computadores. Capítulo 2. Página 3

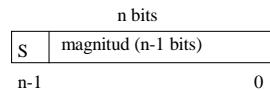
Binario Puro

| n=8 | | | | | | | |
|-----|----|----|----|----|----|----|----|
| x7 | x6 | x5 | x4 | x3 | x2 | x1 | x0 |
| n-1 | | | | | | | 0 |

- Los pesos son:
 - $P_i = 2^i$
- El nº representado tiene el valor:
 - valor = $\sum_{i=0}^{n-1} 2^i * x_i$
- Sistema posicional de base 2 y sin parte fraccionaria.
- Rango de representación [0, $2^n - 1$]
- Dificultades:
 - Posible Overflow
 - No posible resta de resultado negativo.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 4

Signo-magnitud



- Bit de mayor peso indica signo:
 - 0-positivo
 - 1-negativo
- Dificultades en suma y resta.
- Simple en multiplicación y división.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 5

COMPLEMENTO A DOS

- Se reservan los códigos que empiezan por cero para los n° positivos: (00...000 al 011...11).
- Se reservan los que empiezan por uno para los n° negativos (100..00 al 111..11).
- El MSB **indica** el signo pero **no es el bit de signo**.
- Positivos en binario puro y negativos en complemento a dos:
 - » $C_2(A) = 2^n - A$ para $n = n^{\circ}$ bits usado.
- Suma y resta sin tener en cuenta signos.
 - » $A+B=A+B$
 - » $-A-B=(2^n-A)+(2^n-B)=(2^n-(A+B))+\text{acarreo}$
 - » $A-B=A+(2^n-B)=2^n+A-B$
- Dos métodos de cálculo de complemento a dos:
 - » 2^n-A
 - » $A+1$
- Extensión del signo: copiando el MSB ala izquierda tantas veces como sea la extensión.
- Puede haber overflow
- Operación multiplicación compleja.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 6

Suma y resta sin tener en cuenta signos.

$$A+B = A+B$$

$$-A-B = (2^n - A) + (2^n - B) = (2^n - (A+B)) + \text{acarreo}$$

$$A-B = A+(2^n - B) = 2^n + A - B$$

- $A+B$ Como es una cantidad positiva, su complemento a dos es la misma cantidad sin modificar nada.

- $-A-B = (2^n - (A+B)) + 2^n$ (acarreo)

- Simplemente sumando los datos que me llegan en complemento a dos obtengo este resultado. Como se ve, consta de:

- » complemento a dos del número negativo

- $A-B$ pues el complemento a dos de $-X$ es $(2^n - X)$, entonces el complemento a dos de

- $-A-B = -(A+B)$ es igual a $(2^n - (A+B))$

- » añadido un 2^n que es un acarreo por tanto ese acarreo debe despreciarse para que el resultado a la salida sea correcto.

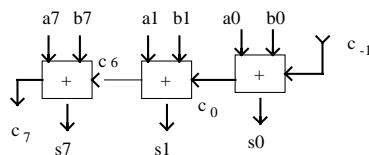
- $A-B = 2^n + A - B$

- Si $A-B$ es positivo su complemento a dos es él mismo por tanto sobra el 2^n (acarreo) y se desprecia.

- Si $A-B$ es negativo, es de la forma $-X$ donde X será $X = (B-A)$, por tanto su complemento a dos es $2^n - X$ es decir $2^n - (B-A) = 2^n + A - B$ luego el resultado es correcto.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 7

Esquema para Overflow



- Sumador paralelo para $n=8$

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 8

OVERFLOW en Complemento a dos.

- Solo puede producirse si:
 - Los dos son positivos.
 - Los dos son negativos.
- Si signo distinto, la suma tendrá de valor absoluto:
 $|A|+|B|$ y como $|A|<2^n$ y $|B|<2^n \Rightarrow |A|+|B|<2^n$
- Condición de desbordamiento:
 $\blacksquare DE = a_{n-1} * b_{n-1} * c_{n-2} + a_{n-1} * b_{n-1} * c_{n-2}$
- Teniendo en cuenta que:
 $\blacksquare C_{n-1} = a_{n-1} * b_{n-1} + b_{n-1} * C_{n-2} + a_{n-1} * C_{n-2}$
- Entonces $\Rightarrow DE = C_{n-1} \text{ XOR } C_{n-2}$

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 9

COMPLEMENTO A UNO

- Se reservan los códigos que empiezan por cero para los nº positivos: (00...000 al 011...11).
- Se reservan los que empiezan por uno para los nº negativos (100..00 al 111..11).
- El MSB **indica** el signo pero **no es el bit de signo**.
- Positivos en binario puro y negativos en complemento a uno:
 - » $C1(A) = 2^{n-1} - A$ para $n = n^{\circ}$ bits usado
- El cálculo del complemento a uno se reduce al complemento lógico.
- El cero tiene doble representación pero el rango es simétrico.
- Extensión de signo copiando el MSB en los bits de la izquierda.
- Puede haber overflow.
- Operación de multiplicación más compleja.
- La suma y resta pueden necesitar correcciones.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 10

COMPLEMENTO A UNO (cont)

