Curso Superior de Bacharelado em Engenharia de Computação

Introdução à Engenharia de Computação

Docente: Me. Vladimir Píccolo Barcelos

INSTITUTO FEDERAL
Mato Grosso do Sul

Semana 4
História dos computadores

INSTITUTO FEDERAL
Mato Grosso do Sul

Sumário

- Introdução à Organização de Computadores
- Evolução dos computadores e suas arquiteturas
- Conhecer o processo de fabricação de um processador

INSTITUTO FEDERAL
Mato Grosso do Sul

Introdução

- O que é um computador?
- Um computador necessariamente necessita ser um equipamento energizado?
-

INSTITUTO FEDERAL
Mato Grosso do Sul

Introdução

- **O que é um computador digital?** → Uma máquina com circuitos eletrônicos que pode resolver problemas para as pessoas executando uma **sequência de instruções** que lhe são dadas.
- **Sequência de Instruções** → descreve como uma tarefa deve ser realizada (conhecemos como **algoritmo** ou **programa**)
- Os circuitos eletrônicos de um computador podem reconhecer e executar diretamente um **conjunto limitado de instruções simples**
- Para que um programa de computador possa ser executado, deve ser convertidos em instruções de modo que as máquinas possam entender

INSTITUTO FEDERAL
Mato Grosso do Sul

Introdução

- **Computadores de propósito específico:** desempenham uma tarefa única, ou seja, fica limitado apenas ao propósito de sua construção.
  - Por exemplo: micro-ondas, SmartTV, máquina de lavar
- **Computadores de propósito geral:** podem ser utilizado para desempenhar diversas tarefas → o programador define quais as funções deste computador.
  - Por exemplo: notebooks e computadores pessoais

As tarefas são executadas por meio de instruções

INSTITUTO FEDERAL
Mato Grosso do Sul

Introdução

- Mas o que seriam essas instruções de computador? As instruções básicas de um computador são como:
  - Some dois números
  - Verifique um número para ver se ele é zero
  - Copie dados de uma parte da memória do computador para outra
- Então todos os programas são convertidos para instruções parecidas como as dos exemplos acima
- Juntas, as linguagens primitivas de um computador formam uma linguagem com a qual as pessoas podem se comunicar com ele, é denominada **linguagem de máquina**.

INSTITUTO FEDERAL
Mato Grosso do Sul

Linguagem de máquina

- Projetistas de computadores devem decidir quais instruções incluir em uma linguagem de máquina. Essas instruções devem ser:
  - as mais simples possíveis
  - coerentes com os requisitos de utilização e desempenho idealizados para o computador
- Como a maioria das linguagens de máquina é bem simples, todos acham difícil e entediante usá-la. Isto permitiu estruturar computadores como uma série de abstrações, cada uma acumulando-se àquela que lhe precede. Isso permitiu projetar sistemas de computador de forma estruturada e sistemática.

INSTITUTO FEDERAL
Mato Grosso do Sul

Organização Estruturada de Computadores

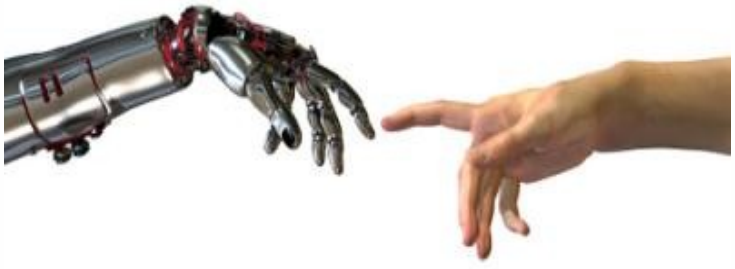
- Há uma lacuna entre o que é conveniente para as pessoas e o que é conveniente para os computadores
  - Por exemplo: uma pessoa quer realizar uma multiplicação mas computadores só podem fazer somas → isso gera problemas

INSTITUTO FEDERAL
Mato Grosso do Sul

Execução de programas

- Programas precisam estar em linguagem de máquina para que possam ser executados
- Os humanos falam idiomas de alto nível, compostos por palavras, normalmente formadas por caracteres do alfabeto

Quais ferramentas permitem criar programas que um computador possa entender?



INSTITUTO FEDERAL
Mato Grosso do Sul

Linguagens de programação

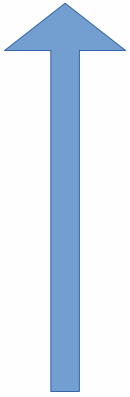
São ferramentas para se escrever códigos de computador em uma linguagem mais próxima da linguagem de máquina

- Classificação das linguagens de programação:
  - Alto nível: são mais parecidas com a linguagem humana (Java, C)
  - Baixo nível: são mais parecidas com a linguagem de máquina (Assembly)

INSTITUTO FEDERAL
Mato Grosso do Sul

Linguagens de programação

Linguagem de programação de alto nível
(mais próxima da linguagem humana)



Linguagem de programação de baixo nível
(mais próxima da linguagem de máquina)

INSTITUTO FEDERAL
Mato Grosso do Sul

Exemplo

Linguagem de máquina:

0010000100000100
0001000100000101
0011000100000110
0111000000000001
0000000001010011
1111111111111110
0000000000000000

