

Introdução à Arquitetura de Computadores

Pedro M. Lavrador

Departamento de Electrónica, Telecomunicações e Informática
Universidade de Aveiro
plavrador@ua.pt

Índice

- Introdução
 - Objetivos e Programa
 - Bibliografia
 - Avaliação
 - Princípios Básicos da Arquitetura de Computadores
- Representação da Informação nos Computadores
 - O sistema binário
 - Representação de números inteiros
 - Conversão entre bases
 - Números negativos:
 - Sinal e módulo
 - Complemento para 2
 - Números reais.
 - Virgula Fixa e Virgula Flutuante
 - O standard IEEE 754
 - Outros Tipos de Dados

Introdução

- Os microprocessadores são o componente que mais contribuiu para a revolução tecnológica do mundo em que vivemos.



13/02/2023

PML – IAC - 2023

3

Introdução

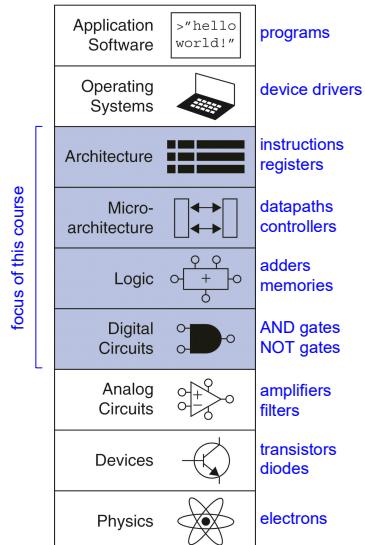
- Hardware ou Software?
 - Não vivem isolados.
 - Um é projetado em função do outro.
- Para ser especialista num dos domínios é preciso ter uma visão das capacidades e limitações do outro.

13/02/2023

PML – IAC - 2023

4

Introdução



13/02/2023

PML – IAC - 2023

5

Objetivos da Disciplina

- **Conhecer as formas de representação da informação nos computadores digitais**, com relevo para a representação da informação numérica e as operações aritméticas básicas.
- **Conhecer as operações lógicas** e as componentes eletrónicas que as realizam.
- **Compreender o funcionamento dos sistemas com memória** e o funcionamento dos principais dispositivos de **armazenamento de informação**.

13/02/2023

PML – IAC - 2023

6

Objetivos da Disciplina

- Compreender a organização interna dos computadores digitais.
- Compreender os mecanismos de comunicação do computador com o exterior.
- Adquirir familiaridade com a arquitetura de processadores através da programação em *assembly*.

13/02/2023

PML – IAC - 2023

7

Programa

- I – Introdução: sistemas de computação de uso geral. Microprocessadores.
- II – Representação da informação e operações básicas
 - Tipos de dados e sua representação
 - O bit como unidade de informação
 - Aritmética Binária
 - Operações lógicas e Álgebra de Boole
- III – Circuitos Lógicos
 - Portas Lógicas: NOT, OR, NOR, AND, NAND
 - Blocos Combinatórios
 - Dispositivos de Armazenamento de Informação. Registos.
 - Memórias.

13/02/2023

PML – IAC - 2023

8

Programa

- IV – O Modelo de Von Neumann
 - Estrutura de um processador
- V – A arquitetura MIPS
 - Modelo de programação; tipos de instruções
- VI – A linguagem *assembly* e o *assembler*
- VII – Organização interna do processador
 - Unidades Operativas e de Controlo.
 - Implementação *Single* ou *Multi Cycle*
- VIII – Comunicação com o exterior: entrada e saída de dados

13/02/2023

PML – IAC - 2023

9

Bibliografia

- D.M.Harris and S.L.Harris, *Digital Design and Computer Architecture*, 2nd. Edition, Morgan Kaufmann, 2013.
- D.A.Patterson, J.Hennessy, *Computer Organization and Design – the hardware/software interface*, Elsevier.
- Y.N. Patt, S.J.Patel, *Introduction to Computing Systems – from bits & gates to C & beyond*, 2nd edition, McGraw_Hill Education, Indian edition.

13/02/2023

PML – IAC - 2023

10

Avaliação

- A Nota da disciplina é obtida pela média ponderada das notas da componente teórica e prática.

$$\text{NotaFinal} = 60\% \times \text{NT} + 40\% \times \text{NP}$$

- A nota final da **Componente Teórica** obtém-se da média ponderada de dois elementos de avaliação:
 - Um teste escrito presencial (T1) com um peso de 50% na nota final, a realizar no dia **31 de Março** de 2023 (à tarde)
 - Um teste escrito presencial (T2) com um peso de 50% na nota final, na época de exames.

$$\text{Nota_Teórica} = \text{Nota_T1} \times 0.5 + \text{Nota_T2} \times 0.5$$

13/02/2023

PML – IAC - 2023

11

Avaliação

- A Nota da disciplina é obtida pela média ponderada das notas da componente teórica e prática.

$$\text{NotaFinal} = 60\% \times \text{NT} + 40\% \times \text{NP}$$

- A nota final da **Componente Prática** obtém-se da média ponderada de três elementos de avaliação:
 - Um teste escrito presencial (TP1) com um peso de 40% na nota final, a realizar nas aulas P de **30 e 31 de Março** (nas aulas P)
 - Um teste escrito presencial (TP2) com um peso de 50% na nota final, a realizar na época de exames.
 - Uma componente de Avaliação do Desempenho nas Aulas Práticas

$$\text{Nota_Prática} = \text{AC} \times 0.1 + \text{P1} \times 0.4 + \text{P2} \times 0.5$$

13/02/2023

PML – IAC - 2023

12

Avaliação

- Em cada uma das duas componentes de avaliação (Teórica-Prática e Prática) é necessário obter pelo menos 7.0 valores para obter aprovação à disciplina.