- Las operaciones de suma y resta pueden necesitar correcciones:
 - » $A+B=A+B$
 - » $-A-B=(2^n-1-A)+(2^n-1-B)=(2^n-1-A-B)+2^n-1 \Rightarrow (2^n-1-(A+B))+\text{acarreo-1}$
- El resultado necesita que se sume 1: recircular acarreo.
- $A-B=A+(2^n-1-B)=2^n-1+A-B$
 - Si $A > B$ 2^n es un acarreo y hay que sumarlo al resultado.
 - Si $A < B$ valor correcto.
- Desbordamiento equivalente al de complemento a dos.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 11

EXCESO A M

- A se representa como $A+M$.
- M suele valer 2^n-1 .
- Se emplea para representar exponentes en coma flotante.
- EJEMPLO:
 - $n=8 \Rightarrow M=2^n-1=128$
 - » $-3 \Rightarrow 125 \Rightarrow 0111\ 1101$
 - » $0 \Rightarrow 128 \Rightarrow 1000\ 0000$
 - » $-128 \Rightarrow 0 \Rightarrow 0000\ 0000$
 - » $127 \Rightarrow 255 \Rightarrow 1111\ 1111$

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 12

BCD

- EJEMPLOS
 - » 3 0011
 - » 79 0111 1001
 - » 154 0001 0101 0100
 - » 06 0000 0110
- Dos clases:
 - » BCD empaquetado.
 - » BCD desempaquetado.

| | |
|------|-----|
| byte | |
| 0000 | BCD |

Desempaquetado

| | |
|------|-----|
| byte | |
| BCD | BCD |

Empaquetado

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 13

COMA FLOTANTE

| | | | |
|------------------------|----------------------|-------|---|
| n bits | | | |
| E exponente(q bits) | M mantisa(p bits) | | |
| $n-1$ | p | $p-1$ | 0 |

$$\boxed{\text{Valor} = M \cdot r^E}$$

- También llamado “punto flotante”.
- En coma fija sólo representábamos enteros; ahora podemos representar números fraccionarios cómodamente.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 14

COMA FLOTANTE

- Añaden a cada número un factor de escala.
- M y E se representan en coma fija.
- Base de M debe coincidir con r de forma que desplazamiento de coma sea incremento o decremento de E. Bases mas utilizadas son 2 y 16.
- E suele tener base 2 y se representa en exceso 2^{q-1} donde q es la longitud del exponente.
- M puede ser : entero; o fracción con la coma a la izquierda de m_{p-1} o m_{p-2} .
- Clasificación:
 - Con mantisa entera.
 - Con mantisa fraccionaria
 - » No normalizada.
 - » Normalizada.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 15

EJEMPLOS



- Bit 0 de signo.
- Exponente de base 2.
- Exponente en bits de 1-8. En exceso de 128.
- Exponentes representables entre 00000000 (-128) y 11111111 (127)
- Mantisa con coma a la izquierda (el bit 9 tiene peso -1).
- Ejercicio: ¿Que representa 00010010111000..000?

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 16

EJERCICIO

- ¿Que representa 0 00100101 1100000000000?
- Signo 0: (positivo)
- Exponente:
- $00100101_2 = 2^5 + 2^2 + 2^0 = 37_{10}$
- E= 37-128= - 91
- Mantisa:
- M= 0,11 $_2 = 2^{-1} + 2^{-2} = 0,75_{10}$
- Luego valor= $0,75 \cdot 2^{-91} = 3,03 \cdot 10^{-28}$

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 17

Coma flotante con mantisa entera

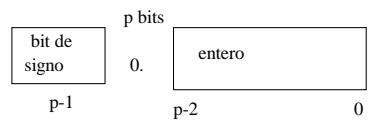
| | exponente | signo | mantisa | | |
|--------|-----------|-------|---------|--------------|-------|
| 100000 | 0 | 00000 | 01101 | 13.2^0 | 13 |
| 100000 | 1 | 00000 | 01100 | -12.2^0 | -12 |
| 100010 | 0 | 00000 | 00101 | 5.2^2 | 20 |
| 011100 | 1 | 00000 | 10100 | -20.2^{-4} | -1.25 |

- Ya no se usa esta representación.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 18

Coma flotante con mantisa fraccionaria

- Mantisa Fraccionaria no normalizada.
 - Ver figura.
- Mantisa normalizada.
 - Lo mismo normalizando la mantisa (eliminando dígitos no significativos a la derecha de la coma). De esta forma se aprovechan al máximo dígitos disponibles.



Valor de la mantisa (con base de exp. 2):

$$\text{entero}/2^{p-1}$$

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 19

EXAMEN para subir nota.

- **PROBLEMA 1.** Dados los datos en complemento a dos:
 - » A=010011
 - » B=010111
 - » Calcular paso a paso en complemento a dos:
 - A+B
 - -A-B
 - A-B

- **PROBLEMA 2.** Dados los datos en complemento a uno:
 - » A=011011
 - » B=000011
 - » Calcular en complemento a uno, las mismas operaciones que en el problema anterior.

- **Nota** para los dos problemas: Expresar el resultado en 6 bits dejando ver los pasos seguidos al operar e indicando, en su caso, si hubo overflow y si se tiene en cuenta el acarreo.

Dpto. Automática. Estructura de computadores. Capítulo 2. Página 20