USP - ICMC - SSC

SSC 0511 - Sist. Informação - 2o. Semestre 2014

Disciplina de Organização de Computadores Digitais

Prof. Fernando Santos Osório

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

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

Material on-line: Wiki ICMC:

http://wiki.icmc.usp.br/index.php/SSC-511-2014(fosorio)

IRM
Lab. de Robótica Móvel
CROB
Centro de Robótica da USP

Aula 01s

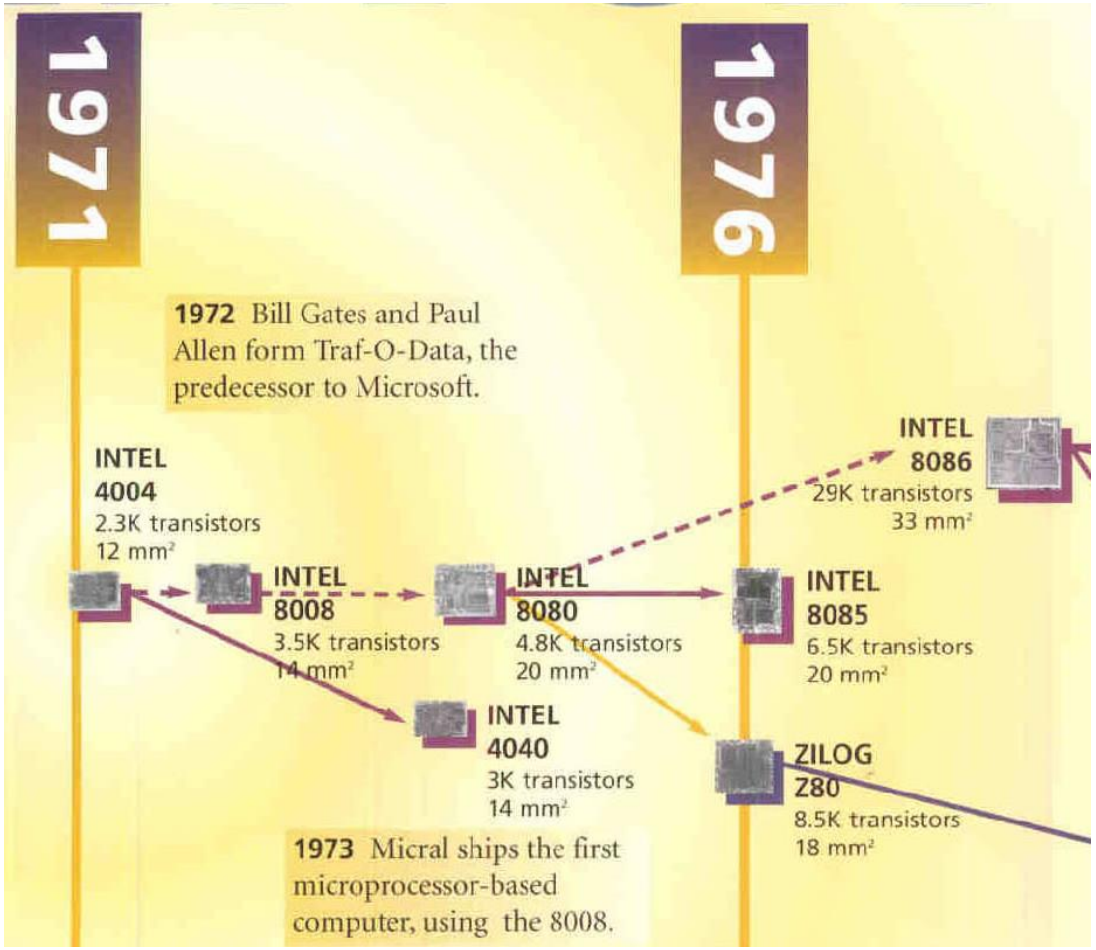
Organização de Computadores

Agenda:

1. Introdução a Organização de Computadores:
   Componentes e Blocos Básicos
2. Arquitetura de Von Neumann
3. Lógica Digital – Conceitos Básicos [Revisão SS0512]
   - Sistemas Numéricos
   - Circuitos combinacionais sequenciais
4. Construindo uma ULA
   - Somadores
   - Representação Numérica (Inteiros)
   - Operadores, Operações & Bits de Controle/Status