13/02/2023

PML – IAC - 2023

13

Regime de Faltas

- Todos os estudantes que, não usufruindo do estatuto de trabalhador-estudante no corrente ano letivo, **falem injustificadamente a mais de 20% das aulas práticas reprovam automaticamente à disciplina** ficando impedidos de se apresentar a qualquer prova da mesma durante o corrente ano letivo.
- A justificação de faltas deve ser entregue na secretaria do DETI, no prazo máximo de 10 dias de acordo com o regulamento de Estudos, a qual remeterá cópia para o docente coordenador da disciplina.

13/02/2023

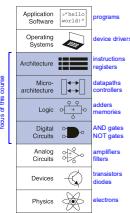
PML – IAC - 2023

14

O professor
acha muito importante!
!

Princípios Básicos da Arquitetura de Computadores

- **Abstração:**
 - Esconder os detalhes sempre que não são necessários;
- **Disciplina:**
 - Restringir intencionalmente as liberdades (por exemplo usando tensões digitais e não contínuas)
- **Hierarquia**
 - Dividir um sistema em módulos e sub-módulos
- **Modularidade**
 - Cada um dos módulos tem interfaces bem definidos
- **Regularidade**
 - Sempre que possível reutilizar os módulos já disponíveis.



13/02/2023

PML – IAC - 2023

15

A abstração Digital

- A maioria das grandezas físicas são contínuas:
 - A temperatura da sala;
 - Uma frequência de oscilação;
 - A tensão num circuito;
 - A posição de um corpo;
- A abstração digital leva-nos a considerar apenas um subconjunto discreto de todos os valores possíveis.
 - Se considerarmos um conjunto suficientemente grande podemos ter aproximações boas. (vídeo e áudio).

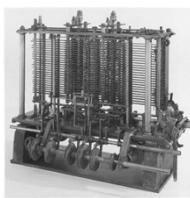
13/02/2023

PML – IAC - 2023

16

Um pouco de história...

- O primeiro computador...
- ... foi projetado por Charles Babbage entre 1834 e 1871.
- Foi o primeiro computador digital, construído com engrenagens mecânicas onde cada “roda” representava um valor discreto entre 0 e 9.
- A máquina consegue calcular 25 dígitos.



13/02/2023

PML – IAC - 2023

17

Um pouco de história...

- Ao contrário da máquina de Babbage, a maioria dos computadores atuais usa uma representação binária.
- A quantidade de informação numa variável discreta com N níveis é medida em número de bits de acordo com a seguinte expressão:

$$D = \log_2 N \quad (\text{bits})$$

Variáveis categóricas
 $N = 100 \Rightarrow D = 4.60$ Térmos de utilização a precisão
 • Um semáforo com R, Y, G ↗
 – 3 níveis precisa de 2 bits ($2^2 = 4$)
 • 7 níveis $\rightarrow 3$ bits 2^3
 • 0-20 $\rightarrow 21$ níveis $\log_2(21) = 4.12\dots \rightarrow 2^5 = 32$

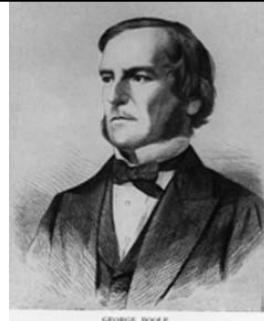
13/02/2023

PML – IAC - 2023

18

Um pouco de história

- George Boole (1815-1864)
 - Filho de pais operários, aprendeu por si matemática.
 - Juntou-se depois ao Queen's College na Irlanda.



GEORGE BOOLE
Scanned at the American
Institute of Physics

- Introduziu as variáveis binárias e os três operadores lógicos fundamentais: AND, OR e NOT
- Foi o precursor da lógica binária na qual se baseiam atualmente os sistemas digitais.

13/02/2023

PML – IAC - 2023

19

Índice

- Introdução
 - Objetivos e Programa
 - Bibliografia
 - Avaliação
- Representação da Informação nos Computadores
 - O sistema binário
 - Representação de números inteiros
 - Conversão entre bases
 - Números negativos:
 - Sinal e módulo
 - Complemento para 2.
 - Números reais.
 - Virgula Fixa e Virgula Flutuante
 - O standard IEEE 754
 - Outros Tipos de Dados

13/02/2023

PML – IAC - 2023

20

Sistemas de Numeração

- O que é um número?
 - Número é uma *coleção de unidades*, Tales de Mileto, sec. VI a.c.
 - Número é a *relação entre a quantidade e a unidade*, Isaac Newton. sec. XVII.
- Importa distinguir o número da sua representação.
 - 12 é sempre “uma dúzia”
 - Quer seja 12, XII, doze, twelve, C₁₆, 14₈, 1100₂, ...

13/02/2023

PML – IAC - 2023

21

Sistemas de Numeração: Decimal

- Sempre trabalhámos com números decimais
 - (porque temos 10 dedos!)
- Em sistemas digitais (com 0's e 1's) o sistema binário ou hexadecimal é mais conveniente.
- No sistema decimal temos 10 dígitos: 0, 1, 2, ..., 9 e o valor de cada um deles depende da posição que ocupa:



13/02/2023

PML – IAC - 2023

22

Sistemas de Numeração: Decimal

- Qual a gama de representação que podemos ter num número decimal com N algarismos?
 - Por exemplo:
 - Com 2 algarismos podemos escrever 100 números diferentes: 0, 1, 2, ..., 99.
 - Com N algarismos podemos escrever 10^N números distintos entre 0 e 10^N-1 .
 - Ao conjunto de números que podemos escrever com N algarismos chamamos **gama de representação**.