INSTITUTO FEDERAL
Mato Grosso do Sul

Exemplo

Linguagem Assembly:

```
.model small
.data
 oprl dw 1234h
 opr2 dw 0002h
 result dw 01 dup(?),'$'
.code
        mov ax,@data
        mov ds,ax
        mov ax,oprl
        mov bx,opr2
        clc
        add ax,bx
        mov di,offset result
        mov [di], ax

        mov ah,09h
        mov dx,offset result
        int 21h

        mov ah,4ch
        int 21h
        end
```

Linguagem de máquina:

0010000100000100
0001000100000101
0011000100000110
0111000000000001
0000000001010011
1111111111111110
0000000000000000

INSTITUTO FEDERAL
Mato Grosso do Sul

Exemplo

Linguagem Assembly:

```
.model small
.data
 opr1 dw 1234h
 opr2 dw 0002h
 result dw 01 dup(?),'$'
.code
        mov ax,@data
        mov ds,ax
        mov ax,opr1
        mov bx,opr2
        clc
        add ax,bx
        mov di,offset result
        mov [di], ax

        mov ah,09h
        mov dx,offset result
        int 21h

        mov ah,4ch
        int 21h
        end
```

Linguagem Pascal:

```
1   Program Soma_numeros;
2
3   var a,b,soma: integer;
4
5   begin
6    a := 1;
7    b := 2;
8    soma:=a+b;
9   End.
10
```

Linguagem de máquina:

0010000100000100
0001000100000101
0011000100000110
0111000000000001
0000000001010011
1111111111111110
0000000000000000

INSTITUTO FEDERAL
Mato Grosso do Sul

Organização Estruturada de Computadores

- Mas se o computador entende uma linguagem e as pessoas preferem usar outra, como é possível programar computadores?
  - Por exemplo: um programa é escrito em uma linguagem L1 pelo programador e é executado pelo computador que só entende a linguagem de máquina L0.

INSTITUTO FEDERAL
Mato Grosso do Sul

Organização Estruturada de Computadores

- Uma técnica usada é substituir cada instrução escrita no programa L1 por uma sequência equivalente de instrução em L0. Esta técnica é denominada de **Tradução** e o programa que faz isso é chamado de **Compilador**.
- Outra técnica é escrever um programa em L0 que considere programas em L1 como dados de entrada e os execute examinando cada instrução por vez e executando diretamente a sequência equivalente de instruções L0. Essa técnica não requer a geração prévia de um novo programa em L0. Esta técnica é denominada **Interpretação** e o programa que a executa é chamado **Interpretador**.

INSTITUTO FEDERAL
Mato Grosso do Sul

Organização Estruturada de Computadores

- Mas se um computador entende a linguagem L0 e nós programamos usando a linguagem L1 por que não se cria um computador capaz de entender a linguagem L1 diretamente?

INSTITUTO FEDERAL
Mato Grosso do Sul

Organização Estruturada de Computadores

- Mas se um computador entende a linguagem L0 e nós programamos usando a linguagem L1 por que não se cria um computador capaz de entender a linguagem L1 diretamente?
  - Por conta do custo e complexidade. Apesar de atualmente sermos capazes de construir uma máquina que entenda Java ou C++ diretamente, seu custo não seria eficiente, ou seja, é um projeto inviável.

INSTITUTO FEDERAL
Mato Grosso do Sul

```
 1            00000000 00000100 0000000000000000
 2  01011110  00001100 11000010 0000000000000010
 3            11101111 00010110 0000000000000101
 4            11101111 10011110 0000000000001011
 5  11111000  10101101 11011111 0000000000010010
 6            01100010 11011111 0000000000010101
 7  11101111  00000010 11111011 0000000000010111
 8  11110100  10101101 11011111 0000000000011110
 9  00000011  10100010 11011111 0000000000100001
10  11101111  00000010 11111011 0000000000100100
11  01111110  11110100 10101101
12  11111000  10101110 11000101 0000000000101011
13  00000110  10100010 11111011 0000000000110001
14  11101111  00000010 11111011 0000000000110100
15            01010000 11010100 0000000000111011
16                     00000100 0000000000111101
```

Exemplo de Linguagem L0

```
00000000                push    ebp
00000001                mov     ebp, esp
00000003                movzx   ecx, [ebp+arg_0]
00000007                pop     ebp
00000008                movzx   dx, cl
0000000C                lea     eax, [edx+edx]
0000000F                add     eax, edx
00000011                shl     eax, 2
00000014                add     eax, edx
00000016                shr     eax, 8
00000019                sub     cl, al
0000001B                shr     cl, 1
0000001D                add     al, cl
0000001F                shr     al, 5
00000022                movzx   eax, al
00000025                retn
```

Exemplo de Linguagem L1

INSTITUTO FEDERAL
Mato Grosso do Sul

Organização Estruturada de Computadores

- Para que o processo de tradução e interpretação de instruções seja viável, as linguagens L0 e L1 não devem ser muito diferentes entre si.
- Embora L1 seja “melhor” para os humanos do que L0, a linguagem L1 ainda está longe de ser ideal para a maioria das aplicações.
  - Como lidar?