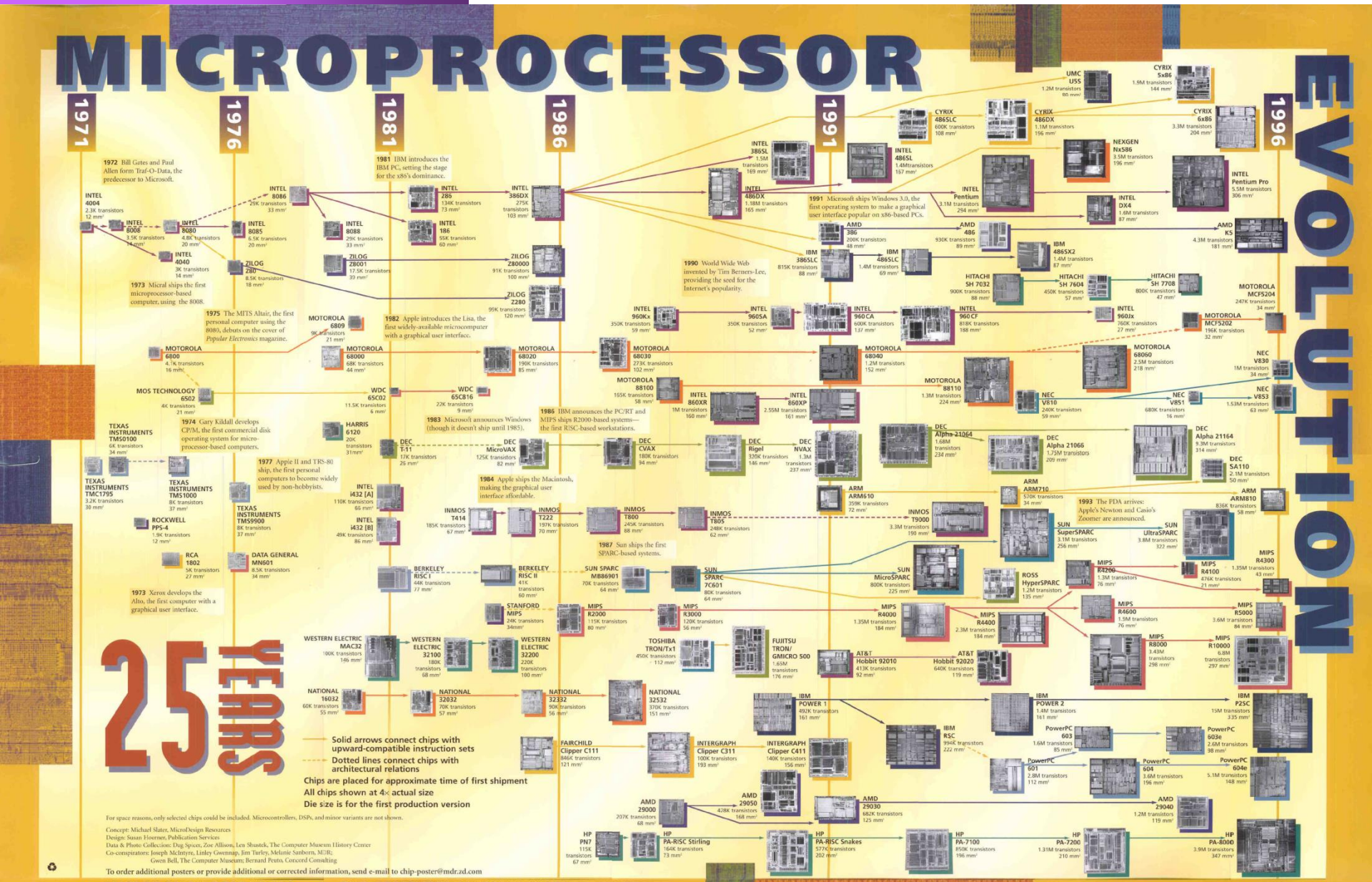
Arquitetura e Organização de Computadores

Motivação...



Origem, História, Evolução
Modelos, Conceitos

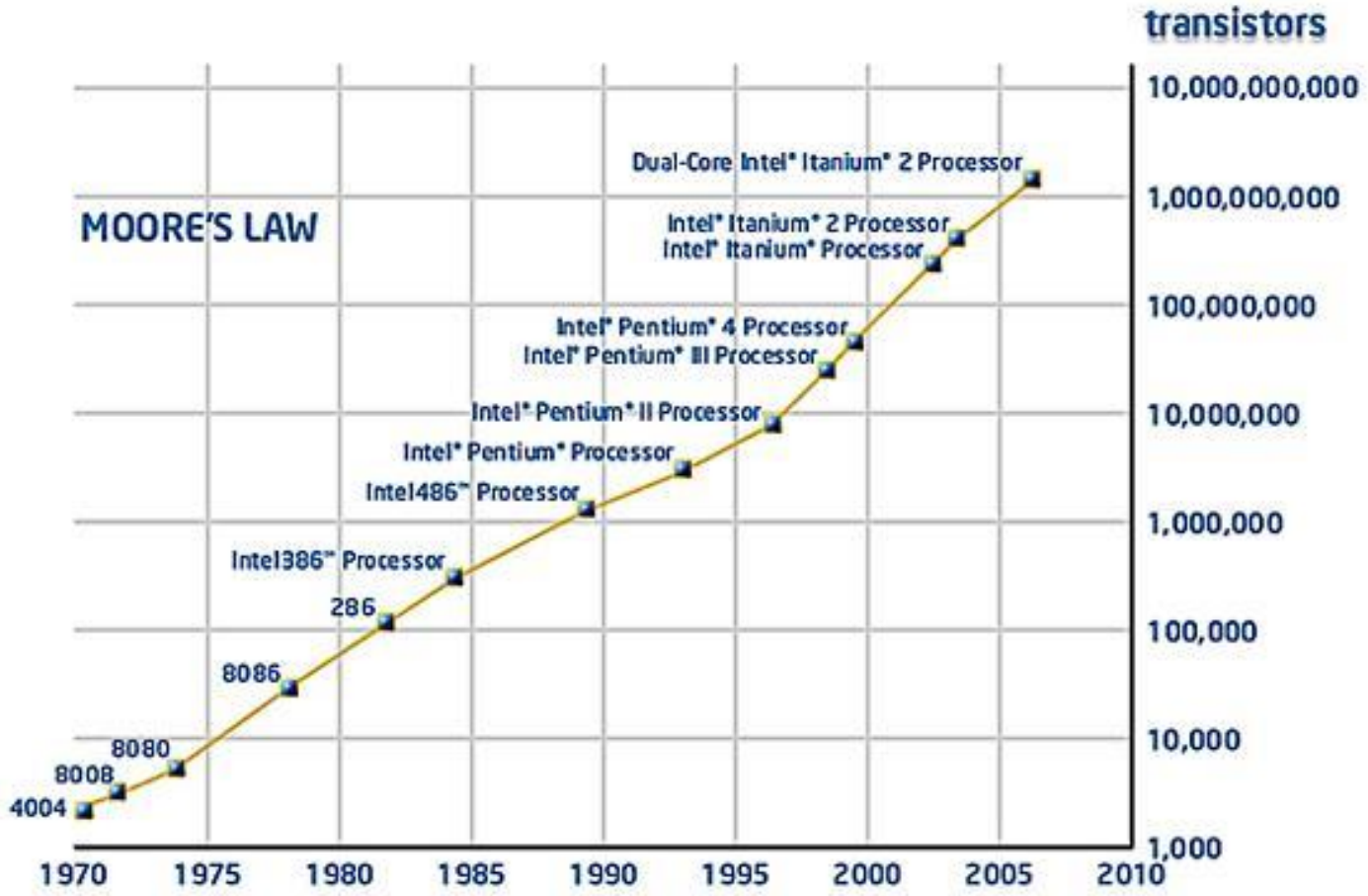
Arquitetura e Organização de Computadores



Fonte: http://research.microsoft.com/~GBell/CyberMuseum_contents/Microprocessor_Evolution_Poster.jpg

Arquitetura e Organização de Computadores

Evolução...



