

USP - ICMC - SSC
SSC 0610 - Eng. Comp. - 2o. Semestre 2010

Disciplina de Organização de Computadores I

Prof. Fernando Santos Osório

Email: fosorio [at] { icmc. usp. br , gmail. com }

Página Pessoal: <http://www.icmc.usp.br/~fosorio/>

Estagiário PAE Maurício Dias - Email: [macccdias \[at\] gmail.com](mailto:macccdias@gmail.com)

Material on-line Wiki ICMC - <http://wiki.icmc.usp.br/index.php/Ssc-610>

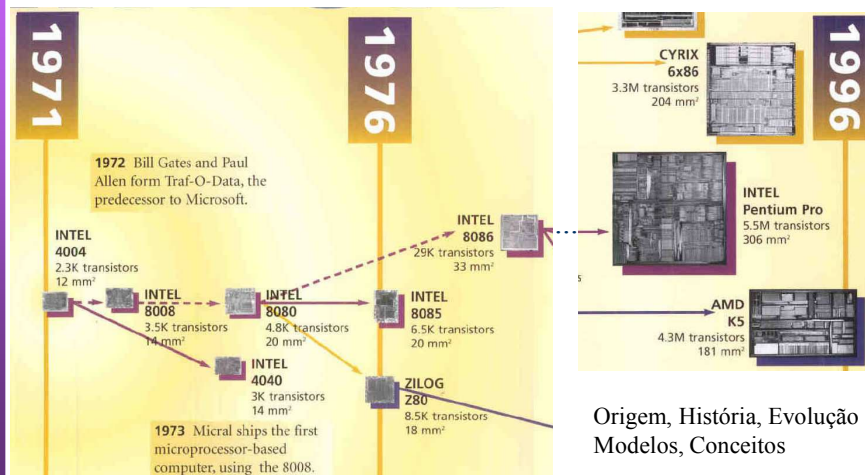
Aula 01q

Apresentação da Disciplina

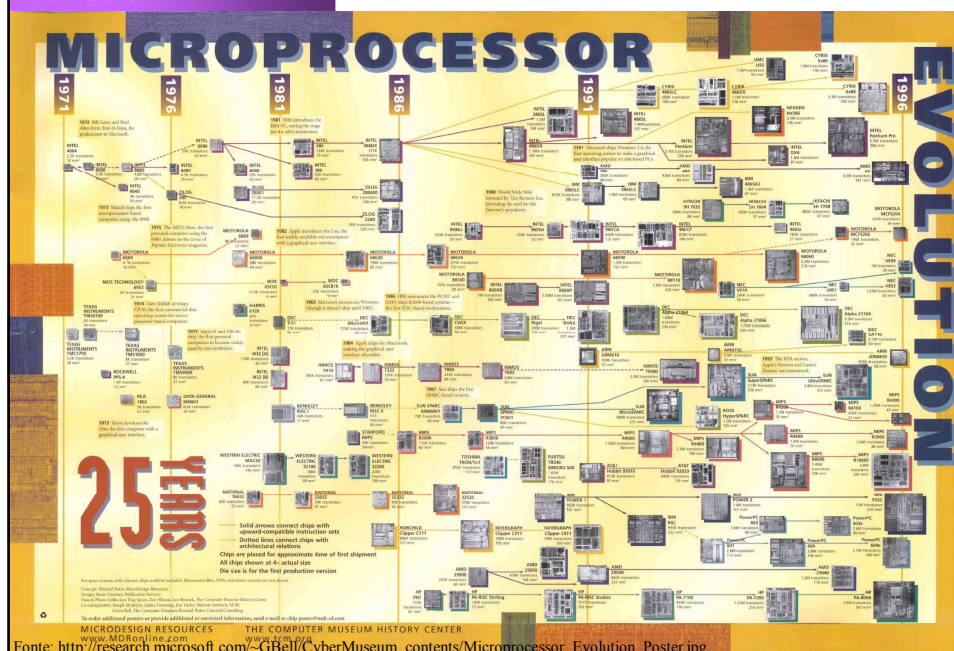
Agenda:

- 1. Introdução a Organização de Computadores:**
Componentes e Blocos Básicos
- 2. Arquitetura de Von Neumann**
- 3. Lógica Digital – Conceitos Básicos**
 - Sistemas Numéricos**
 - Circuitos combinacionais seqüenciais**
- 4. Construindo uma ULA**
 - Somadores**
 - Representação Numérica (Inteiros)**
 - Operadores, Operações & Bits de Controle/Status**

Motivação...

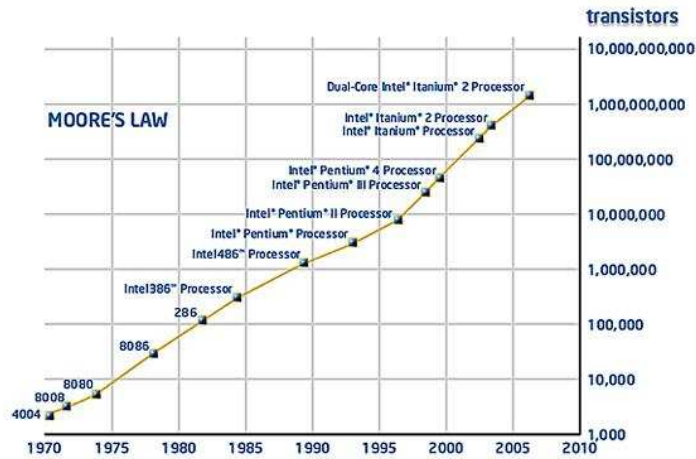


Origem, História, Evolução
Modelos, Conceitos



Arquitetura de Computadores

Evolução...