INSTITUTO FEDERAL
Mato Grosso do Sul

Organização Estruturada de Computadores

- Para que o processo de tradução e interpretação de instruções seja viável, as linguagens L0 e L1 não devem ser muito diferentes entre si.
- Embora L1 seja “melhor” para os humanos do que L0, a linguagem L1 ainda está longe de ser ideal para a maioria das aplicações.
  - Como lidar? → Cria-se um conjunto de instruções mais dirigido a pessoas e menos a máquinas do que L1. Esse terceiro conjunto forma uma linguagem que denominaremos de L2.
- Assim um programa escrito em L2 pode ser traduzido para L1 que por consequência passível de conversão para L0.

INSTITUTO FEDERAL
Mato Grosso do Sul

Evolução dos computadores

- Os computadores sempre existiram do jeito que os conhecemos hoje?
- Sob sua perspectiva, o que seria um computador antigo na sua visão?
- Você conhece algum computador antigo?

INSTITUTO FEDERAL
Mato Grosso do Sul

Marcos da arquitetura de computadores

- Geração zero – computadores mecânicos (1642 – 1945)
- Primeira geração – válvulas (1945-1955)
- Segunda geração – transistores (1955-1965)
- Terceira geração – Circuitos Integrados (CI) (1965-1980)
- Quarta geração – Integração em escala muito grande (1980-?)
- Quinta geração – computadores invisíveis e computação ubíqua

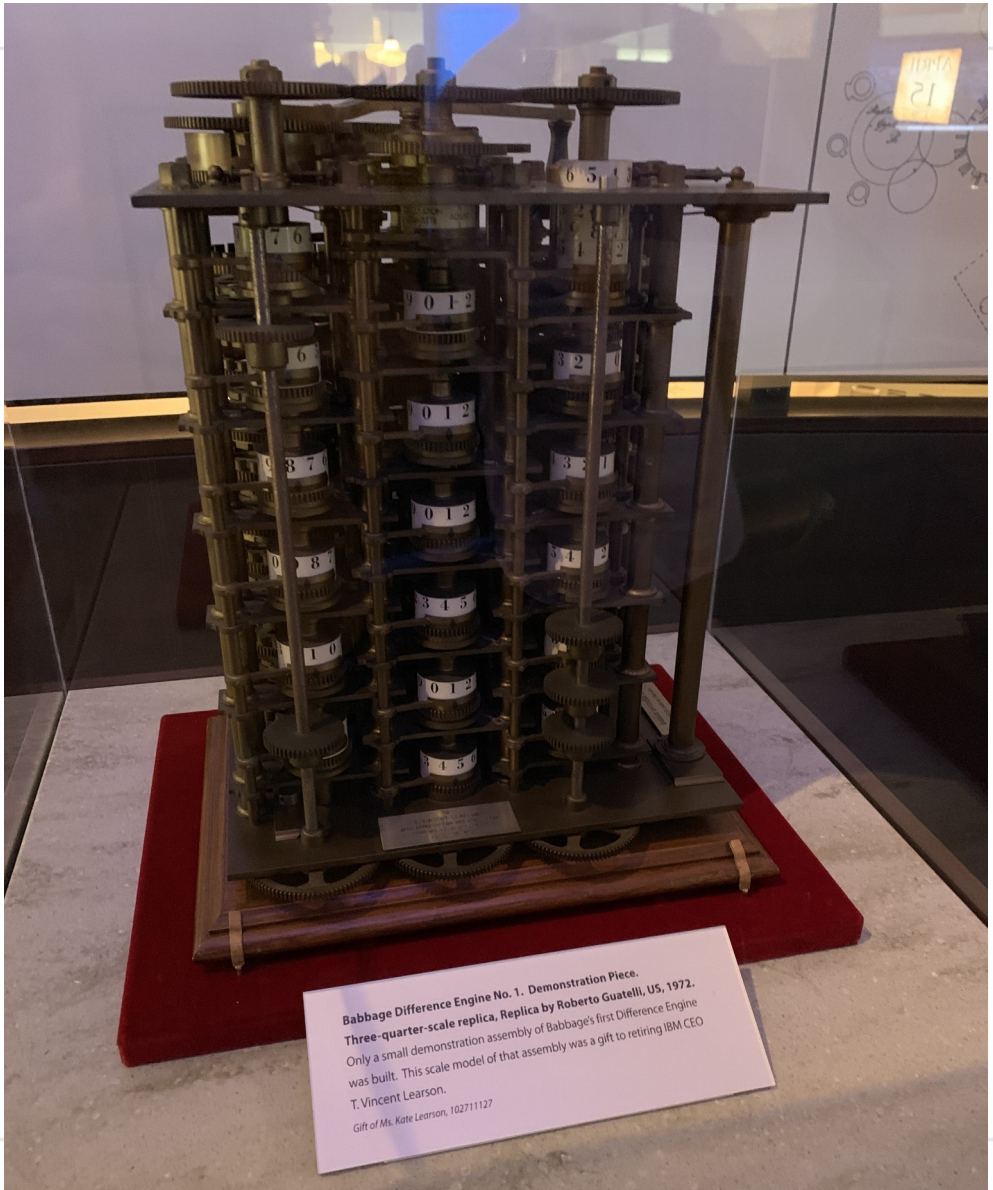
INSTITUTO FEDERAL
Mato Grosso do Sul

• Geração zero – computadores mecânicos (1642 – 1945)