13/02/2023

PML – IAC - 2023

23

Sistemas de Numeração: Binário

- Os bits representam dois valores possíveis e podem ser agrupados para formar números binários. (em base 2).
- Um número binário com 4 bits na forma:

$$b_3 b_2 b_1 b_0$$
- Tem o valor:

$$b_3 * 2^3 + b_2 * 2^2 + b_1 * 2^1 + b_0 * 2^0$$

13/02/2023

PML – IAC - 2023

24

Sistemas de Numeração: Binário

- Um número binário geral com N bits na forma:

$$b_{N-1} \dots b_3 b_2 b_1 b_0$$

- Tem o valor:

$$\sum_{n=0}^{N-1} b_n 2^n$$

- A gama de representação de um número binário com N bits é 0, 1, ..., $2^N - 1$.

$$\begin{array}{r} 00 \\ 01 \\ 10 \\ 11 \end{array} \quad 2^2 - 1 = 3 \\ [0, 3]$$

13/02/2023

PML – IAC - 2023

25

Sistemas de Numeração: Binário/Decimal

- Os números decimais:

1000's → milhares
 100's → centenas
 10's → dezenas
 1's → unidades

5 3 7 4₁₀

- Os números binários:

8's → oitavos
 4's → quadrados
 2's → duzentos
 1's → unidades

1 0 1 1₂

13/02/2023

PML – IAC - 2023

26

Sistemas de Numeração: Binário

- Para trabalhar fluentemente em binário é útil conhecer as potências de 2:

$$\begin{aligned}
 2^0 &= 1 \\
 2^1 &= 2 \\
 2^2 &= 4 \\
 2^3 &= 8 \\
 2^4 &= 16 \\
 2^5 &= 32 \\
 2^6 &= 64 \\
 2^7 &= 128 \\
 2^8 &= 256 \\
 2^9 &= 512 \\
 2^{10} &= 1024 \\
 2^{11} &= 2048
 \end{aligned}$$

13/02/2023

PML – IAC - 2023

27

Sistemas de Numeração: Conversão entre bases

- Converter para Decimal:

- Fazer a soma de cada dígito b_n multiplicado pela potência de 2 correspondente:

$$valor = \sum_{n=0}^{N-1} b_n 2^n$$

- Converter para Binário:

- Fazer divisões sucessivas por 2 até obter quociente 0.
- O número em binário é o conjunto dos restos das divisões dispostos pela ordem inversa.

13/02/2023

PML – IAC - 2023

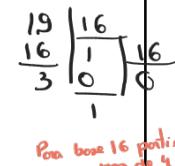
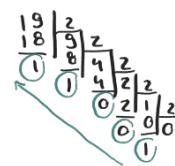
28

Sistemas de Numeração: Conversão entre bases

- Converter para Decimal:

$$10011_2 = 2^4 + 2 + 1 = 16 + 2 + 1 = 19$$

$$01101_2 = 2^3 + 2^2 + 1 = 13$$



$$19_{10} = 13_{16}$$

$$13_{16} = 1 \times 16^1 + 3 \times 16^0 = 16 + 3 = 19$$

- Converter para Binário:

$$47_{10}$$

$$21_{10}$$

13/02/2023

PML – IAC - 2023

29

Sistemas de Numeração: Hexadecimal

- Escrever longas sequências binárias é chato e muito suscetível a erros.
- Um grupo de 4 bits, representa uma de $\underline{2^4=16}$ possibilidades.
- Assim é possível e mais cômodo trabalhar num sistema de base 16, chamado hexadecimal.
- Os algarismos hexadecimais são:
 - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

13/02/2023

PML – IAC - 2023

30

Sistemas de Numeração: Hexadecimal

Hexadecimal	Decimal	Binário
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

13/02/2023

PML - IAC - 2023

31

Ex → ^{8 Atenção que o zero é à esquerda}
 $\underline{0} \underline{111} \underline{1010}_2 = ?A_{16}$
 A

Sistemas de Numeração: Hexadecimal

- Conversão binária para hexadecimal:

$$\begin{array}{l} \underline{0} \underline{111} \underline{1010}_2 = ?A \\ \underline{0} \underline{0111} \underline{1101}_2 = ?D \end{array}$$

- Conversão decimal para hexadecimal e binário:

$$333_{10} = ?_2 = ?_{16}$$

$$\begin{array}{r} 333 \\ 332 \Big| 2 \\ \underline{1} \quad | 166 \\ 166 \Big| 2 \\ \underline{0} \quad | 83 \\ 83 \Big| 2 \\ \underline{1} \quad | 41 \\ 41 \Big| 2 \\ \underline{0} \quad | 20 \\ 20 \Big| 2 \\ \underline{0} \quad | 10 \\ 10 \Big| 2 \\ \underline{0} \quad | 5 \\ 5 \Big| 2 \\ \underline{0} \quad | 2 \\ 2 \Big| 1 \\ \underline{0} \quad | 1 \\ 1 \Big| 0 \\ \underline{0} \quad | 0 \\ 0 \Big| 0 \\ 0 \end{array}$$

$$333_{10} = \underline{10} \underline{100} \underline{110}_2$$

$$= 14 D_{16}$$

13/02/2023

PML - IAC - 2023

32

Sistemas de Numeração: Jargão

- Um **byte** é um conjunto de 8 bits:
 - Pode representar $2^8 = 256$ valores diferentes.
(0 à 255 não é representável [0, 255])
- Um **nibble** é um meio byte (4 bits):
 - Pode representar $2^4 = 16$ valores diferentes.
- Um processador opera sobre conjuntos de bits chamados **palavras** (word):
 - O tamanho da palavra depende da arquitectura.
(Atualmente a maioria dos processadores são de 64 bits o que significa que operam sobre palavras de 64 bits.)

