

Circuitos Lógicos

Aula 22

Aula passada

- Armazenamento e transferência
- Paralela x Serial
- Divisão de frequência
- Contador
- Microprocessador

Aula de hoje

- Aritmética binária
- Representação binária com sinal
- Complemento a 2
- Adição e subtração
- Multiplicação

Aritmética Binária

- Idêntica a aritmética na base 10
 - mesmos princípios
- Soma é a operação fundamental
 - CIs modernos usam apenas adição para implementar subtração, multiplicação e divisão
- Aspectos da soma no sistema posicional
 - Base 10: “vai um” quando dígitos somados ultrapassam 9
 - Base 2: “vai um” quando dígitos somados ultrapassam 1
 - Outras bases: mesma coisa!

Soma Binária

■ Exemplos

■ $9 + 5 = 14$

$$9 = 1001$$

$$5 = 0101$$

$$1110 = 14$$

■ $11 + 12 = 23$

$$11 = 1011$$

$$12 = 1100$$

$$10111 = 23$$

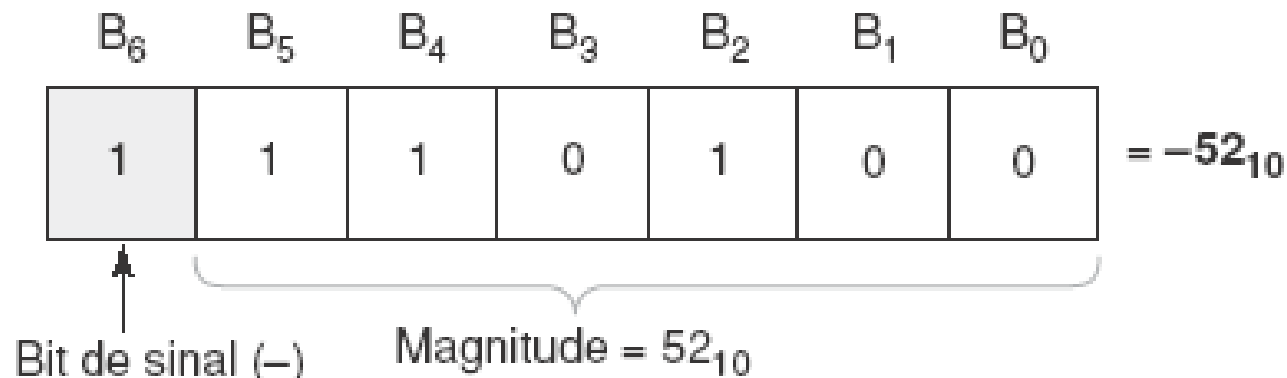
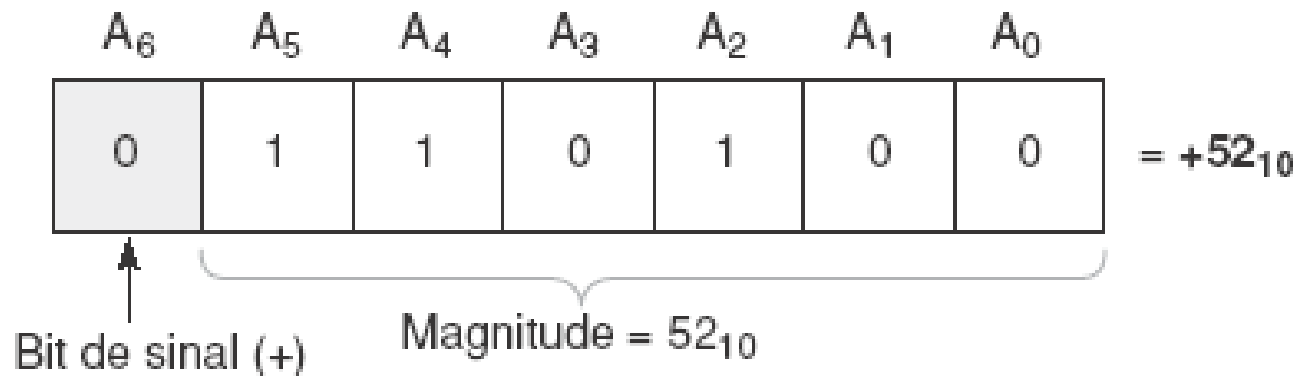
Números Negativos