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

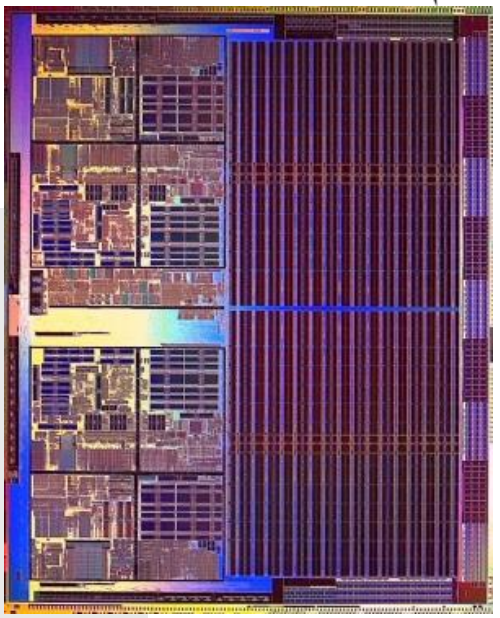
Arquitetura e Organização de Computadores

Evolução...

Select by chip logo

CPUs 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	
NCR	Sun Microsystems
NEC	Synertek
NexGen	Texas Instruments
OKI	Thomson
Performance Semi	Toshiba
Philips	ULSI
QED	UMC
Rise	VIA
Rockwell	VLSI Technology
SGS	WDC
SGS-Thomson	Weitek
Siemens	Zilog
Signetics	
Sony	

Select by brand label



Athlon64-x2

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

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 => COMPUTADOR DIGITAL

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

    Exemplo: 2014 = 2* 10^3 + 0*10^2 + 1*10 + 4*1

- Sistema binário
  - 2 dígitos: 0 1
  - Casas binárias: .... 2^3 2^2 2 1

    Exemplo: 0101 1111 = 0*128 + 1*64 + 0*32 + 1*16 + 1*8 + 1* 4 + 1*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
  - Exemplo: 5F = 0101 1111 = 5*16 + 15*1 = 95

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

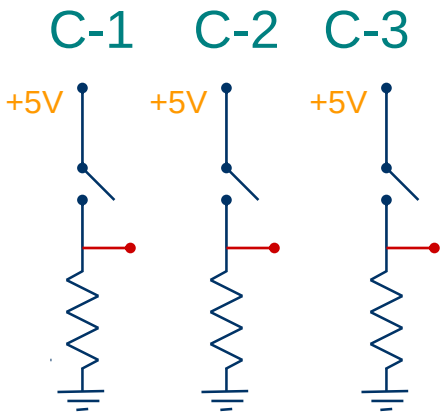
1 0 1 0 1 0 1 0 => ?

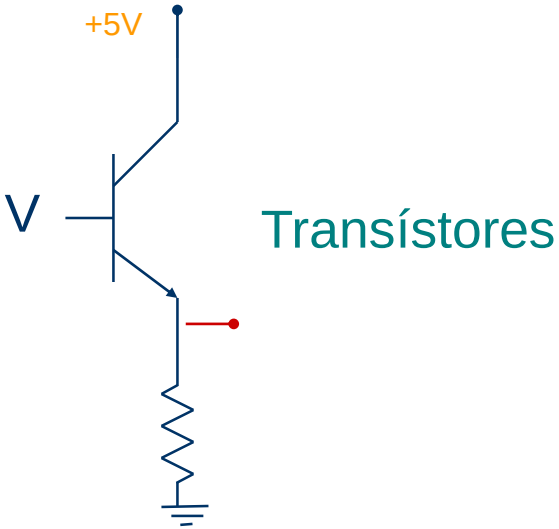
8 bits, 16 bits, 32 bits - MSB / LSB

Big-endian / Little-endian

FF00h ou 00FFh ?

LÓGICA BINÁRIA

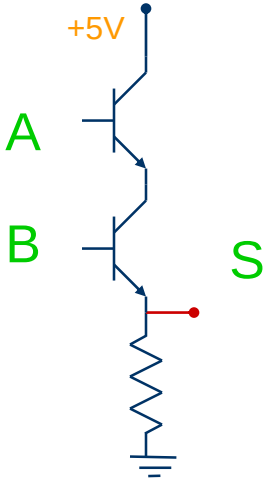




Transístores

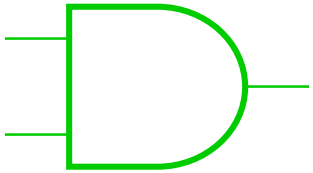
C-1	C-2	C-3	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

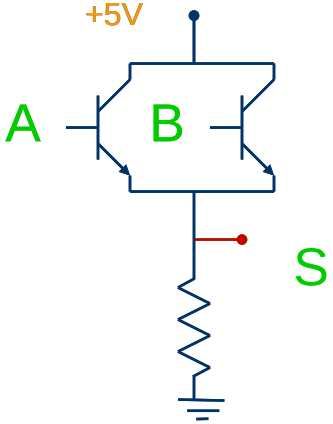
CIRCUITOS LÓGICOS



E (AND)