5

Agosto 2009

Fonte: <http://www.edumax.com/computer-hardware-cpu-parts-and-evolution.html>

Arquitetura de Computadores

Evolução...

Select by chip logo



CPU's by Manufacturer

AMD
AMI
Chips and Technologies
Cypress
Cyrix
DEC
Eastern Bloc
EPSON
Fairchild
Fujitsu
Harris
Hitachi
HP
IBM
IDT
IIT
Intel
Intergraph
LSI Logic
MHS
Mitsubishi
MOS
Mostek
Motorola
National Semiconductor
NEC
NEC
NexGen
OKI
Performance
Semi
Philips
QED
Rise
Rockwell
SGS
SGS-Thomson
Siemens
Signetics
Sony

Select by brand label

Am486 Am5 x 86™ AMD-K5
AMD-K6 **Athlon** Cx486
DX2 536
FasMath™ Nx586™ BLUE LIGHTNING
6x86 6x86MX PowerPc™
MII It's 57 i386 i486
OVERDRIVE celeron pentium
pentium®II PENTIUM®PRO SPARC
Athlon64-x2

6

Agosto 2009

Fonte: <http://www.cpu-collection.de/>

2. Arquitetura de Von Neumann

Breve Histórico:

Blaise Pascal (1623 - 1662)

1642 - Máquina de calcular mecânica (engrenagens e alavancas)

Permitia fazer adições e subtrações - Inovação: vai-um, repr. numérica

Leibniz (1646 - 1716)

1671 - Máquina de calcular mecânica

Permitia fazer as 4 operações - Inovação: multiplicação e divisão

Charles Babbage (1792 – 1871)

1827 - Máquina Diferencial: implementava o método de diferenças finitas

Inovação: operação automática com diversos passos

1834 - Máquina Analítica: proposta de uma máquina de propósito geral.

Inovação: Mecanismo automático de controle de seqüência de programa

Zuse: Z3 - 1941 (Computadores com circuitos eletro-mecânicos)

Aiken: Mark I - 1944

John von Neumann (1903-1957)

ENIAC (Electronic Numerical Integrator and Computer) - 1946 (Válvulas)

EDVAC (Electronic Discrete Variable Automatic Calculator)

IAS (Institute for Advanced Study – Princeton, USA) => **Arquitetura Von Neumann**

3. Lógica Digital – Conceitos Básicos Sistemas Numéricos

- Sistema decimal
 - 10 dígitos: 0 1 2 3 4 5 6 7 8 9
 - Casas decimais: ... 10^3 10^2 10^1 1
- Sistema binário
 - 2 dígitos: 0 1
 - Casas binárias: 2^3 2^2 2^1 1
- Sistema hexadecimal
 - 16 dígitos : 0 1 2 3 4 5 6 7 8 9 A B C D E F
 - Casas hexadecimais: ... 16^3 16^2 16^1 1

3. Lógica Digital – Conceitos Básicos

Sistemas Numéricos

0 0 0 0 0 0 0 0
0 0 00h - 0d

0 1 0 1 1 1 1 1 1+2+4+8+16+64
5 F 5Fh - 95d

1 0 1 0 1 0 1 0 => ?

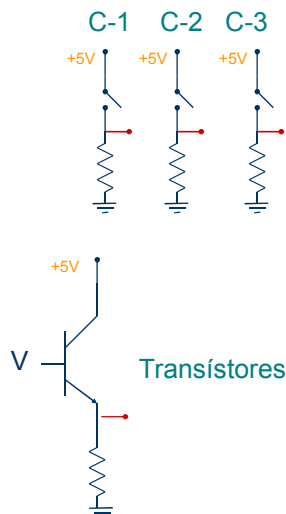
8 bits, 16 bits, 32 bits - MSB / LSB

Big-endian / Little-endian

9

Agosto 2009

LÓGICA BINÁRIA



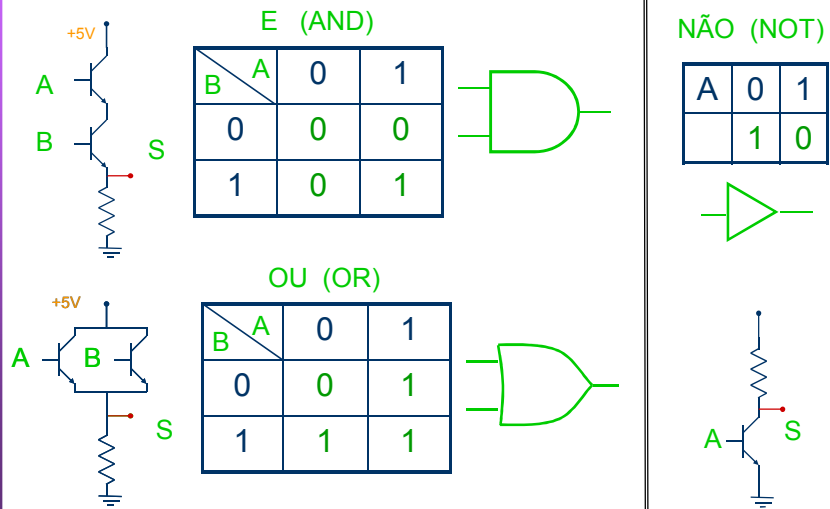
C-1	C-2	C-3	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