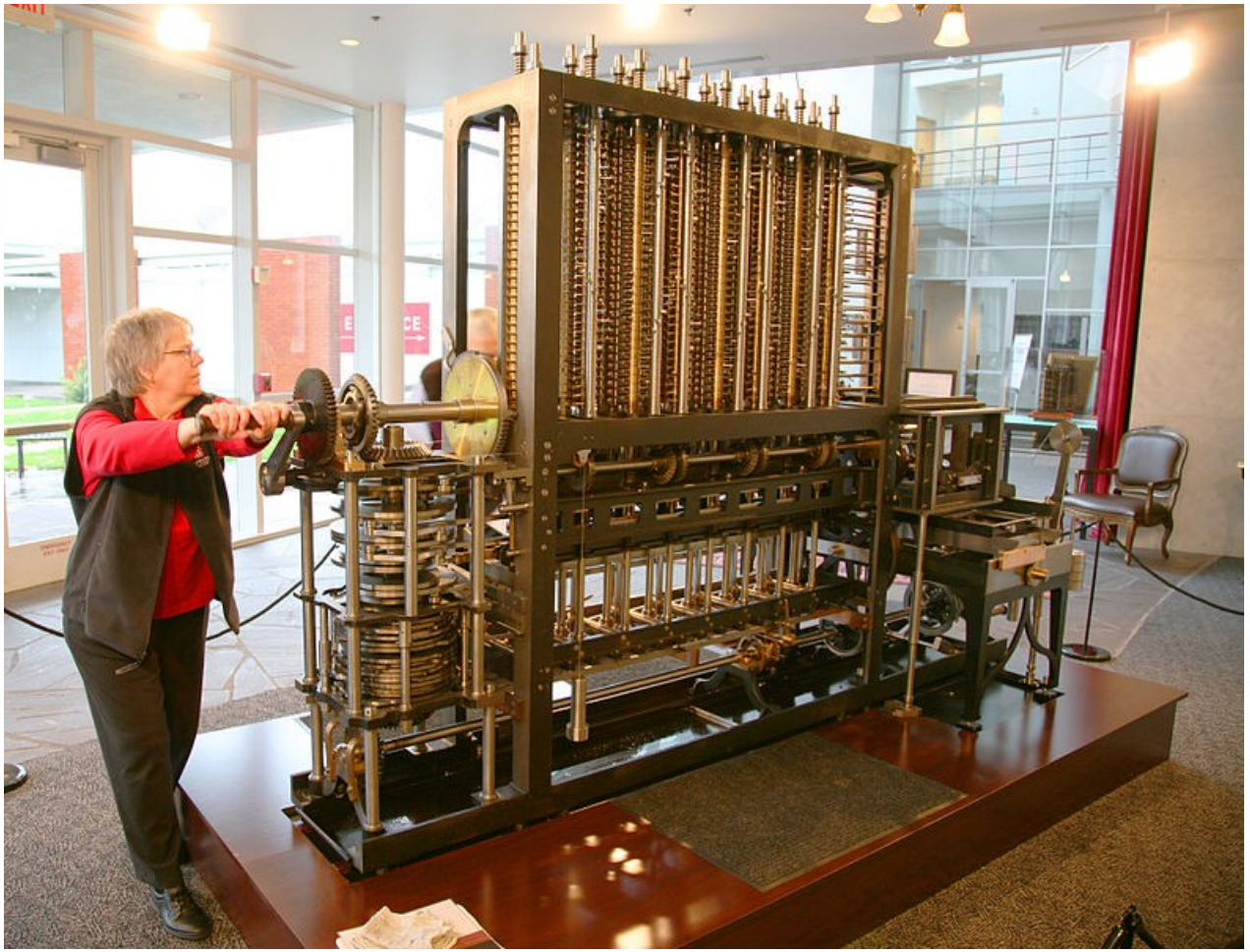
- Blaise Pascal (1623-1662) foi um cientista francês que construiu a primeira **máquina de calcular operacional** aos 19 anos para ajudar seu pai que era coletor de impostos.
  - A máquina era toda mecânica, usava engrenagens e funcionava com uma manivela operada à mão.
- Charles Babbage (1792-1871) inventou a primeira **máquina diferencial**. Realizava somas e subtrações e foi usado para auxiliar a navegação naval.
  - Idealizou também a **máquina analítica** – a primeira com propósito geral. Recebeu financiamento francês e gastou fortunas mas a máquina apresentava problemas, pois seu projeto estava a frente de seu tempo.

INSTITUTO FEDERAL
Mato Grosso do Sul

INSTITUTO FEDERAL
Mato Grosso do Sul



INSTITUTO FEDERAL
Mato Grosso do Sul



INSTITUTO FEDERAL
Mato Grosso do Sul

• Geração zero – computadores mecânicos (1642 – 1945)

- Mark I – Primeiro computador norte americano de uso geral. Construído em 1944 usando relés. Tinha 72 palavras de 23 algarismos decimais cada e um tempo de instrução de 6 segundos. Entrada e saída usavam fita e papel perfurado.

INSTITUTO FEDERAL
Mato Grosso do Sul

- **Primeira geração – válvulas (1945-1955)**

Válvula termiônica

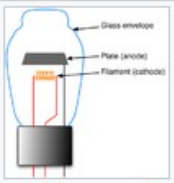
Origem: Wikipédia, a enciclopédia livre.