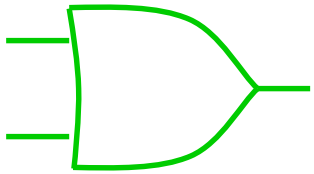
B \ A	0	1
0	0	0
1	0	1





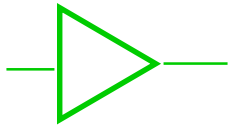
OU (OR)

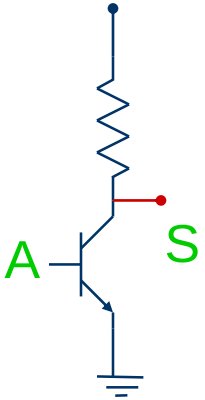
B \ A	0	1
0	0	1
1	1	1



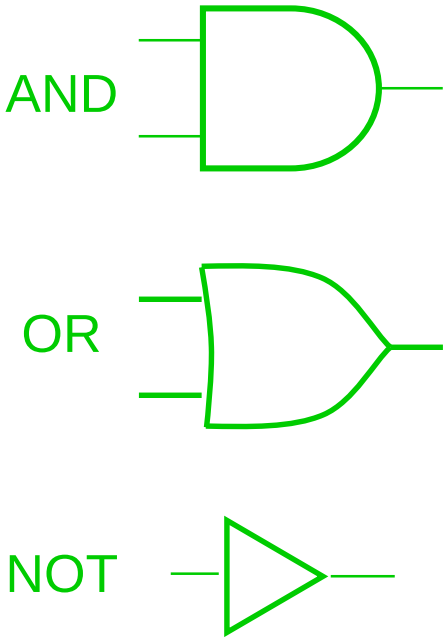
NÃO (NOT)