10

Agosto 2009

Slides By E. Simões, Sarita, Wikipedia

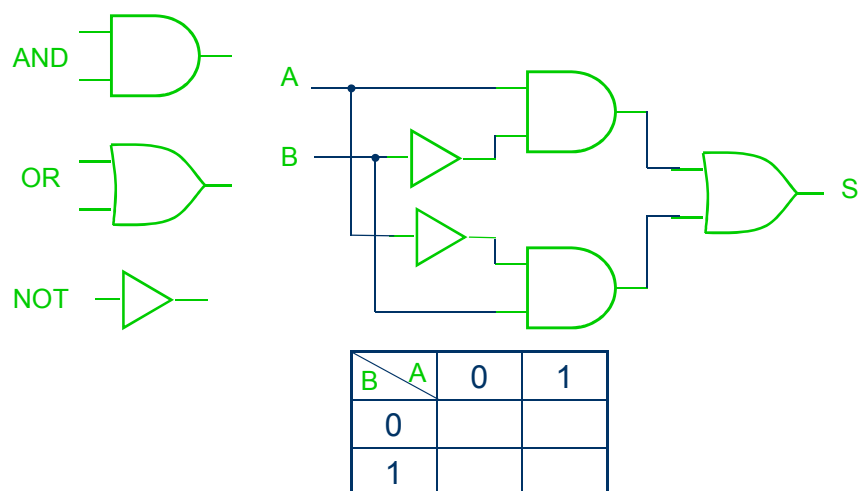
CIRCUITOS LÓGICOS



11

Agosto 2009

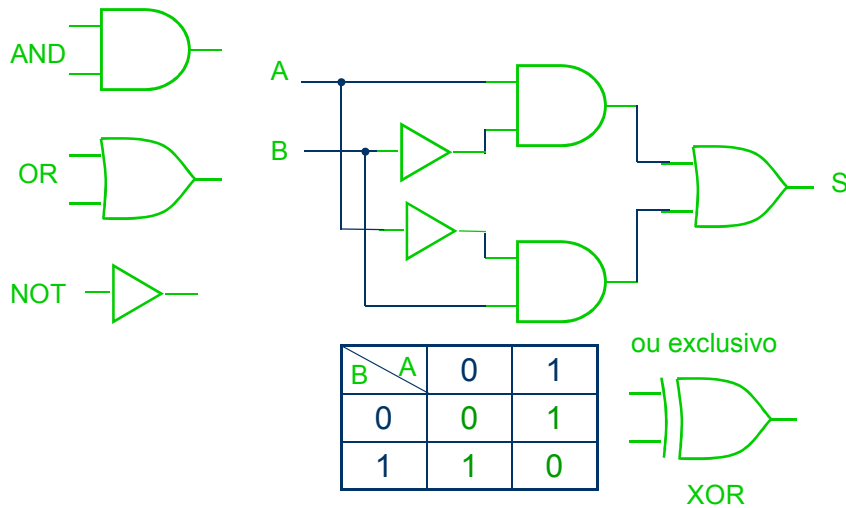
CIRCUITOS LÓGICOS



12

Agosto 2009

CIRCUITOS LÓGICOS

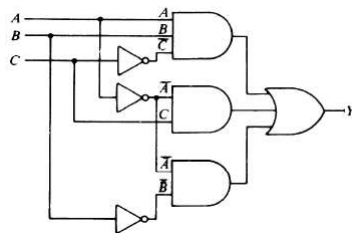


13

Agosto 2009

3. Circuitos Combinacionais Seqüenciais

Circuito Combinacional



Expressão Lógica: $A.B.\bar{C} + \bar{A}.C + \bar{A}.\bar{B}$

Otimização, Simplificação, Mapas de Karnaugh, ...
Circuitos: Comparadores, Somadores, (De)Multiplexadores,
(De)Codificadores, Flip-Flops, Contadores, ...

Ref.: Idoeta/Capuno. Elementos de Eletrônica Digital

14

Agosto 2009

2. Arquitetura de Von Neumann

Breve Histórico:

Blaise Pascal (1623 - 1662)

1642 - Máquina de calcular mecânica (engrenagens e alavancas)

Permitia fazer adições e subtrações - Inovação: vai-um, repr. numérica

Leibniz (1646 - 1716)

1671 - Máquina de calcular mecânica

Permitia fazer as 4 operações - Inovação: multiplicação e divisão

Charles Babbage (1792 – 1871)

1827 - Máquina Diferencial: implementava o método de diferenças finitas

Inovação: operação automática com diversos passos

1834 - Máquina Analítica: proposta de uma máquina de propósito geral.

Inovação: Mecanismo automático de controle de seqüência de programa

Zuse: Z3 - 1941 (Computadores com circuitos eletro-mecânicos)

Aiken: Mark I - 1944

John von Neumann (1903-1957)

ENIAC (Electronic Numerical Integrator and Computer) - 1946 (Válvulas)

EDVAC (Electronic Discrete Variable Automatic Calculator)

IAS (Institute for Advanced Study – Princeton, USA) => **Arquitetura Von Neumann**

15

Agosto 2009

2. Arquitetura de Von Neumann

Arquitetura - Modelo Inicial:

- John von Neumann
 - Construiu em 1952 o computador IAS (*Institute for Advanced Study – Princeton, USA*)
 - Programa Armazenado: programas e dados representados de forma digital em memória
 - Processamento baseado em aritmética binária, ao invés de decimal
- Máquina de Von Neumann
 - Componentes:
 - > Memória
 - > Unidade Lógica e Aritmética (ULA)
 - > Unidade de Controle
 - > Dispositivos de entrada/saída
 - Memória EDVAC: 1024 palavras de 44 bits
 - Memória IAS: 4096 palavras de 40 bits (2 instruções de 20 bits / inteiro c/sinal)
 - Instrução IAS: 8 bits para indicar o tipo, 12 bits para endereçar a memória
 - Acumulador IAS: registrador especial de 40 bits. Tem por função armazenar um operando e/ou um resultado fornecido pela ULA.