Válvula termiônica (português brasileiro) ou **válvula termiónica** (português europeu), também chamada por vezes de **tubo de vácuo**[1] é um dispositivo eletrônico formado por um invólucro de vidro de alto vácuo chamada ampola contendo vários elementos metálicos. Já as válvulas do tipo Tiratron, para aplicações de alta potência, são preenchidas com gás.[1]

Válvula de potência ainda fabricada.

Índice [mostrar]

Constituição interna [editar | editar código-fonte]



Diôdo Termiônico, diagrama simplificado.

Os elementos metálicos internos são, o filamento, cuja função é o aquecimento do cátodo para a emissão de elétrons, o cátodo, emissor de elétrons, a placa, ou ânodo, receptor de elétrons, a grade de controle, que, dependendo de sua polarização, aumenta ou diminui o fluxo eletrônico do cátodo ao ânodo, além de outras grades que podem formar as válvulas tríodos, pentodos, etc.

Diôdos [editar | editar código-fonte]

Díodos termiônicos, são válvulas eletrônicas de construção mais simplificada, inicialmente construídos por Thomas Alva Edison antes da invenção da lâmpada incandescente.

O díodo é formado mecanicamente de um filamento, cuja função é aquecer ao cátodo, acelerando desta forma os elétrons em direção ao ânodo, ou placa, que consiste num invólucro metálico que veste ao cátodo e filamento.

Funcionamento [editar | editar código-fonte]


Válvula tríodo utilizada em 1906.

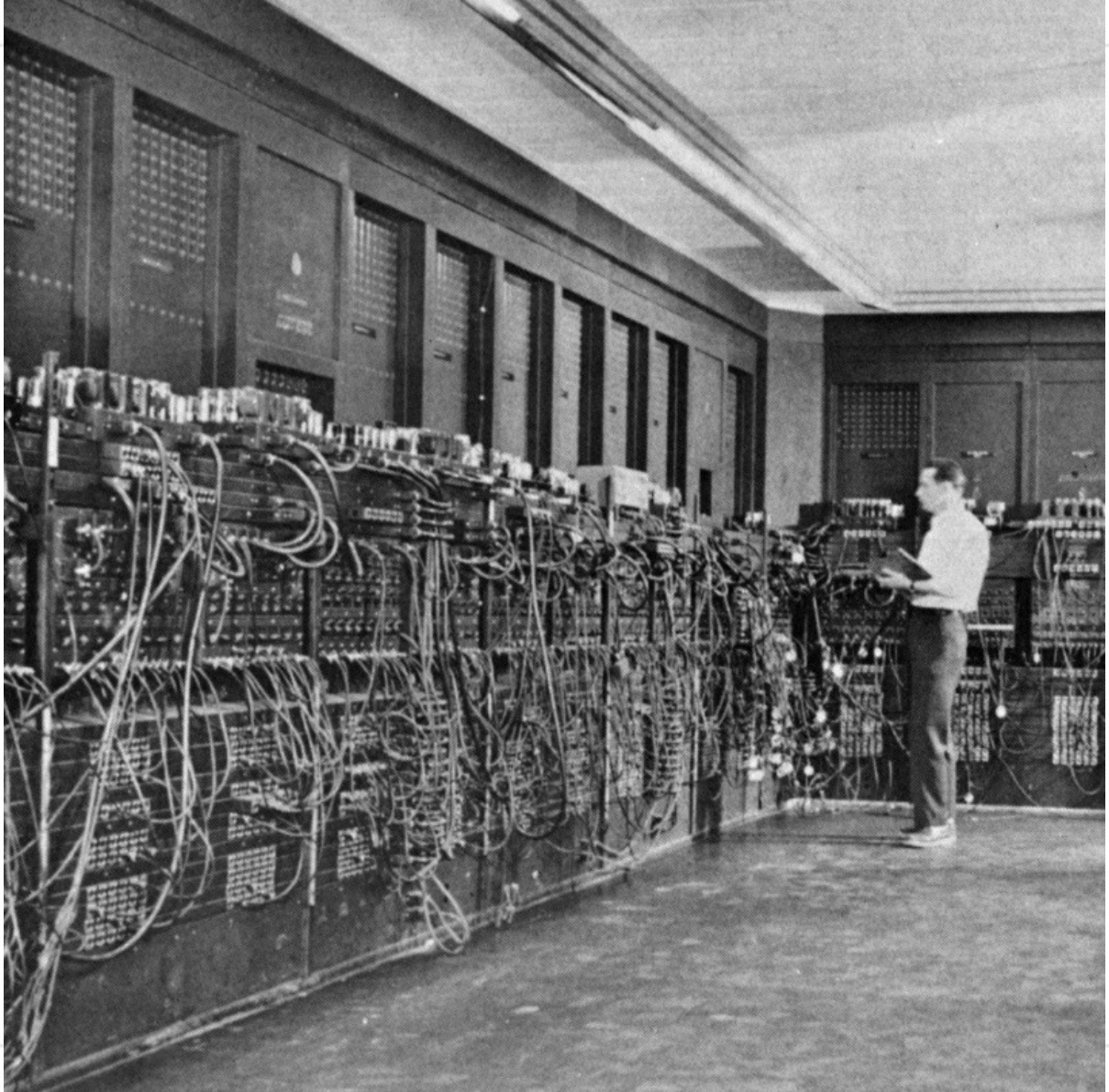
O funcionamento do diodo termiônico é bem simples, ao ligarmos uma bateria e um miliamperímetro em série, sendo o polo positivo à placa e o polo negativo ao cátodo, este sendo aquecido a determinada temperatura e a partir de uma certa tensão elétrica aplicada ao sistema, começará fluir uma corrente elétrica constante entre cátodo e placa (ânodo), não importando a oscilação da tensão, a intensidade de corrente será sempre a mesma, a este fenômeno se deu o nome de Efeito Édison.