Sistemas de Numeração: Jargão

- Dentro de um grupo de bits chama-se **bit menos significativo** (*lsb*) ao da direita e **bit mais significativo** ao mais à esquerda (*msb*).
- Dentro de uma palavra chama-se **byte menos significativo** (*LSB*) ao da direita e **byte mais significativo** ao mais à esquerda (*MSB*).
- O prefixo grego *kilo*, que designa 10^3 , é reutilizado em binário para designar $2^{10} = 1024$.
 - Mega designa 2^{20} ,
 - Giga 2^{30} ,
 - e assim sucessivamente.

*1 kilobyte $\approx 10^3$
1024 ≈ 1000*

Adição Binária

- A adição binária funciona do mesmo modo que a adição decimal.
 - É até mais simples de efetuar porque a tabela de somas é menor:
 - $0+0 = 0$
 - $0+1 = 1+0 = 1$
 - $1+1 = 0$ e vai 1 para a esquerda (*carry*).

$$\begin{array}{r}
 11 \\
 4277 \\
 +5499 \\
 \hline
 9776_{10}
 \end{array}
 \quad
 \begin{array}{r}
 110 \\
 1011 \\
 +0011 \\
 \hline
 1110_2
 \end{array}$$

← carry →

fica 1 e vai 1

$$\begin{array}{r}
 1 \\
 +1 \\
 \hline
 2_{10}
 \end{array}
 \quad
 \begin{array}{r}
 1 \\
 +1 \\
 \hline
 10_2
 \end{array}
 = 2_{10}$$

$$\begin{array}{r}
 C \\
 +9 \\
 \hline
 15
 \end{array}$$

C 2⁴
 D 2³
 E 2²
 F 10
 11
 12
 13
 14
 15

$+9 = 15_{10}$

13/02/2023

PML – IAC - 2023

35

$$\begin{array}{r}
 1 \\
 0 \\
 + \\
 1 \\
 \hline
 10
 \end{array}$$

Adição Binária

- Calcule a soma binária:

$$\begin{array}{l}
 0110_2 + 0101_2 = \\
 1001_2 + 0011_2 = \\
 1101_2 + 0101_2 =
 \end{array}
 \quad
 \begin{array}{r}
 1101 \\
 1101 \\
 +0101 \\
 \hline
 10010
 \end{array}
 \quad
 \text{Se quisermos em 4 bits, o resultado não pode ser representado} \Rightarrow \text{Overflow}$$

- Quantos bits são necessários para representar o resultado?

- Poderá ocorrer *overflow*?

13/02/2023

PML – IAC - 2023

36

Adição Binária: Overflow

- As máquinas digitais operam com um número fixo de bits.
- Se o resultado da soma exceder o número de bits disponível dizemos que ocorreu *overflow*.
- Por exemplo num sistema de 4 bits $1100_2 + 0110_2$

$$\begin{array}{r}
 \textcolor{red}{11} \\
 1100 \\
 +0110 \\
 \hline
 10010
 \end{array}$$

- Não é representável com 4 bits, logo origina um *overflow*.

13/02/2023

PML – IAC - 2023

37

Índice

- Introdução
 - Objetivos e Programa
 - Bibliografia
 - Avaliação
- Representação da Informação nos Computadores
 - O sistema binário
 - Representação de números inteiros
 - Conversão entre bases
 - Números negativos:
 - Sinal e módulo
 - Complemento para 2.] ⓘ
 - Números reais.
 - Virgula Fixa e Virgula Flutuante
 - O standard IEEE 754
 - Outros Tipos de Dados

13/02/2023

PML – IAC - 2023

38

Representação sinal e módulo

	Menor representado	Maior representado
Sem sinal	0 000	$1111 = 2^4 - 1 = 15$
Sinal e módulo	$1\underline{111} = -7 = -(2^{4-1})$	$0111 = +2^{4-1} - 1 = 7$
Complemento para 2	$1000 = -8 = -2^{4-1}$	$0111 = 7 = 2^{4-1} - 1$

Representação da Informação nos Computadores

Representação de Números com Sinal

- Até aqui apenas vimos a representação de números positivos (ou sem sinal).
- Se quisermos tratar números positivos e/ou negativos precisamos de um sistema de numeração diferente.
- Uma representação em Sinal e Módulo é, para nós, a mais intuitiva.
 - Porque é a que estamos habituados.
- Nesta representação o bit mais significativo representa o sinal e os restantes o módulo.
- Implicações:
 - Uma vez que bit mais significativo representa o sinal é preciso fixar o número de bits

$$\begin{array}{l} 0 \ 111 = +7 = 0\ 000\ 111 \\ 1 \ 111 = -7 = 1\ 000\ 111 \end{array}$$

13/02/2023

PML – IAC - 2023

39

Representação da Informação nos Computadores

Representação de Números com Sinal

- Representação sinal e módulo:
 - Um número com N bits em sinal e módulo tem 1 bit (o da esquerda) para representar o sinal e N-1 bits para representar o módulo.
- Qual a representação de +3 e -3, em sinal e módulo com 4 bits?

$+3 = 0011$	$-3 = 1011$
-------------	-------------
- Qual o valor mínimo e máximo que se pode representar em sinal e módulo com N bits?
 - Máximo = $2^{N-1}-1$ $0111 [+7]$
 - Mínimo = $-(2^{N-1}-1)$ $1111 [-7]$

13/02/2023

PML – IAC - 2023

40

Representação de Números com Sinal

- Representação sinal e módulo:
- Vantagens:
 - Intuitiva
- Desvantagens:
 - Dupla representação para zero (+0 e -0)
 - A realização de somas não é imediata, por exemplo $+3 + (-3)$, não faz sentido (implica cálculos auxiliares)!