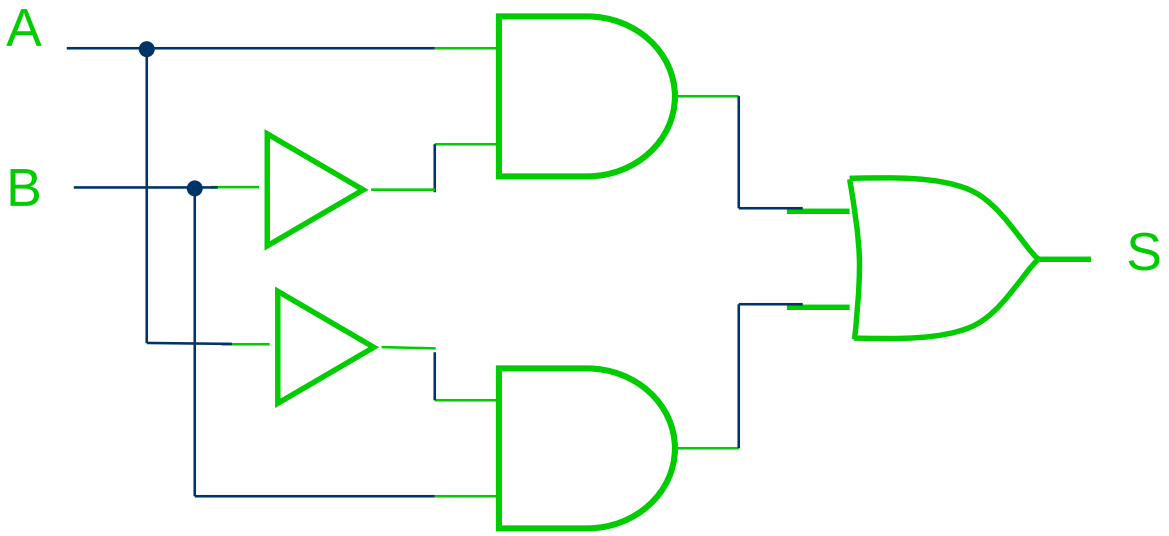
A	0	1
	1	0





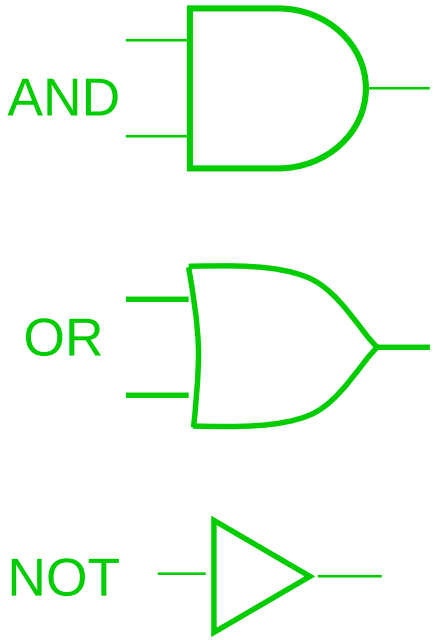
CIRCUITOS LÓGICOS

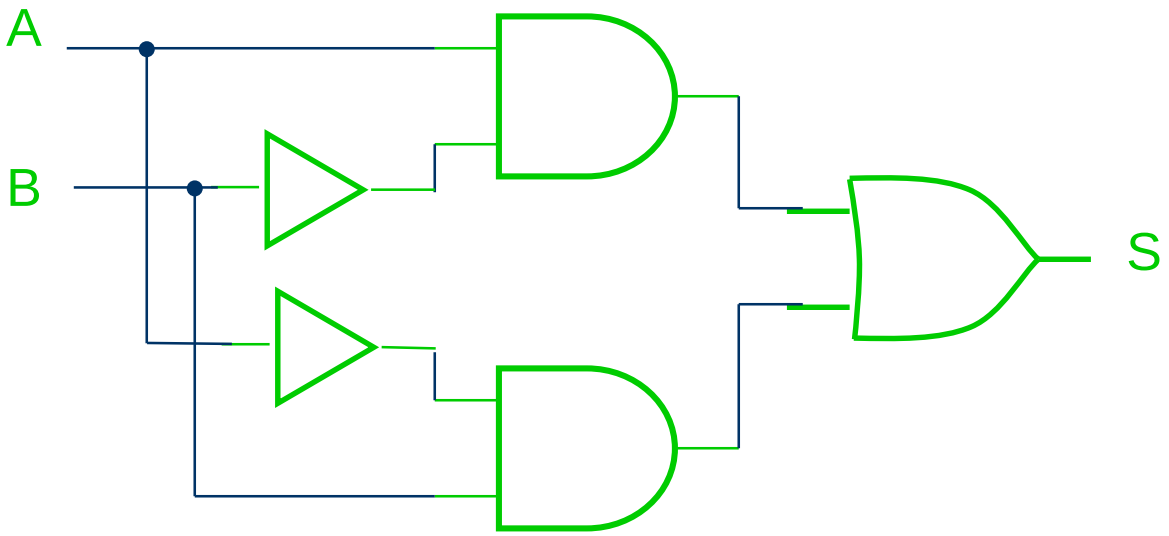




B \ A	0	1
0		
1		

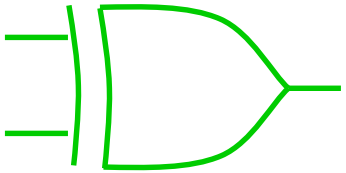
CIRCUITOS LÓGICOS





B \ A	0	1
0	0	1
1	1	0

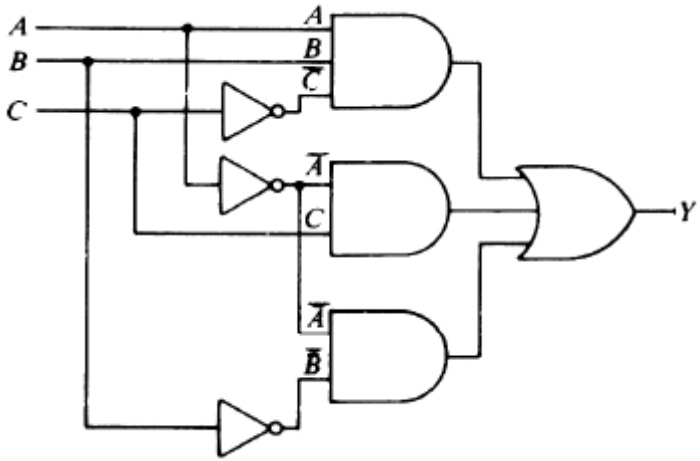
ou exclusivo



XOR

Circuitos Combinacionais Seqüenciais

Circuito Combinacional



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

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

Ref.: Idoeta/Capuano. Elementos de Eletrônica Digital

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**

Arquitetura de Von Neumann

Arquitetura - Modelo Inicial:

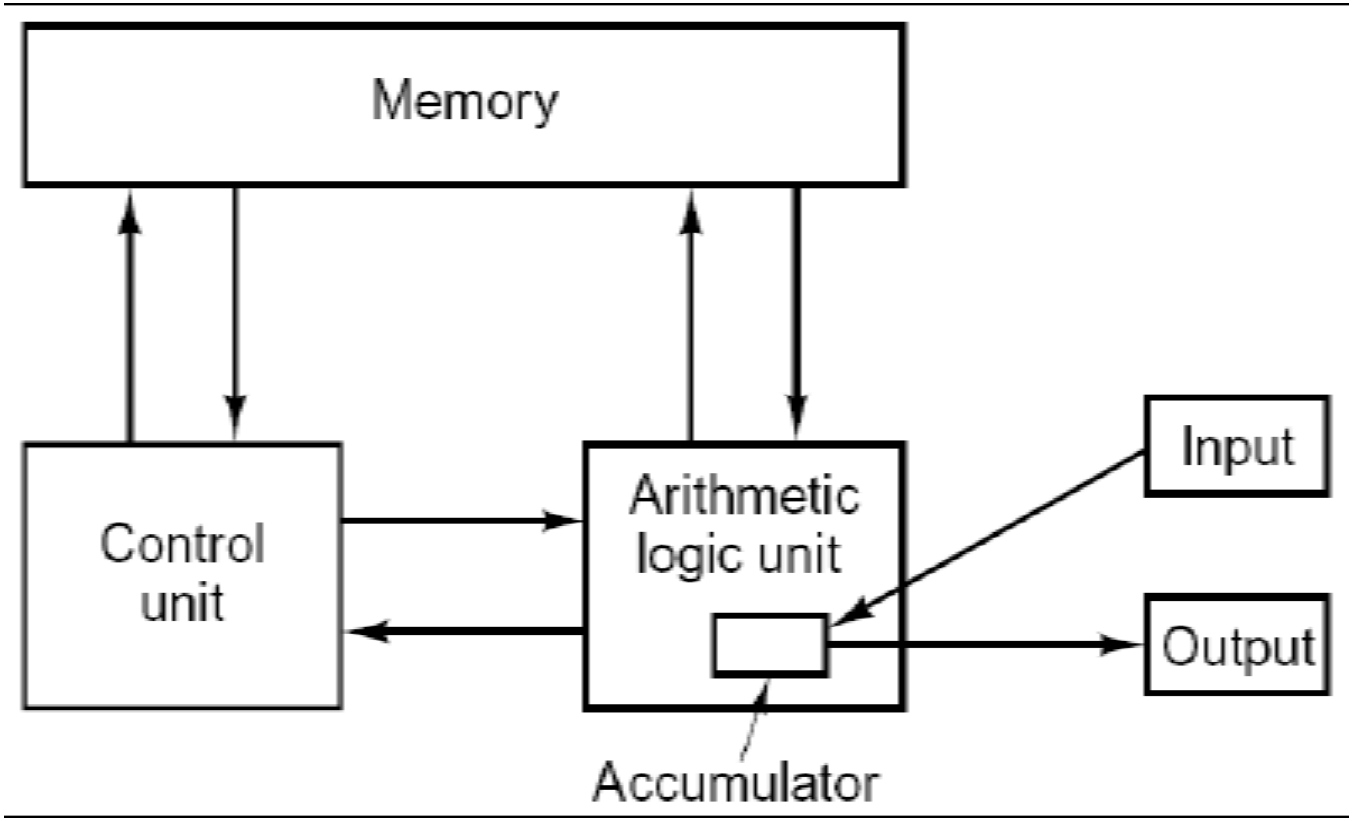
- John von Neumann
  - Construiu em 1952 o computador IAS (*Institute for Advanced Study* – Princeton, USA)
  - Programa Armazenado: programas e dados representados de forma digital em memória
  - Processamento baseado em aritmética binária, ao invés de decimal

- 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.