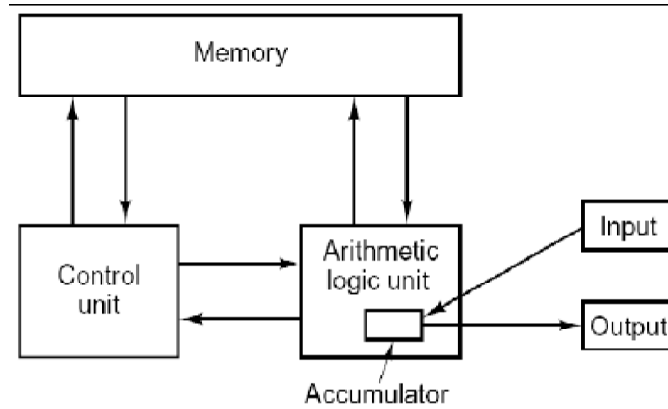
16

Agosto 2009

2. Arquitetura de Von Neumann

Arquitetura - Modelo Inicial:

- Máquina de von Neumann



17

Agosto 2009

Refs: http://en.wikipedia.org/wiki/Von_Neumann_architecture

2. Arquitetura de Von Neumann

Arquitetura - Modelo Inicial:

Von Neumann: Arquitetura de Computadores, Mecânica Quântica,
Teoria de Jogos - Theory of Games and Economic Behavior
Inteligência Artificial / Redes Neurais - The Computer and the Brain



John von Neumann in the 1940s



Help us provide free content to the world by [donating today!](#)
[article](#) [discussion](#) [edit this page](#) [history](#)

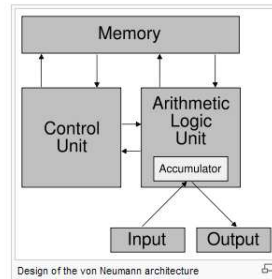
Von Neumann architecture

From Wikipedia, the free encyclopedia

The **von Neumann architecture** is a design model for a stored-program digital computer that uses a processing unit and a single separate storage structure to hold both instructions and data. It is named after mathematician and early computer scientist John von Neumann. Such a computer implements a universal Turing machine, and the common "referential model" of specifying sequential architectures, in contrast with parallel architectures.

A **stored-program** digital computer is one that keeps its program instructions as well as its data in read-write, random access memory. Stored-program computers were an advancement over the program-controlled computers of the 1940s, such as Colossus and ENIAC, which were programmed by setting switches and inserting patch leads to route data and control signals between various functional units. In the majority of modern computers, the same memory is used for both data and program instructions.

The terms "von Neumann architecture" and "stored-program computer" are generally used interchangeably, and that usage is followed in this article. In contrast, the Harvard architecture stores a program in a modifiable form, but without using the same physical storage or format as for general data.



Design of the von Neumann architecture

18

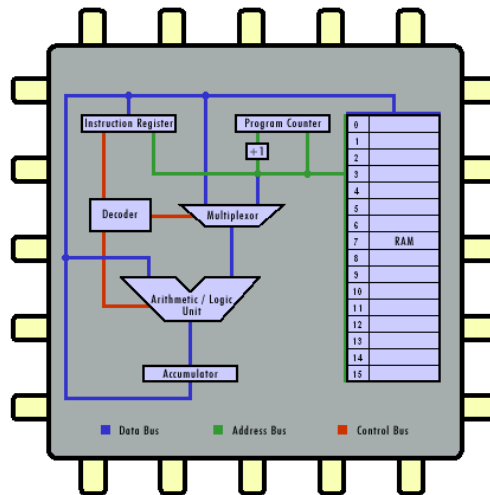
Agosto 2009

Refs: http://en.wikipedia.org/wiki/Von_Neumann_architecture
http://en.wikipedia.org/wiki/Von_Neumann

<http://en.wikipedia.org/wiki/Edvac>
http://en.wikipedia.org/wiki/IAS_Computer

2. Arquitetura de Von Neumann

CPU – Processador



19

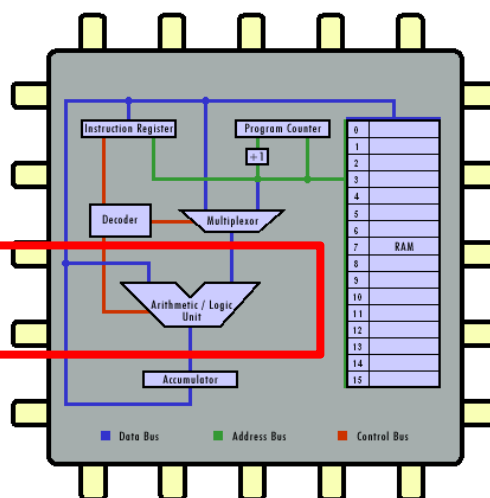
Agosto 2009

Fonte: <http://courses.cs.vt.edu/~csonline/MachineArchitecture/Lessons/CPU/index.html>

2. Arquitetura de Von Neumann

CPU – Processador

ULA / ALU
Unidade Lógico-Aritmética



20

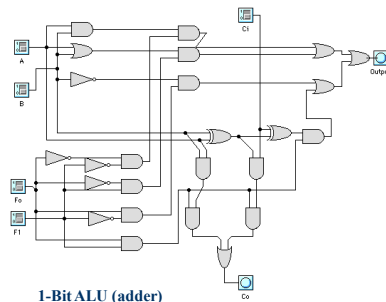
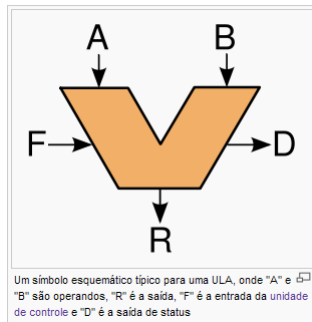
Agosto 2009

Fonte: <http://courses.cs.vt.edu/~csonline/MachineArchitecture/Lessons/CPU/index.html>