13/02/2023

PML – IAC - 2023

41

Complemento para 2:

$$b_3 b_2 b_1 b_0 = \underline{-b_3} \times 2^3 + b_2 \times 2^2 + b_1 \times 2^1 + b_0 \times 2^0$$

$$0(111 = 7) \quad 1111 = -1$$

$$1000 = -8$$

	Menor representado	Maior representado
Sem sinal	0000	$1111 = 2^4 - 1 = 15$
Sinal e módulo	$1111 = -7 = -(2^3 - 1)$	$0111 = +2^{4-1} - 1 = 7$
Complemento para 2	$1000 = -8 = 2^{4-1}$	$0111 = 7 = 2^{4-1} - 1$

Representação da Informação nos Computadores

Representação de Números com Sinal

O que realmente
o computador
utiliza

- Representação em complemento para 2:
- A representação de um número em complemento para 2 é idêntica à representação de números sem sinal, exceto que o bit mais significativo em vez de valer 2^{N-1} vale -2^{N-1} .
- Por exemplo, em complemento para 2, com 4 bits:
 - Zero escreve-se: 0000 $0 * (-2^3) + 0 * 2^2 + 0 * 2^1 + 0 * 2^0$
 - O mais positivo é: 0111 $0 * (-2^3) + 1 * 2^2 + 1 * 2^1 + 1 * 2^0$
 - O mais negativo é: 1000 $1 * (-2^3) + 0 * 2^2 + 0 * 2^1 + 0 * 2^0$
 - -1 escreve-se: 1111 $1 * (-2^3) + 1 * 2^2 + 1 * 2^1 + 1 * 2^0$

13/02/2023

PML – IAC - 2023

42

Representação de Números com Sinal

- Representação em complemento para 2:
- Em complemento para 2, com N bits:
 - O mais positivo é $2^{N-1}-1$
 - O mais negativo é -2^{N-1}
- O bit mais significativo em complemento para dois pode ser visto como o sinal do número:
 - Os números positivos em complemento para dois têm o bit mais significativo 0 e os negativos têm bit mais significativo 1.

13/02/2023

PML – IAC - 2023

43

Representação de Números com Sinal

- Representação em complemento para 2:
- O valor de um número negativo em complemento para 2 pode ser obtido num processo que chamamos **calcular o complemento para 2** que consiste em inverter todos os bits e somar 1.
 - A representação em complemento para 2 com 4 bits de -3?
 - $3_{10} = 0011_2$
 - O complemento para 2 de 0011_2 é:

$$\begin{array}{r} 1100_2 \\ + 1 \end{array}$$
 - Logo -3 em complemento para 2 com 4 bits escreve-se:

$$1101_2$$

13/02/2023

PML – IAC - 2023

44

Representação de Números com Sinal

- Representação em complemento para 2:
- Qual o valor de 1001_2 assumindo que está representado em complemento para 2 com 4 bits?
 - O bit mais significativo é 1, logo o número é negativo.
 - Então precisamos de calcular o complemento para 2, negando todos os bits e somando 1:

$$0110_2 + 1 = 0111_2 \\ = 7$$

Concluímos que 1001_2 representa o -7.

Para extender os números em complemento para 2:

$$\begin{array}{r} 0000 \quad | 0111 = 0111 = 7 \\ 1111 \quad | 1000 = 1000 = -8 \end{array} \quad [-8, 7]$$

↓
replicamos RSB

13/02/2023

PML - IAC - 2023

0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	-8
1001	-7
1010	-6
1011	-5
1100	-4
1101	-3
1110	-2
1111	-1

45

Complemento para 2

$$\begin{array}{r} 0100 \\ -1100 \end{array}$$

$$-2$$

Representação de Números com Sinal

- Representação em complemento para 2:
- A vantagem **esmagadora** do complemento para 2 é que as somas funcionam de modo transparente quer se trate de números positivos quer sejam negativos.
- Exemplo $+2_{10} + (-1_{10})$ (com 4 bits)
 - $-2_{10} = 0010_2$
 - $-1_{10} = 0001$ em complemento para 2: $1110_2 + 1 = 1111_2$

Importante:

Desaparece porque estamos em 4 bits

$$\begin{array}{r} 1110 \\ 0010 \\ +1111 \\ \hline X0001 \end{array}$$

13/02/2023

PML - IAC - 2023

46

Para passar de 2 para (-2) temos de trocar os bits e somar 1

$$\begin{array}{r} 0010 \rightarrow 2 \\ 1101 \downarrow \times(-1) \\ 1110 \hline +1 \\ 0001 \end{array} \rightarrow -2$$

Representação de Números com Sinal

- Representação em complemento para 2:
- Exemplo $+5_{10} + (-3_{10})$ (com 4 bits)
 - $5_{10} = 0101_2$
 - $-3_{10} = 1101_2$
 - $3_{10} = 0011$ em complemento para 2: $1100_2 + 1 = 1101_2$

$$\begin{array}{r}
 1101 \\
 0101 \\
 +1101 \\
 \hline
 X0010
 \end{array}$$

13/02/2023

PML – IAC - 2023

47

Representação de Números com Sinal

- Representação em complemento para 2:
- Exemplo $+3_{10} + (-5_{10})$ (com 4 bits)
 - $3_{10} = 0011_2$
 - $-5_{10} = 1011_2$
 - $5_{10} = 0101$ em complemento para 2: $1010_2 + 1 = 1011_2$

$$\begin{array}{r}
 011 \\
 0011 \\
 +1011 \\
 \hline
 1110
 \end{array}$$

- 1110_2 em complemento para 2 é $0001 + 1 = 0010 = 2$
- $1110_2 = -2$

13/02/2023

PML – IAC - 2023

48

Representação de Números com Sinal

- Representação em complemento para 2:

- Deteção de Overflow

- Exemplo $+3_{10} + (5_{10})$ (com 4 bits)

– $3_{10} = 0011_2$

– $5_{10} = 0101_2$

$$\begin{array}{r}
 111 \\
 0011 \\
 +0101 \\
 \hline
 1000
 \end{array}$$

- A soma de dois números positivos deu um número negativo!