Arquitetura de Von Neumann

Arquitetura - Modelo Inicial:

- Máquina de von Neumann



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

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

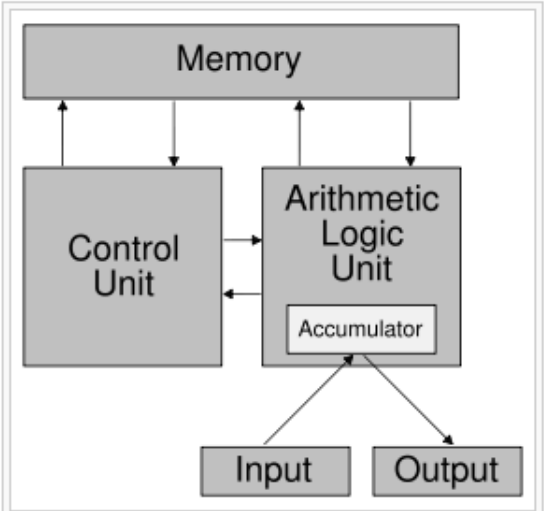
Von Neumann architecture

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.

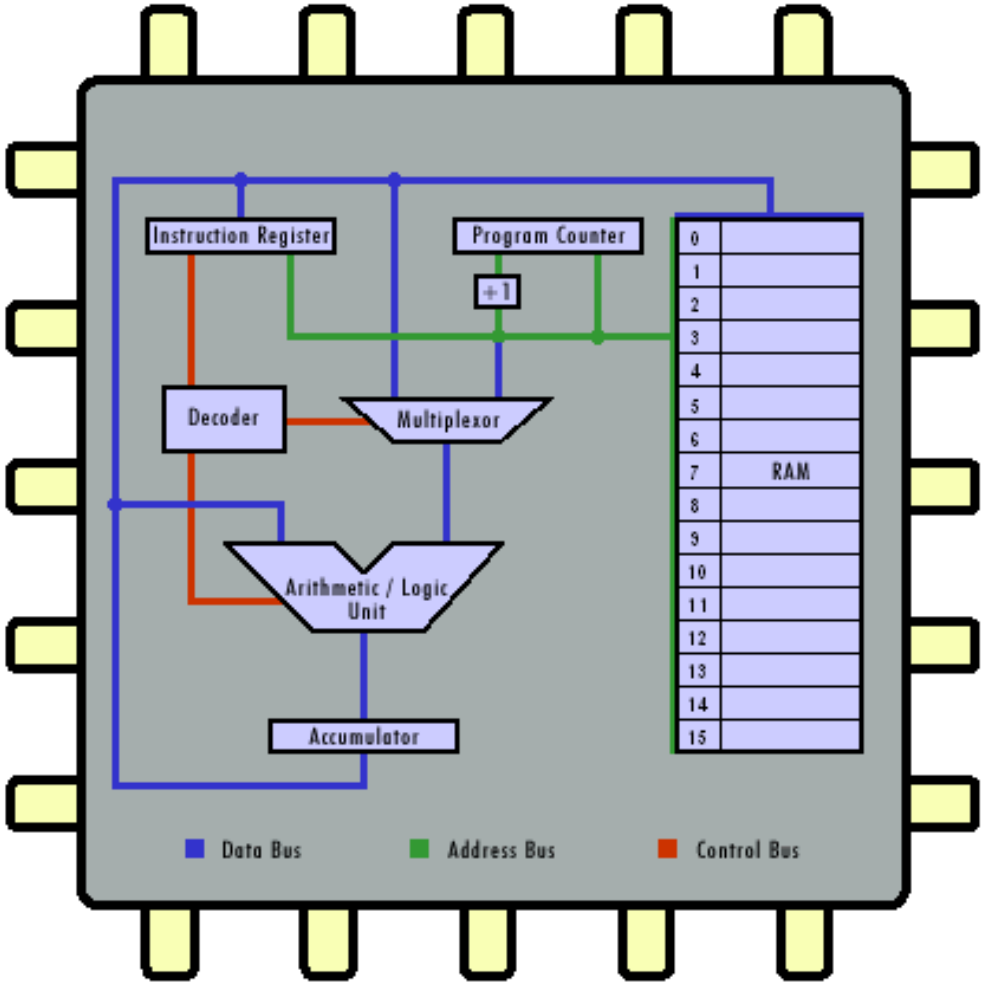


Design of the von Neumann architecture

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

Arquitetura de Von Neumann