4. Unidade Lógica e Aritmética (ULA)

Unidade Lógico Aritmética - ULA

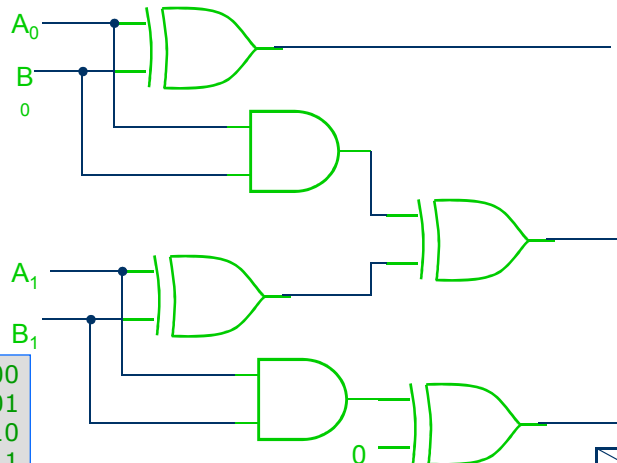
A *Unidade Lógica e Aritmética* (ULA) ou em inglês Arithmetic Logic Unit (ALU) é uma parte da Unidade Central do Processador (Central Processing Unit - CPU). Esta unidade é que realmente executa as operações aritméticas e lógicas referenciadas pelos opcodes das instruções.



21

Agosto 2009

SOMADOR



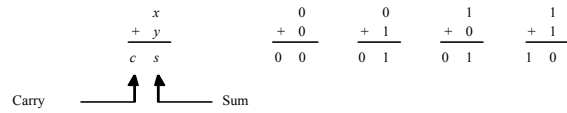
Decimal: 2
Binário: 10
+
Decimal: 3
Binário: 11

010
+
011
=
101

0 = 000
1 = 001
2 = 010
3 = 011
4 = 100
5 = 101
6 = 110
7 = 111

B \ A	0	1
0	0	1
1	1	0

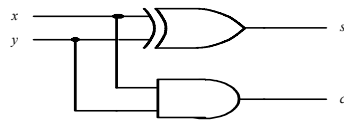
SOMADOR: Half Adder



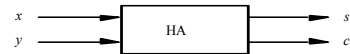
(a) The four possible cases

x	y	Carry c	Sum s
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

(b) Truth table



(c) Circuit



(d) Graphical symbol

23

Agosto 2009

SOMADOR: Full Adder

c_i	x_i	y_i	c_{i+1}	s_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

(a) Truth table

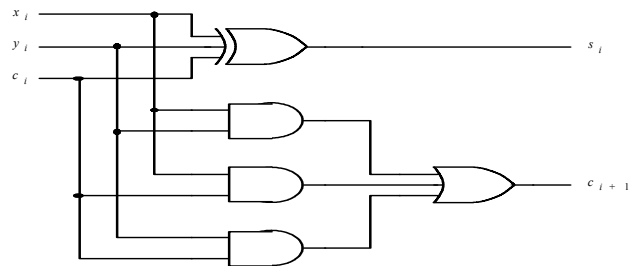
c_i	$x_i y_i$	00	01	11	10
0			1		1
1		1		1	

$$s_i = x_i \oplus y_i \oplus c_i$$

c_i	$x_i y_i$	00	01	11	10
0				1	
1			1	1	1

$$c_{i+1} = x_i y_i + x_i c_i + y_i c_i$$

(b) Karnaugh maps

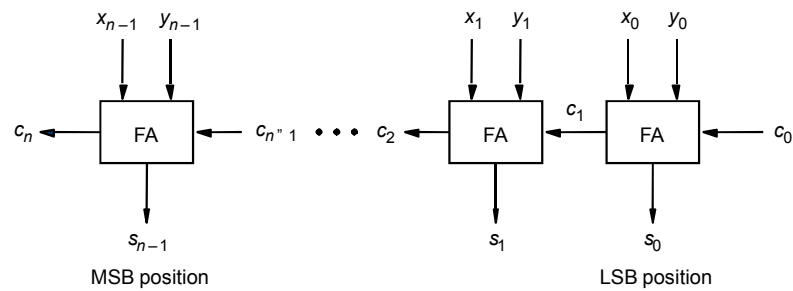


(c) Circuit

24

Agosto 2009

SOMADOR: n-bit Adder

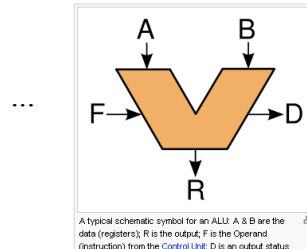
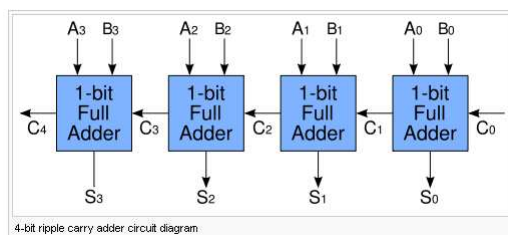
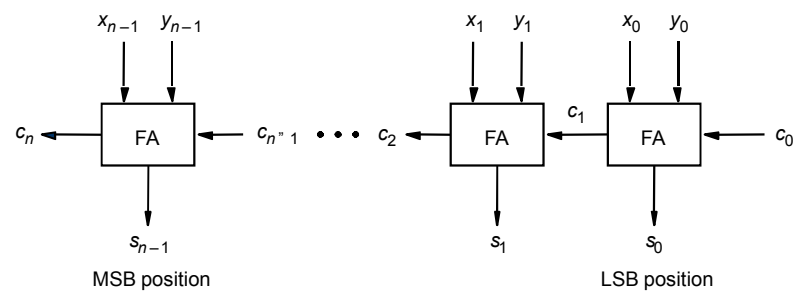