- Ocorreu overflow!

13/02/2023

PML – IAC - 2023

49

Representação de Números com Sinal

- Representação em complemento para 2:

- Deteção de Overflow:

- Um **overflow** ocorre se a **soma de dois números positivos** for maior que o máximo positivo representável ($2^{N-1}-1$), ou seja **originar um número negativo**.

- Ou se a **soma de dois números negativos** for menor que o menor número representável (-2^{N-1}), ou seja **originar um número positivo**.

- Nunca ocorre overflow quando somamos números de sinais diferentes. (Porquê?)

- A existência de um carry out no bit mais significativo não é indicadora de overflow.

13/02/2023

PML – IAC - 2023

50

Representação de Números com Sinal

- Extensão do número de bits.
- Podemos necessitar de mudar o número de bits usado na representação de um número.
- Em números sem sinal:
 - Basta acrescentar zeros à esquerda.
- Em sinal e módulo:
 - Mudamos o sinal para a esquerda e acrescentar zeros no módulo.
- Em complemento para 2:
 - Acrescenta-se o sinal à esquerda:
 - 0's se o número é positivo
 - 1's se o número é negativo

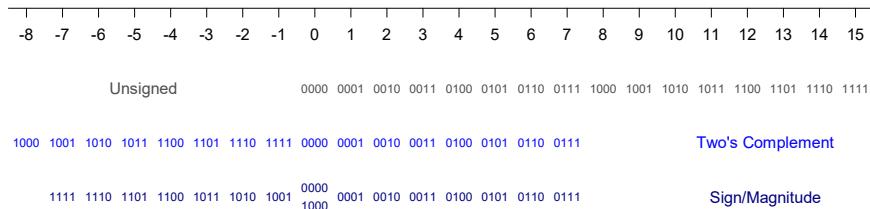
13/02/2023

PML – IAC - 2023

51

Comparação de Representação de Números

Sistema Numeração	Gama
Sem sinal	$[0, 2^N-1]$
Sinal e Módulo	$[-(2^{N-1}-1), 2^{N-1}-1]$
Complemento para 2	$[-2^{N-1}, 2^{N-1}-1]$



13/02/2023

PML – IAC - 2023

52

Índice

- Introdução
 - Objetivos e Programa
 - Bibliografia
 - Avaliação
- Representação da Informação nos Computadores
 - O sistema binário
 - Representação de números inteiros
 - Conversão entre bases
 - Números negativos:
 - Sinal e módulo
 - Complemento para 2.
 - Números reais.
 - Virgula Fixa e Virgula Flutuante
 - O standard IEEE 754
 - Outros tipos de Dados

13/02/2023

PML – IAC - 2023

53

Representação da Informação nos Computadores

Representação de Números Reais

- Até aqui vimos como representar quantidades inteiras (positivas e/ou negativas).
- Podemos generalizar a representação de inteiros na base 2, para representar números reais usando a vírgula, à semelhança do que fazemos em base 10:

$$\text{Parte_Inteira}, \text{Parte_Fracionária}$$

$$x = b_3 b_2 b_1 b_0, b_{-1} b_{-2} b_{-3}$$

$$x = b_3 2^3 + b_2 2^2 + b_1 2^1 + b_0 2^0 + b_{-1} 2^{-1} + b_{-2} 2^{-2}$$

13/02/2023

PML – IAC - 2023

54

<u>Base 10</u>	
$\begin{array}{cccccc} 2 & 1 & 0 & -1 & -2 \\ \hline 1 & 2 & 3 & , & 4 & 5 \end{array}$	
$6,02 \times 10^{23}$	
$-1,6 \times 10^{-19}$	
$0,123 \times 10^4 = 1,23 \times 10^3$	

<u>Base 2</u>	
	$b_3 \ b_2 \ b_1 \ b_0 \ b_{-1} \ b_{-2}$
	$b_3 \times 2^3 + b_2 \times 2^2 + b_1 \times 2 + b_0 + b_{-1} \times 2^{-1} + b_{-2} \times 2^{-2}$

Soma de representação (sem sinal)

Numerismos

$$O \ a \ 10^n - 1$$

$$O \ a \ 2^n - 1$$

Erro de representação

Up relaciona-se Distância mínima entre dois pontos consecutivos com

Representação de Números Reais

- Virgula Fixa: Gama e Erro de Representação
- O Maior (e menor) valor representável é função do número de algarismos da parte inteira
- O erro de representação (absoluto) é função do número de algarismos da parte fracionária:
 - O erro é metade do peso do último algarismo. (Porquê?)

Máximo

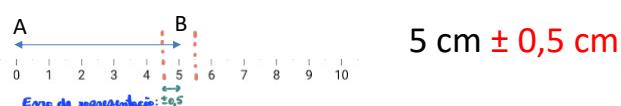
13/02/2023

PML – IAC - 2023

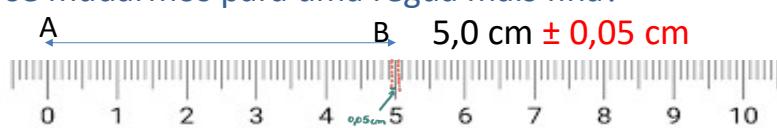
55

O erro de representação

- Qual o comprimento do segmento AB?



- E se mudarmos para uma régua mais fina?



- O erro máximo é sempre metade do valor do último dígito representado!