CPU – Processador

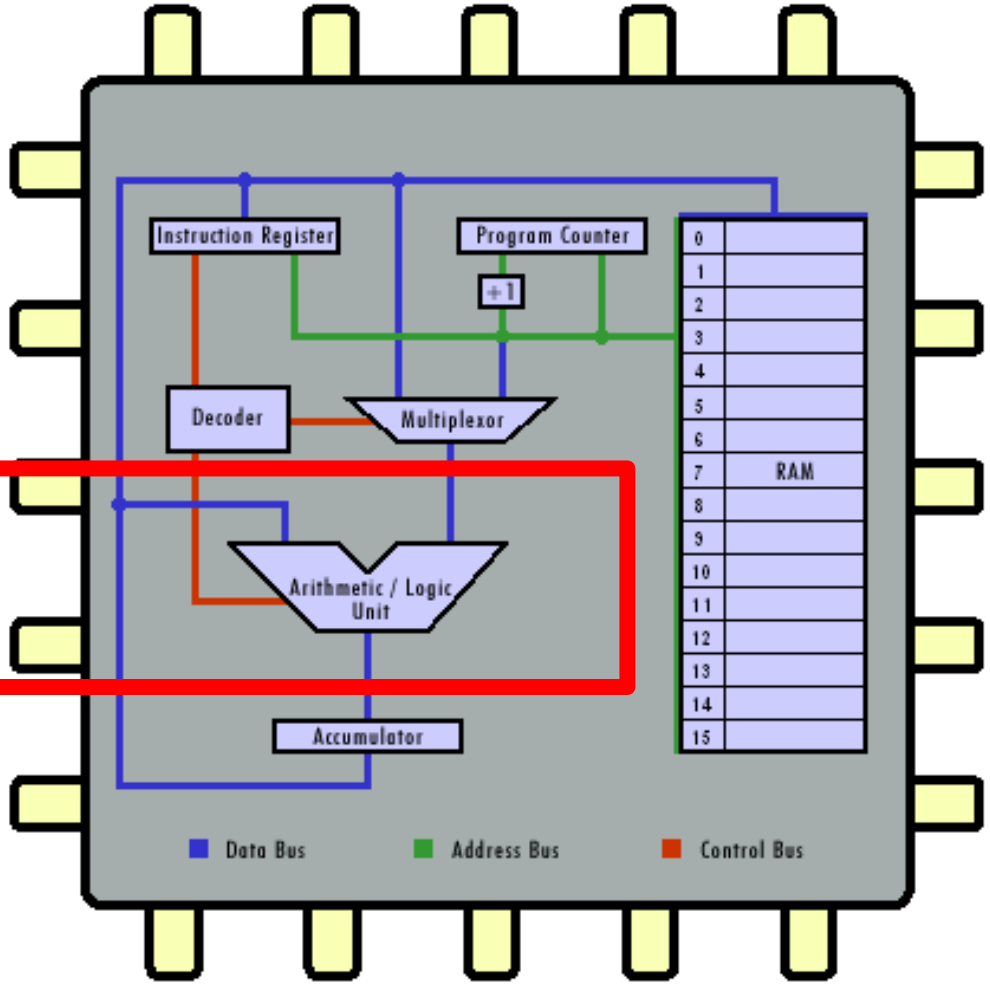


Fonte: http://courses.cs.vt.edu/~csonline/MachineArchitecture/Lessons/CPU/index.html

Arquitetura de Von Neumann

CPU – Processador

ULA / ALU
Unidade Lógico-Aritmética

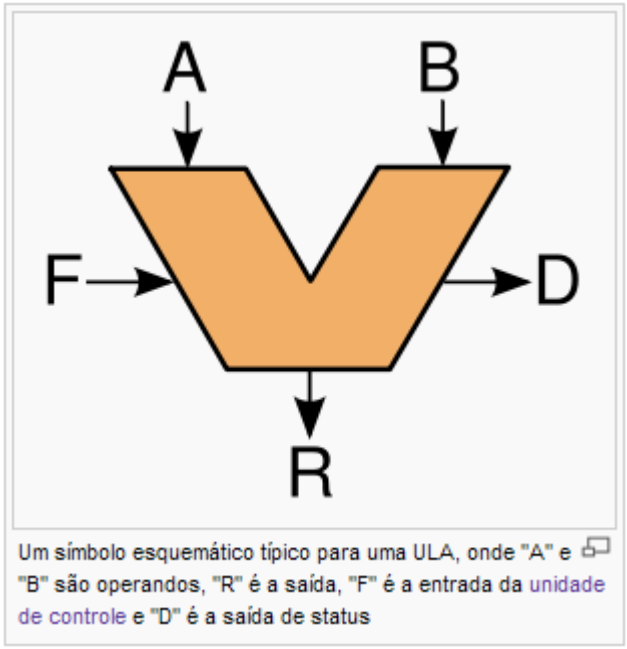


Fonte: http://courses.cs.vt.edu/~csonline/MachineArchitecture/Lessons/CPU/index.html

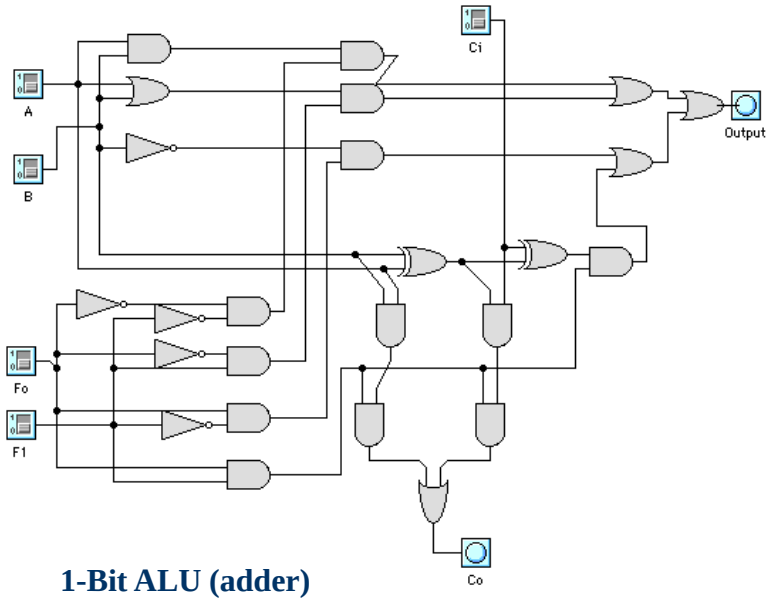
Unidade Lógica e Aritmética (ULA)

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.