An n -bit ripple-carry adder

25

Agosto 2009

SOMADOR: n-bit Adder



26

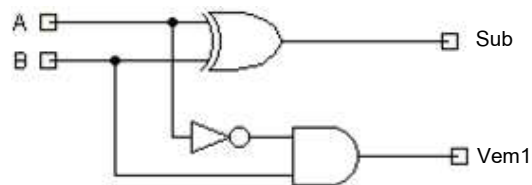
Agosto 2009

SUBTRATOR

Subtração entre 2 bits: meio-subtrator

$$A - B = A + \overline{B} + 1$$

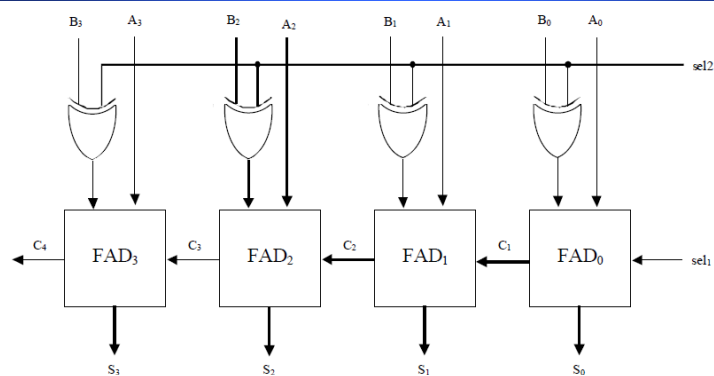
A	B	Sub	Vem1
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



27

Agosto 2009

SOMA e SUBTRAÇÃO



sel ₂	sel ₁	operação	descrição
0	0	$S = A + B + 0$	adiciona $A \in B$ ($S = A + B$)
0	1	$S = A + B + 1$	adiciona $A \in B$ incrementado ($S = A + B + 1$)
1	0	$S = A + \overline{B} + 0$	subtrai B decrementado de A ($S = A - B \cdot 1$)
1	1	$S = A + \overline{B} + 1$	subtrai B de A ($S = A - B$)

28

Agosto 2009

SOMA e SUBTRAÇÃO

Representação Numérica: Complemento de 2

Dados representados em complemento de 1: Inverte bits

Dados representados em complemento de 2: Inverte + 1

Valores Negativos:

-1 => 0001 em C2 => 1111

Subtração:

$8 - 3 = 8 + (-3) = 1000 + 1101 = 0101$

$3 - 5 = 3 + (-5) = 0011 + 1011 = 1110$

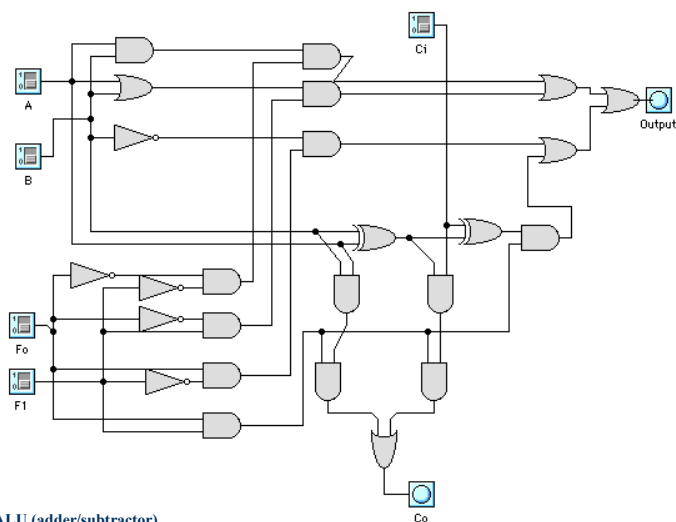
Complemento de 2	Decimal
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	-8

sel ₂	sel ₁	operação	descrição
0	0	$S = A + B + 0$	adiciona A e B ($S = A + B$)
0	1	$S = A + B + 1$	adiciona A e B incrementado ($S = A + B + 1$)
1	0	$S = A + \overline{B} + 0$	subtrai B decrementado de A ($S = A - B - 1$)
1	1	$S = A + \overline{B} + 1$	subtrai B de A ($S = A - B$)

29

Agosto 2009

ULA: SOMA e SUBTRAÇÃO



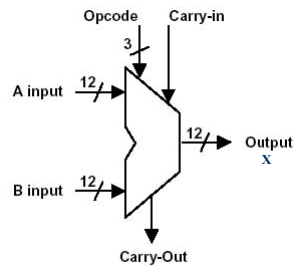
1-Bit ALU (adder/subtractor)

30

Agosto 2009

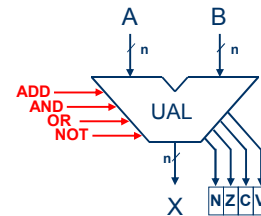
4. Unidade Lógica e Aritmética (ULA)

Unidade Lógico Aritmética - ULA



ALU OPCODEs:

- Add
- Subtract
- Multiply
- Divide
- Logical: AND, OR, NOT, XOR



ALU Inputs:

- > A (Accumulator), B: Data Input
- > OPCODE: Select Operation
- > Carry-In (vem-um)

ALU Outputs:

- > X (Accumulator): Data Output
- > Carry-Out (vai-um)
- > Outras informações de status:
Negativo/Sinal (N), Zero (Z), Carry (C),
Overflow (V), Underflow (U), Erros (E)