INSTITUTO FEDERAL
Mato Grosso do Sul

• Primeira geração – válvulas (1945-1955)

- O computador ENIAC: O exército americano quer uma máquina que faça complexos cálculos balísticos. John Mauchly e J. Presper Eckert apresentam o projeto de uma máquina com válvulas eletrônicas. Em 1945 começa a funcionar o ENIAC (Eletronical Numerical Integrator and Computer).
- O ENIAC mede 5,5m de altura por 25m de comprimento e pesa 30 toneladas.
- O ENIAC, durante o período de funcionamento fez mais cálculos do que toda a humanidade realizou desde sua existência.

INSTITUTO FEDERAL
Mato Grosso do Sul




FUNCTION TABLE C


ENIAC
In 1942, physicist John Mauchly proposed an all-electronic calculating machine. The U.S. Army, meanwhile, needed to calculate complex wartime ballistics tables. Proposal met patron.
The result was ENIAC (Electronic Numerical Integrator And Computer), built between 1943 and 1945—the first large-scale computer to run at electronic speed without being slowed by any mechanical parts. For a decade, until a 1955 lightning strike, ENIAC may have run more calculations than all mankind had done up to that point.
ORIGINAL ENIAC

INSTITUTO FEDERAL
Mato Grosso do Sul

• Primeira geração – válvulas (1945-1955)

- John Von Neumann decidiu fazer uma versão que sucederia o ENIAC.
  - Neumann era um gênio com memória fotográfica e era um matemático extremamente inteligente
- Neumann viu que programar computadores com quantidades imensas de interruptores e cabos era algo lento, tedioso e sujeito a falhas.
  - Percebeu que o programa podia ser representado em forma digital na memória, junto com os dados.
  - Também percebeu que a aritmética decimal era desajeitada e podia ser substituída pela aritmética binária.
- Inventou a **arquitetura de Von Neumann**.

INSTITUTO FEDERAL
Mato Grosso do Sul

Máquina original de Von Neumann

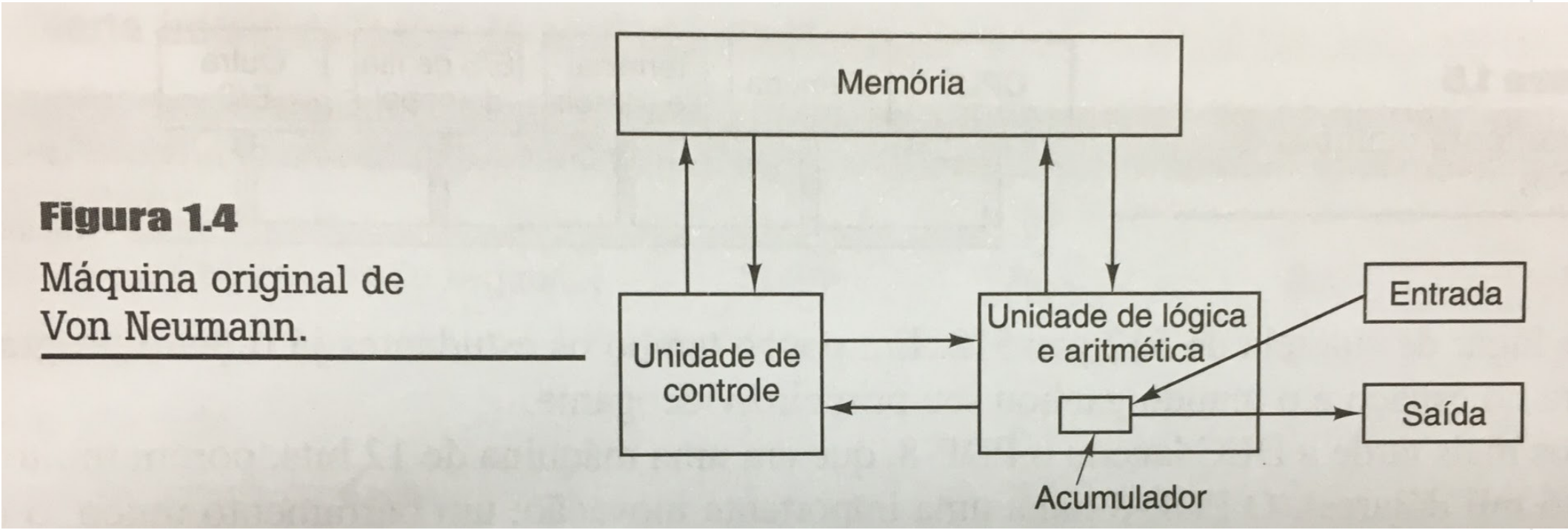


Figura 1.4

Máquina original de Von Neumann.