↳ O erro de representação difere com a base

13/02/2023

PML – IAC - 2023

56

$$\begin{array}{c} 123,45 \quad \text{ou } b_2 b_1 b_0, b_{-1}, b_{-2} \\ \downarrow \qquad \qquad \qquad \downarrow \\ 10^2 = 0,01 \qquad \qquad \qquad 10^{-2} = 0,005 \end{array}$$

$$< \qquad \qquad \qquad <$$

$$10^3 = 0,12$$

Gama e Erro de representação

- Exemplo:

- Representação decimal, com sinal mais 3 dígitos na parte inteira e 3 dígitos na parte decimal.

- Gama de representação = $\{ -999,999 \dots 999,999 \}$

- Erro de representação = ↴

$$\frac{1}{2} \times 10^{-3} = 5 \times 10^{-4} = 0,0005$$

13/02/2023

PML – IAC - 2023

57

Representação da Informação nos Computadores

Representação de Números Reais

- Virgula Fixa Binária
- A vírgula no sistema binário funciona do mesmo modo que no sistema decimal.
 - A função da vírgula é identificar as potências de 2 positivas e as potências de 2 negativas.
- Desde que a soma (subtração) seja feita com as vírgulas alinhadas **somar números inteiros ou fracionários em complemento para 2 é idêntico.**

13/02/2023

PML – IAC - 2023

58

Representação de Números Reais

- Virgula Fixa Binária
- Exemplo:
 - Representar com 8 bits, cinco inteiros e três fracionários os números:
 - 10.25_{10}
 - 4.625_{10}
 - 2.5_{10}
 - Efetuar em complemento para 2 as operações:
 - $10.25_{10} + 4.625_{10}$
 - $4.625_{10} + 2.5_{10}$

Representação de Números Reais

- Virgula Fixa Binária: Problemas.
- Representar valores muito grandes (ou muito pequenos) em virgula fixa requer uma enorme quantidade de algarismos.
- Em decimal a solução encontrada foi a notação científica:
 - $6.02 \cdot 10^{23}$
 - $1.6 \cdot 10^{-19}$
- Isto é **Virgula Flutuante**

$$10,25_{10} = 01010,010_2$$

$$\begin{array}{r} 10 \mid 2 \\ 10 \mid 5 \mid 2 \\ \hline 0 \quad 4 \quad 2 \quad 2 \\ \hline 1 \quad 2 \quad 1 \quad 2 \\ 0 \quad 0 \quad 0 \quad 0 \\ \hline 1 \end{array}$$

$$4,625_{10} = 00100,101_2$$

$$2,5_{10} = 00010,100_2$$

$$0,25 \times 2 = 0,5$$

$0,5 \times 2 = 1,0$ → retira-se sempre a parte inteira
 $0,0 \times 2 = 0,0$ (neste caso acaba mos)
 podia não acabar

$$10,25_{10} + 4,625_{10} = \begin{array}{r} 01010,010 \\ + 00100,101 \\ \hline 01110,111 \end{array}$$

$$10,25_{10} + (-4,625) =$$

$$\begin{array}{r} 01010,010 \\ + 11011,011 \\ \hline 00101,101 \end{array}$$

→ Não é representável em binário
 com poucos zeros

$$0,1_{10} = 00000,0001100110011\dots$$

$$00100\ 101$$

$$\begin{array}{r} 11011,010 \\ + 1 \\ \hline 11011,011 \end{array}$$

→ Complemento para 2
 funciona com frações.

dígima
 infinita

$$\left. \begin{array}{l} 0,1 \times 2 = 0,2 \\ 0,2 \times 2 = 0,4 \\ 0,4 \times 2 = 0,8 \\ 0,8 \times 2 = 1,6 \\ 0,6 \times 2 = 1,2 \\ 0,2 \times 2 = 0,4 \end{array} \right]$$

Só pediram 3 zeros
 decimais

Representação de Números Reais

- Virgula Flutuante Binária:

$$X = \pm 1, m * 2^{\text{exp}}$$

— m: mantissa
conjunto de números finitos

— exp: expoente
- A mantissa determina precisão.
→ medida da distância entre 2 pontos consecutivos
- O expoente determina a gama de representação.
- A representação em Virgula Flutuante expressa um compromisso entre gama de representação e precisão.

Na notação científica em decimal:

$$x = \pm x_{\text{mantissa}} \times 10^{\text{exp}}$$

A mesma coisa

13/02/2023

PML – IAC - 2023

61

O Formato IEEE 754

- Especifica o modo de representar números binários em virgula flutuante com:
 - Precisão simples: 32 bits
 - Precisão dupla: 64 bits
- As regras da soma, subtração, multiplicação e divisão, raiz quadrada, resto e comparação.
- As conversões entre formatos.
- Os modos de arredondamento.
- As exceções.
 - A representação de 0, $\pm\infty$, NaN, e valores desnormalizados.

13/02/2023

PML – IAC - 2023

62

O Formato IEEE 754

Simples 8 bits
Dupla 11 bits

23 bits = 32
52 bits = 64



caso especial

Se EXP = MANTISSA = 0
o número é 0

$$X = (-1)^S * 1, MANTISSA * 2^{(EXP-BIAS)}$$

- S: sinal (0: positivo $(-1)^0 = 1 > 0$ 1: negativo $(-1)^1 = -1 < 0$)
- EXP: expoente representado em excesso do Bias.
– Bias = 127 em Single, 1023 em Double,
- MANTISSA tem um bit escondido pois o algarismo à esquerda da vírgula é sempre 1.