- Como representar números negativos?
- Usar um bit para representar o sinal do número
 - *bit de sinal*
- Bit de mais alta ordem (MSB)
 - 0 : positivo, 1 negativo
- Outros bits representam magnitude do número

Exemplo

- Número representado com sete bits
- MSB é bit de sinal



Complemento a 2

- Forma de representar números binários
- Mais usada pois é mais fácil de manipular
 - adição e subtração são idênticas
 - facilita construção do circuito
- Complemento a 1
 - Inverter os bits do número (magnitude)
- Complemento a 2
 - Somar 1 ao complemento a 1
- Exemplos

9 → 1001 → 0110 → 0111

4 → 100 → 011 → 100

Sinal e Complemento a 2

- Sistema para representar números com sinal utilizando complemento a 2
 - número fixo e pré-determinado de bits (ex. 8)
- MSB faz papel do bit de sinal
- Se número é positivo: MSB = 0 e número representado em binário (normal)
- Se número é negativo: MSB = 1 e número é representado com complemento a 2
- Exemplos com 6 bits (1 + 5)

9 → 01001 → 001001

-10 → 01010 → 10101 → 10110 → 110110

-4 → 00100 → 11011 → 11100 → 111100

Significado do Complemento a 2

- Considere representação com n bits
- Considere número $x \leq 2^n$
- Complemento a 2 de x é quanto falta para x chegar a 2^{n+1} (ou seja $2^{n+1} - x$)
 - valor representado em binário normal
- Usado para representação de números negativos ($-x$)
 - usar x em módulo para calcular
- Ex: $n=5$, $x=-18$
- $2^{5+1} - 18 = 46 = 101110$

Negação do Número

- Negação do número (multiplicar por -1)
 - trocar de sinal
- Neste sistema, basta calcular o complemento a 2 da atual representação
 - todos os bits, incluindo sinal
- Exemplo

9 → 001001 : -9 → 110111
-10 → 110110 : 10 → 001010
-4 → 111100 : 4 → 000100

Adição

- No sistema de Complemento a 2 com sinal
- Realizar adição binária dos dígitos da representação, incluindo bit de sinal
 - ignorar vai um do bit de sinal
- Resultado for positivo estará representado na forma binária
- Resultado for negativo estará representado na forma complemento a 2
- 4 casos
 - dois números positivos, dois números negativos, positivo maior do que um negativo, negativo maior que positivo

Exemplos

- Representação com 6 bits (1 + 5)
- Ex. $9+5$, $(-10)+(-4)$, $9+(-5)$, $(-10)+9$

$+9 = 001001$	$-10 = 110110$
$+5 = 000101$	$-4 = 111100$
-----	-----
001110	1110010

$+9 = 001001$	$-10 = 110110$
$-5 = 111011$	$+9 = 001001$
-----	-----
1000100	111111

Subtração

- Utilizar a operação de negação e adição
- Subtração: $A - B$
 - negação do número a ser subtraído (B) seguido da adição (com A)
- Mesmo 4 casos da adição
- Exemplo: $9 - 5$

+5 = 000101 : -5 = 111011

9 : 001001

-5 : 111011

- - - - -

1000100

Overflow



- O que ocorre quando resultado da adição não cabe dentro da sua representação?

- Ex: 6 bits (1 + 5), 20 + 15?

20 : 010100

15 : 001111

Bit do sinal
está errado!