31

Agosto 2009

4. Unidade Lógica e Aritmética (ULA)

Unidade Lógico Aritmética - ULA

Exemplo: TTL 74181 ALU

Signetics

74181, LS181, S181
Arithmetic Logic Units

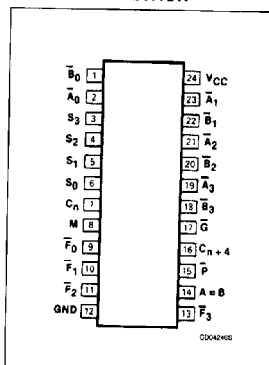
4-Bit Arithmetic Logic Unit
Product Specification

Logic Products

FEATURES

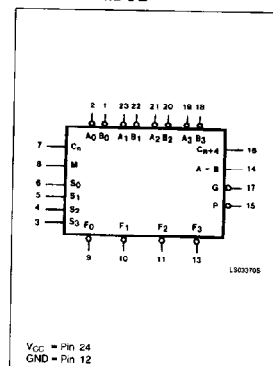
- Provides 16 arithmetic operations: ADD, SUBTRACT, COMPARE, DOUBLE, plus 12 other arithmetic operations
- Provides all 16 logic operations of two variables: Exclusive-OR, Compare, AND, NAND, NOR, OR, plus 10 other logic operations
- Full lookahead carry for high-speed arithmetic operation on long words

PIN CONFIGURATION



December 4, 1985

LOGIC SYMBOL



Vcc = Pin 24
GND = Pin 12

5-350

32

Agosto 2009

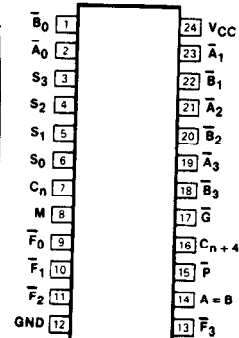
4. Unidade Lógica e Aritmética (ULA)

Unidade Lógico Aritmética - ULA

Exemplo: TTL 74181 ALU

MODE SELECT — FUNCTION TABLE

MODE SELECT INPUTS				ACTIVE HIGH INPUTS & OUTPUTS	
S ₃	S ₂	S ₁	S ₀	Logic (M = H)	Arithmetic** (M = L) (C _n = H)
L	L	L	L	$\bar{A}$	A
L	L	L	H	$\bar{A} + \bar{B}$	A + B
L	L	H	L	$\bar{A}B$	A + $\bar{B}$
L	L	H	H	Logical 0	minus 1
L	H	L	L	$\bar{A}\bar{B}$	A plus $\bar{A}\bar{B}$
L	H	L	H	B	(A + B) plus $\bar{A}\bar{B}$
L	H	H	L	$A \odot B$	A minus B minus 1
L	H	H	H	$\bar{A}B$	AB minus 1
H	L	L	L	$\bar{A} + B$	A plus AB
H	L	L	H	$\bar{A} \odot \bar{B}$	A plus B
H	L	H	L	B	(A + $\bar{B}$) plus AB
H	L	H	H	AB	AB minus 1
H	H	L	L	Logical 1	A plus A*
H	H	L	H	$A + \bar{B}$	(A + B) plus A
H	H	H	L	A + B	(A + B) plus A
H	H	H	H	A	A minus 1



33

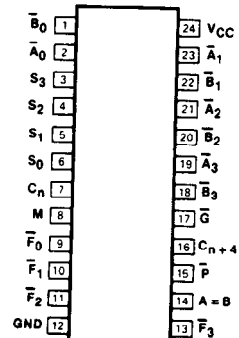
Agosto 2009

4. Unidade Lógica e Aritmética (ULA)

Unidade Lógico Aritmética - ULA

Exemplo: TTL 74181 ALU

MODE SELECT INPUTS				ACTIVE LOW INPUTS & OUTPUTS	
S ₃	S ₂	S ₁	S ₀	Logic (M = H)	Arithmetic** (M = L) (C _n = L)
L	L	L	L	$\bar{A}$	A minus 1
L	L	L	H	$\bar{A}\bar{B}$	AB minus 1
L	L	H	L	$\bar{A} + B$	AB minus 1
L	L	H	H	Logical 1	minus 1
L	H	L	L	$\bar{A} + \bar{B}$	A plus (A + $\bar{B}$)
L	H	L	H	$\bar{B}$	AB plus (A + $\bar{B}$)
L	H	H	L	$\bar{A} \odot \bar{B}$	A minus B minus 1
L	H	H	H	$A + \bar{B}$	A + $\bar{B}$
H	L	L	L	$\bar{A}B$	A plus (A + B)
H	L	L	H	$\bar{A} \odot B$	A plus B
H	L	H	L	B	$\bar{A}B$ (A + B)
H	L	H	H	A + B	A + B
H	H	L	L	Logical 0	A plus A*
H	H	L	H	$\bar{A}B$	AB plus A
H	H	H	L	AB	$\bar{A}\bar{B}$ plus A
H	H	H	H	A	A



L = LOW voltage

H = HIGH voltage level

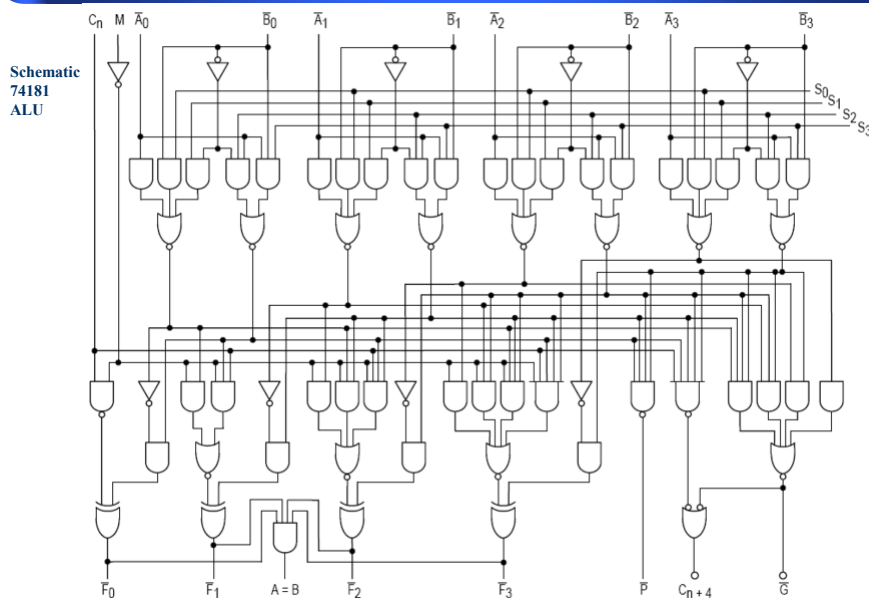
*Each bit is shifted to the next more significant position.