INSTITUTO FEDERAL
Mato Grosso do Sul

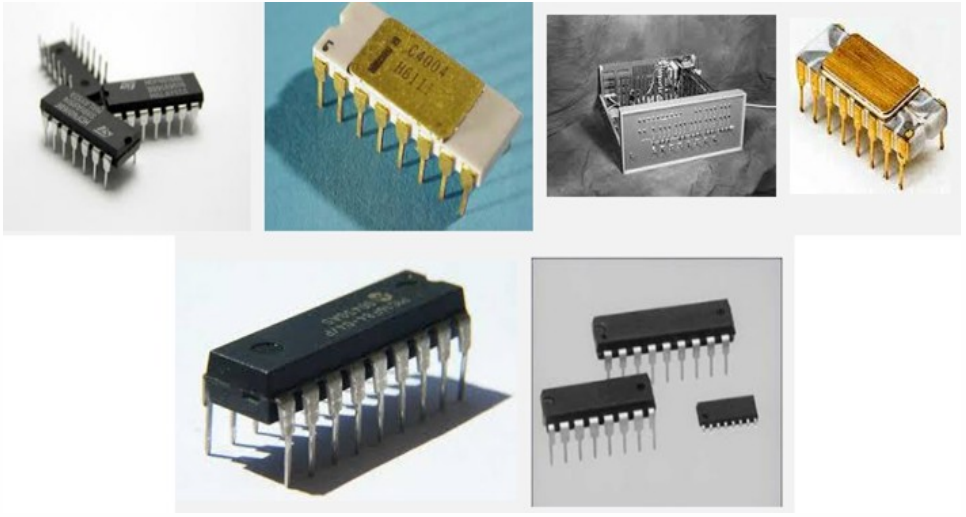
• Segunda geração – transistores (1955-1965)

- A invenção do transistor substitui as válvulas, servindo de base para a criação dos circuitos integrados e mais tarde, dos modernos processadores.

INSTITUTO FEDERAL
Mato Grosso do Sul

• Terceira geração – Circuitos Integrados (CI) (1965-1980)

- O circuito integrado utiliza transistores alojados em pequenas cápsulas de material semicondutor. Circuitos eletrônicos imensos passam a ser compactados em pequenos chips.
-



Quarta Geração (1980-?):

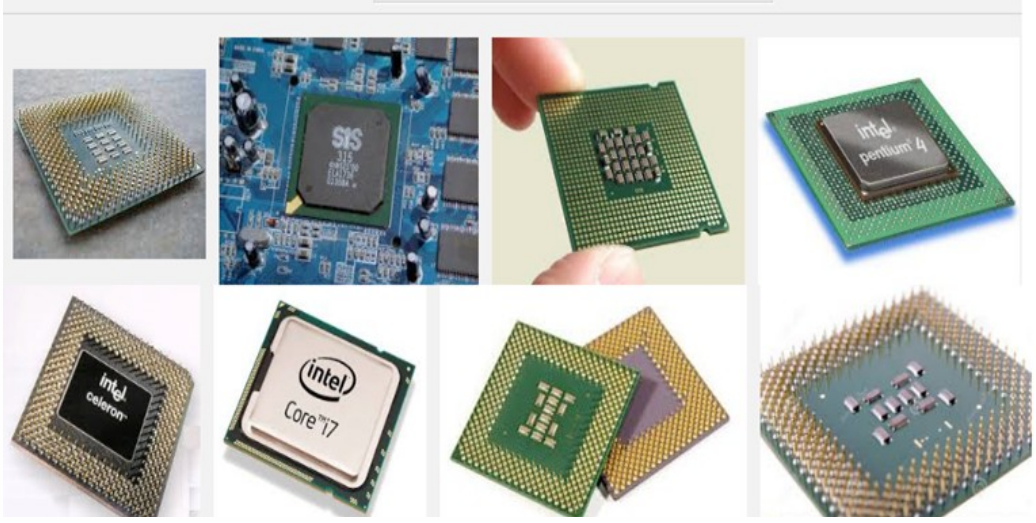
Computadores com Integração em Escala Muito Grande – Very Large Scale Integration (VLSI)

- Década de 80: grande compactação dos circuitos integrados
  - Dezenas de milhares, depois centenas de milhares e finalmente milhões de transistores em um chip
  - Desempenho aumentou muito
  - Preços caíram muito
  - Computadores deixaram de ser privilégio de grandes corporações
  - Início da era do **computador pessoal**

INSTITUTO FEDERAL
Mato Grosso do Sul

• Quarta geração – Integração em escala muito grande (1980-?)

- São usados microprocessadores, houve um ganho no tamanho, custo e velocidade no processamento de dados.



INSTITUTO FEDERAL
Mato Grosso do Sul

• Quinta geração – computadores invisíveis

- Também chamado de computação ubíqua (ou pervasiva): computadores estão presentes em locais da nossa rotina e nem nos damos conta
  - Carros, cafeteiras, micro-ondas, relógios, etc.

INSTITUTO FEDERAL
Mato Grosso do Sul

Conceitos

- **Dispositivos de entrada:** dispositivos de entrada são aqueles que o usuário utiliza para entrar com dados no computador.
  - Exemplos: teclado, mouse, scanner, microfone, webcam, leitor de digital
- **Dispositivos de saída:** são os dispositivos que o computador utiliza para mostrar dados e informações ao usuário.
  - Exemplos: monitor, impressora, caixa de som
- Você acha que existem dispositivos que podem atuar tanto como dispositivo de entrada quanto de saída?

