**Representação circular:** Quando o valor imediatamente a seguir ao maior valor que uma arquitetura de barramento n pode tomar (2^n-1) é o zero. Visto tambem como um array circular.

**Representação de sinal e modulo:**

**Num número de n bits:**

* Bit mais significativo é usado para representar o sinal.
* Magnitude é representada pelos n-1 LSBits.
* Gama de representação:
* 2 representações para o 0

**Exemplo: 4 bits:**

* Bit mais significativo: 4º bit
* Magnitude: 0(000) a 0(111)
* Gama de representação:
* 2 representações para zero: 0(000) e 1(000)

**Problemas:**

* Existem duas representações para um mesmo valor (o zero...)
* Necessário implementar um somador e um subtrator distintos
* Temos que comparar as magnitudes dos operandos para saber o sinal do resultado.
* Bit de sinal tem que ser tratado independentemente dos restantes.

**Algoritmo:**

* Olhar para os sinais
* Se forem iguais faço a soma:
* se forem diferentes faço a subtração:
* Acrescentar o sinal do maior ao resultado

**Overflow:**

* Quando o carry\_ out é

**Representação em complemento para um:**

**Como funciona:**

* Inverter os bits caso o número for negativo

**Problemas:**

* 2 representações para o mesmo número (o zero)
* Exemplo: -2+5=3, passamos pelos dois zeros para ir para o +3, **precisamos de somar mais 1**

**Representação em complemento para dois:**

|  |  |
| --- | --- |
| **0 000** | **+0** |
| **0 001** | **+1** |
| **0 010** | **+2** |
| **0 011** | **+3** |
| **0 100** | **+4** |
| **0 101** | **+5** |
| **0 110** | **+6** |
| **0 111** | **+7** |
| **1 000** | **-8** |
| **1 001** | **-7** |
| **1 010** | **-6** |
| **1 011** | **-5** |
| **1 100** | **-4** |
| **1 101** | **-3** |
| **1 110** | **-2** |
| **1 111** | **-1** |

* Única representação para o número zero
* O bit mais significativo pode ser representado como o de sinal
* Codificação assimétrica: mais um negativo do que positivos
* A subtração é obtida pela soma com o complemento para dois:
  + .

**Passagem de complemento para dois para decimal:**

* Meter um sinal de menos no MSB:
* Exemplo:

**Overflow:**

* Quando a soma de dois positivos dá negativo (ultrapassa o limite dos negativos e como o array é circular, vai para os Positivos)
* Quando a soma de dois negativos dá positivo
* **Ou: quando o bit carry-in do bit de sinal não é igual ao bit carry-out:**

**Deteção de Overflow com e sem sinal (R = Resultado):**

* + **n = 32... 0...31, é o ultimo bit do operador A, B e do resultado R**