13/02/2023

PML – IAC - 2023

63

O Formato IEEE 754

- Exemplo 1:
- Que número está representado em formato IEEE 754 precisão simples como:

1011 1111 0100 0000 0000 0000 0000 (0xBF400000)

- Solução, determinar:

- Sinal: 1
- Expoente: 011 1111 0 = 126
- Mantissa: 1,100 0000 0000 0000 0000

– Escrever o número como no slide anterior.

$$X = (-1)^1 * 1,100 * 2^{(126-127)} = -1,1_2 * 2^{-1} = -0,75$$

sai nos testes

✓
0

13/02/2023

PML – IAC - 2023

64

O Formato IEEE 754

- Exemplo 2:
- Determinar a representação em formato IEEE 754 precisão simples do número: 5.5

$$\begin{aligned}
 5.5_{10} &= 101.10000\dots \\
 &= 1,0110 \times 2^2 = (-1)^0 \times 1,0110 \times 2^{129}
 \end{aligned}$$

• Solução:

- Escrever o número na forma canónica.
- Agrupar cada um dos campos nos 32 bits.

Sinal = 0
*Exp = 129 → 1000 0001
 0110 00...*

$$X = (-1)^S * 1, MANTISSA * 2^{(EXP-BIAS)}$$

0100 0000 1011 0000 0000 0000 0000 0000, (0x40B0 0000)

129₁₀

Mantissa

13/02/2023

PML – IAC - 2023

65

O Formato IEEE 754

- Casos Especiais:

Número	Sinal	Expoente	Mantissa
0	X	00000000	0000000000000000000000000000
∞	0	11111111	0000000000000000000000000000
$-\infty$	1	11111111	0000000000000000000000000000
NaN	X	11111111	Non-zero

13/02/2023

PML – IAC - 2023

66

O Formato IEEE 754

- Adição:
 1. Extrair os bits do expoente e da mantissa.
 2. Acrescentar 1 à esquerda (o bit escondido) para obter a mantissa completa.
 3. Comparar os expoentes
 4. Deslocar à direita a mantissa do número menor para alinhar os expoentes.
 5. Somar as mantissas e ajustar o expoente se necessário.
 6. Arredondar o resultado
 7. Juntar expoente e fração no formato do standard

13/02/2023

PML – IAC - 2023

67

O Formato IEEE 754

- Adição exemplo:
- Somar os números seguintes representados na norma IEEE 754.

$$0x3FC00000 + 0x40500000$$

- Resolução:
- Extrair os bits do exp e mant:
 $1,1_2 * 2^0 + 1,101_2 * 2^1$
- Alinhar os expoentes:
 $0,11_2 * 2^1 + 1,101_2 * 2^1$
- Solução (0x4098000)

13/02/2023

PML – IAC - 2023

68

Índice

- Introdução
 - Objetivos e Programa
 - Bibliografia
 - Avaliação
- Representação da Informação nos Computadores
 - O sistema binário
 - Representação de números inteiros
 - Conversão entre bases
 - Números negativos:
 - Sinal e módulo
 - Complemento para 2.
 - Números reais.
 - Virgula Fixa e Virgula Flutuante
 - O standard IEEE 754
 - Outros Tipos de Dados

13/02/2023

PML – IAC - 2023

69

Representação da Informação nos Computadores

Representação de Caracteres

- O código ASCII (American Standard Code for Information Interchange)
- É um código binário que codifica 128 sinais:
 - 95 Gráficos: letras do alfabeto, caracteres de pontuação e símbolos matemáticos
 - 33 Controlo: Esc, Del, mudanças de linhas, tabulações, etc.
- Podemos explorar propriedades da tabela ASCII:
 - Para determinar a ordem alfabética.
 - Para determinar a representação ASCII de um algarismo.
 - Para converter maiúsculas em minúsculas.

13/02/2023

PML – IAC - 2023

70

Representação de Caracteres: ASCII

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	@	96	60	"
1	1	Start of heading	SOH	CTRL-A	33	21	!	65	41	A	97	61	a
2	2	Start of text	STX	CTRL-B	34	22	"	66	42	B	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	c
4	4	End of xmit	EOT	CTRL-D	36	24	\$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	e
6	6	Acknowledge	ACK	CTRL-F	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	CTRL-H	40	28	(72	48	H	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29)	73	49	I	105	69	i
10	0A	Line feed	LF	CTRL-J	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	O	111	6F	o
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	p
17	11	Device control 1	D1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	D2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	D3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	D4	CTRL-T	52	34	4	84	54	T	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	v
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	x
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	y
26	1A	Substitute	SUB	CTRL-Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[59	3B	;	91	5B	[123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	GS	CTRL-]	61	3D	=	93	5D]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL-_	63	3F	?	95	5F	_	127	7F	DEL

13/02/2023

PML – IAC - 2023

71

Representação de Caracteres

- O código ASCII apenas permite representar o alfabeto latino.
- Para representar outros alfabetos (chinês, cirílico, árabe, etc.) são necessários mais símbolos.
 - Logo mais bits para cada caractere.
- O standard UNICODE representa os caracteres em 8, 16 ou 32 bits. Sendo que a representação de 8 bits (UTF-8) é compatível com o código ASCII.

13/02/2023

PML – IAC - 2023

72

Outros tipos de dados

- **Imagens**
 - São tratadas como arrays de pixels
 - Monocromáticas: 1 bit define se é preto ou branco
 - Cor:
 - sistema RGB com 8 bits para a intensidade de cada cor
 - Sistema HSV, etc.
- **Som**
 - O som é um sequência de números armazenados que representam a amplitude sonora ao longo do tempo.

13/02/2023

PML – IAC - 2023

73

Sumário

- Nos computadores os números são representados em **base 2**.
- Para números inteiros positivos é possível com N bits armazenar os valores entre 0 e 2^N-1
- Inteiros com sinal representados em complemento para 2, com N bits podemos representar entre -2^{N-1} e $2^{N-1}-1$
- Os números reais são representados em vírgula flutuante (sinal, expoente e mantissa) em 32 ou 64 bits: norma IEEE 754.
- Caracteres são representados em ASCII (1 byte) ou UNICODE.

13/02/2023

PML – IAC - 2023

74

Próxima Aula...

- Lógica binária. As operações lógicas básicas.
- Álgebra de Boole.