**Arithmetic operations expressed in 2's complement notation.

34

Agosto 2009

4. Unidade Lógica e Aritmética (ULA)



35

Agosto 2009

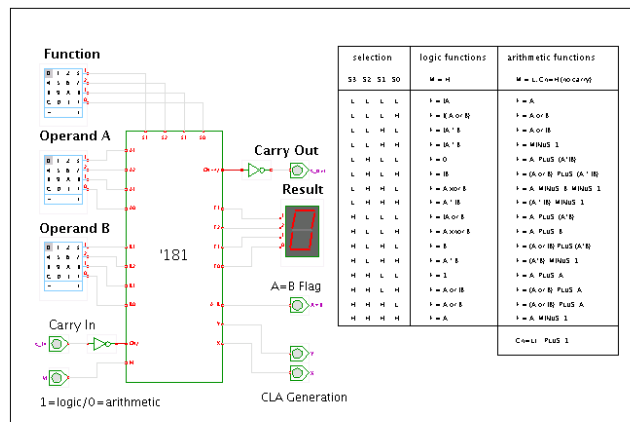
4. Unidade Lógica e Aritmética (ULA)

Hades, the Hamburg Design System, a framework for interactive simulation

HADES - JAVA Interactive Demo - Logic Circuits |

<http://tams-www.informatik.uni-hamburg.de/applets/hades/webdemos/index.html>

TTL-series 74181 ALU demonstration



36

Agosto 2009

Fonte: http://tams-www.informatik.uni-hamburg.de/applets/hades/webdemos/20-arithmetic/50-74181/demo-74181-ALU_print.html

4. Unidade Lógica e Aritmética (ULA)

SOMADOR: Definição em uma Linguagem de Descrição de Hardware (HDL)

```
c_1_addr.vhd => VHDL

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY c_1_addr IS
  PORT
  (
    x_in   : IN  STD_LOGIC_VECTOR(7 DOWNTO 0);
    y_in   : IN  STD_LOGIC_VECTOR(7 DOWNTO 0);
    carry_in : IN  STD_LOGIC;
    sum    : OUT STD_LOGIC_VECTOR(7 DOWNTO 0);
    carry_out : OUT STD_LOGIC
  );
END c_1_addr;

ARCHITECTURE behavioral OF c_1_addr IS

  SIGNAL h_sum      : STD_LOGIC_VECTOR(7 DOWNTO 0);
  SIGNAL carry_generate : STD_LOGIC_VECTOR(7 DOWNTO 0);
  SIGNAL carry_propagate : STD_LOGIC_VECTOR(7 DOWNTO 0);
  SIGNAL carry_in_internal : STD_LOGIC_VECTOR(7 DOWNTO 1);
```

37

Agosto 2009

Fonte: http://www.altera.com/support/examples/vhdl/v_cl_addr.html

4. Unidade Lógica e Aritmética (ULA)

SOMADOR: Definição em uma Linguagem de Descrição de Hardware (HDL)

c_1_addr.vhd (Continuação... Parte II)

```
BEGIN
  h_sum <= x_in XOR y_in;
  carry_generate <= x_in AND y_in;
  carry_propagate <= x_in OR y_in;

  PROCESS (carry_generate,carry_propagate,carry_in_internal)
  BEGIN
    carry_in_internal(1) <= carry_generate(0) OR (carry_propagate(0) AND carry_in);
    inst: FOR i IN 1 TO 6 LOOP
      carry_in_internal(i+1) <= carry_generate(i) OR (carry_propagate(i) AND carry_in_internal(i));
    END LOOP;
    carry_out <= carry_generate(7) OR (carry_propagate(7) AND carry_in_internal(7));
  END PROCESS;

  sum(0) <= h_sum(0) XOR carry_in;
  sum(7 DOWNTO 1) <= h_sum(7 DOWNTO 1) XOR carry_in_internal(7 DOWNTO 1);
END behavioral;
```

38

Agosto 2009

Fonte: http://www.altera.com/support/examples/vhdl/v_cl_addr.html



INFORMAÇÕES SOBRE A DISCIPLINA

USP - Universidade de São Paulo - São Carlos, SP
ICMC - Instituto de Ciências Matemáticas e de Computação
SSC - Departamento de Sistemas de Computação

Prof. Fernando Santos OSÓRIO

Web institucional: <http://www.icmc.usp.br/ssc/>

Página pessoal: <http://www.icmc.usp.br/~fosorio/>

E-mail: fosorio [at] icmc. usp. br ou fosorio [at] gmail. com

Disciplina de Organização de Computadores I / Eng. Comp.

Estagiário PAE: Maurício A. Dias

Web disciplina: <http://wiki.icmc.usp.br/index.php/Ssc-610>

> Programa, Material de Aulas, Critérios de Avaliação,

> Lista de Exercícios, Trabalhos Práticos, Datas das Provas