100011

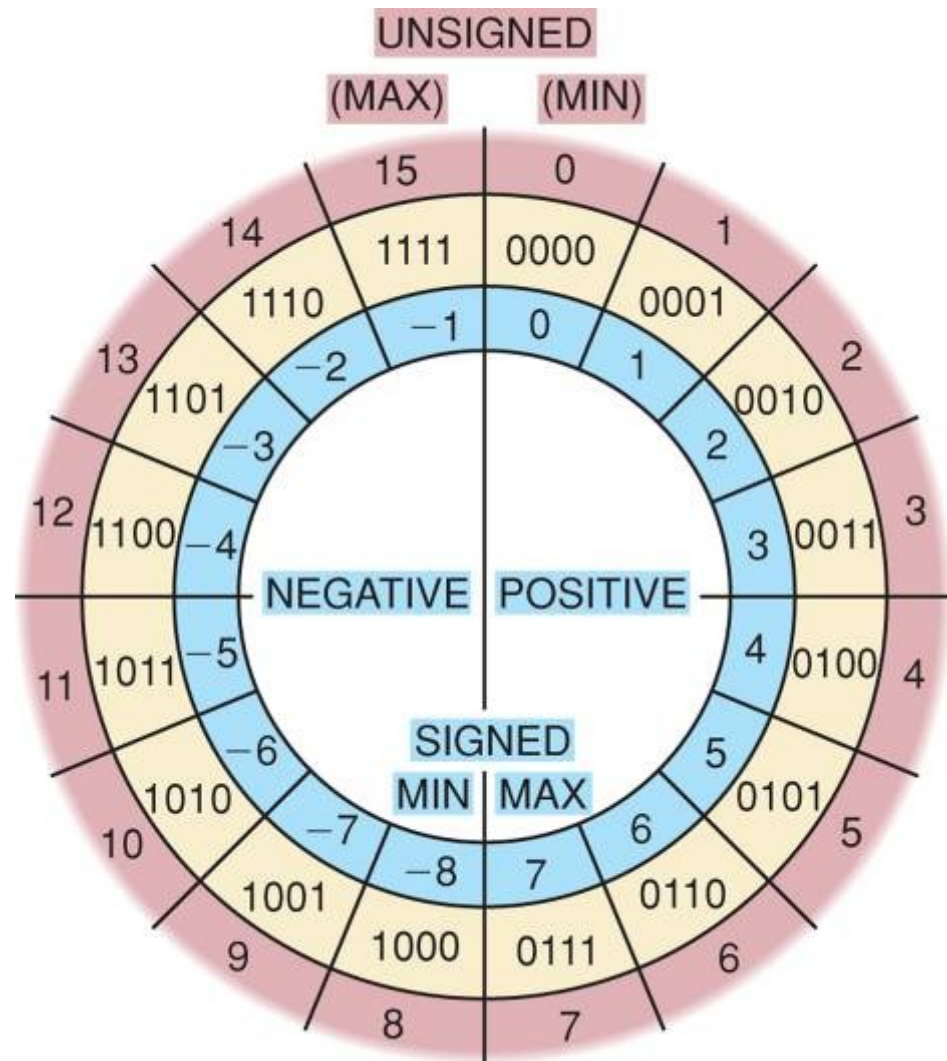
- Overflow: bits da magnitude invadem bit de sinal (e outros)
- Problema numérico fundamental na computação

Lidando com Overflow



- Como detectar o overflow?
- Só ocorre quando somando dois números de mesmo sinal (positivos ou negativos)
- Neste caso, verificar se resultado da soma possui mesmo sinal dos operadores
- Caso negativo, temos overflow (e temos que avisar)

Visualizando Números e Operações



- Adição: andar no sentido horário
 - Número de casas da parcela
 - Ex: $1 + 4$ ou $-3 + 5$
- Subtração: andar no sentido anti-horário
 - Número de casas da parcela
 - Ex. $7 - 3$ ou $-3 - 2$

Multiplicação Binária

- Mesma idéia da multiplicação base 10
- Cada bit do multiplicador multiplica o multiplicando
- Resultado deslocado um bit a cada multiplicação do LSB para o MSB
- Resultados somados para obter resultado
- Ex: $5 * 4 = 20$

```
5 : 0101
4 : 0100
- - - - -
      0000
      0000
      0101
      0000
- - - - -
0010100
```

Multiplicação no Sistema de Complemento a 2

- Converter números para representação positiva
 - usando negação
- Multiplicar cada dígito e somar em acumulador
 - soma apenas dois números de cada vez
- Voltar a representação negativa, caso números tenham sinais trocados