INSTITUTO FEDERAL
Mato Grosso do Sul

Lei de Moore

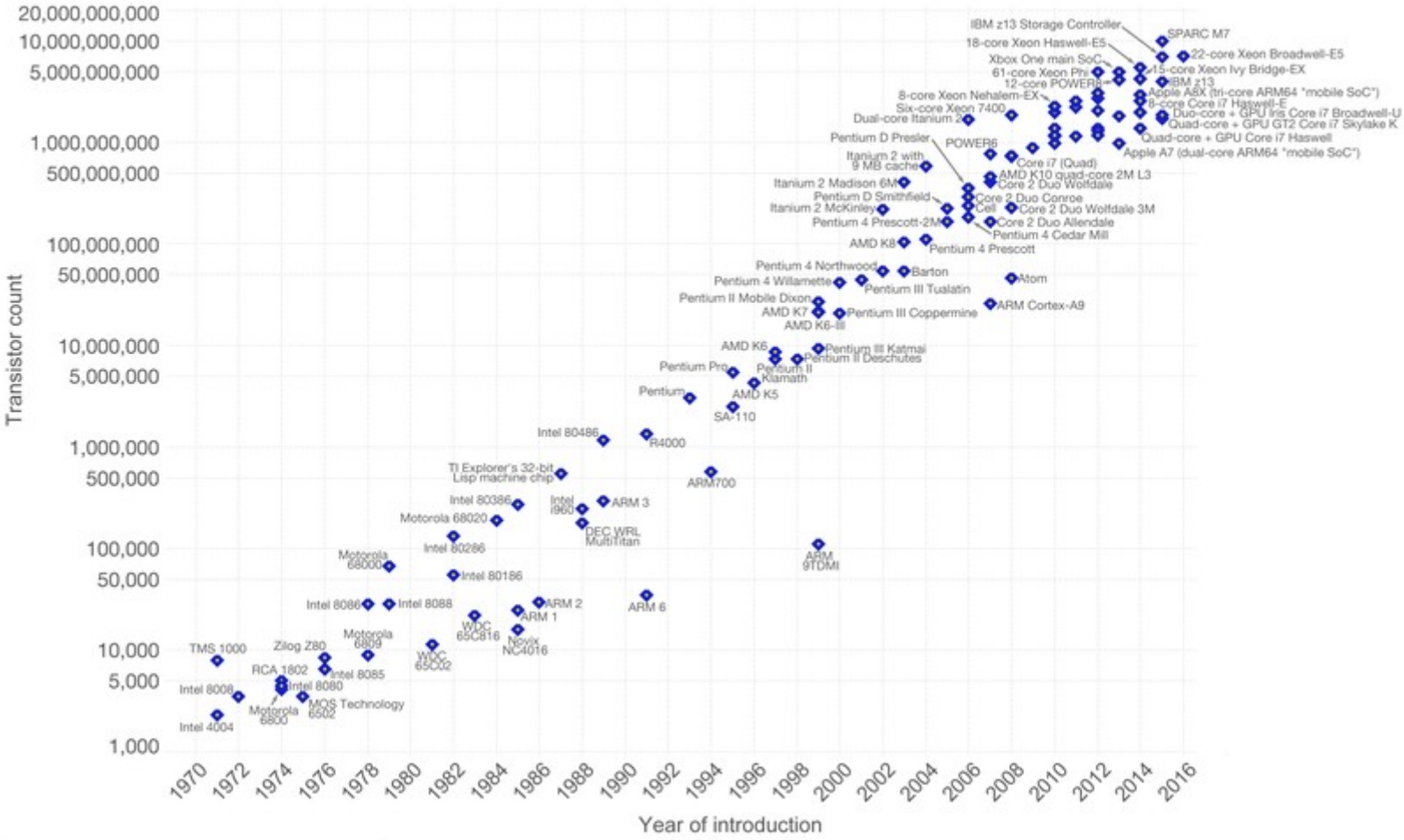
- Gordon Moore, 1965, Intel
- Número de transistores em um chip dobra a cada 18 meses.
- **Círculo Virtuoso**:
  - Avanço tecnológico propicia melhores produtos a preços mais baixos.
  - Preços mais baixos induz ao surgimento de novas aplicações (exemplo, video games)
  - Novas aplicações aumentam as possibilidades de mercado e fazem surgir novas empresas.
  - Novas empresas leva a competição, criando demanda econômica para o avanço tecnológico.

INSTITUTO FEDERA
Mato Grosso do Sul

Moore's Law – The number of transistors on integrated circuit chips (1971-2016)

Our World in Data

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are strongly linked to Moore's law.



Data source: Wikipedia (https://en.wikipedia.org/wiki/Transistor_count)

The data visualization is available at OurWorldinData.org. There you find more visualizations and research on this topic.

INSTITUTO FEDERAL
Mato Grosso do Sul

Bibliografia

- TANEMBAUM, A. S. - Organização Estruturada de Computadores – 5ª ed. Editora Pearson. 2007. ISBN 85-7605-067-6. 449 p.

Dúvidas?

INSTITUTO FEDERAL
Mato Grosso do Sul