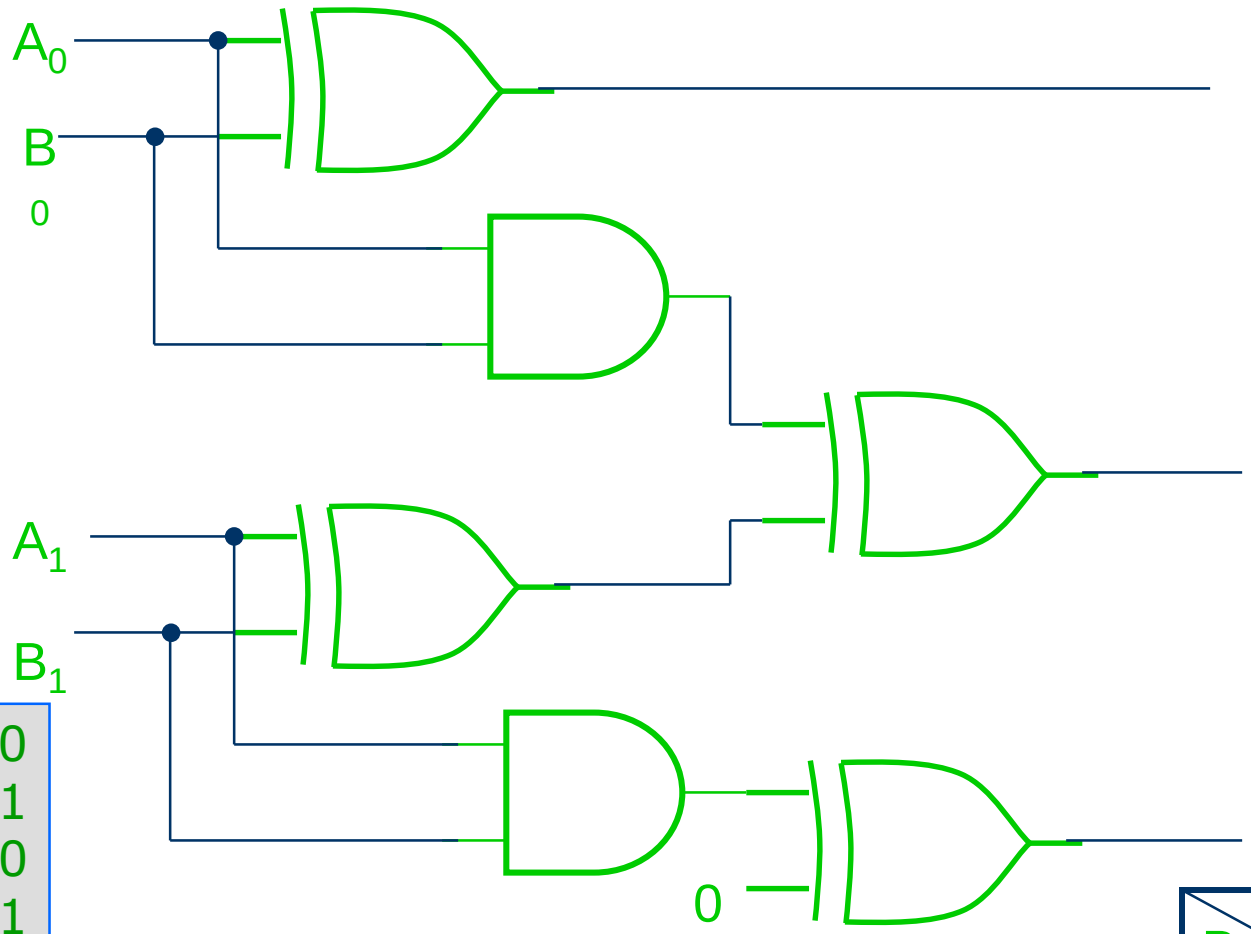


Um símbolo esquemático típico para uma ULA, onde "A" e "B" são operandos, "R" é a saída, "F" é a entrada da unidade de controle e "D" é a saída de status



1-Bit ALU (adder)

SOMADOR



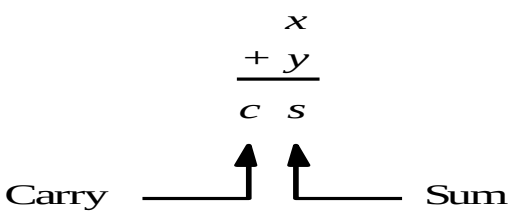
Decimal: 2
Binário: 10
+
Decimal: 3
Binário: 11

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

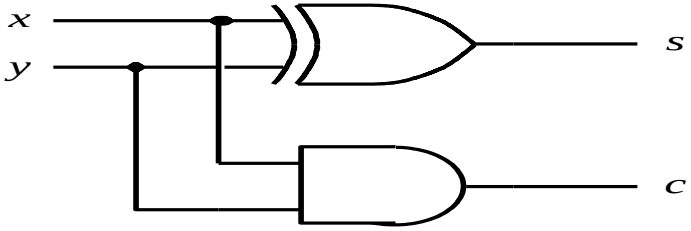


0	0	1	1
+ 0	+ 1	+ 0	+ 1
0 0	0 1	0 1	1 0

(a) The four possible cases

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

(b) Truth table



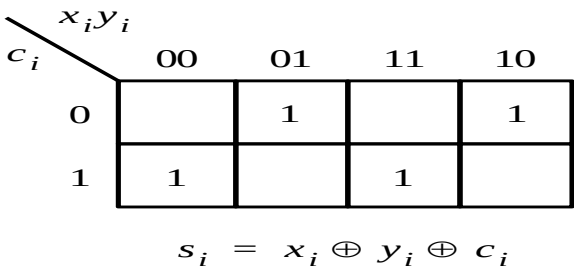
(c) Circuit

(d) Graphical symbol

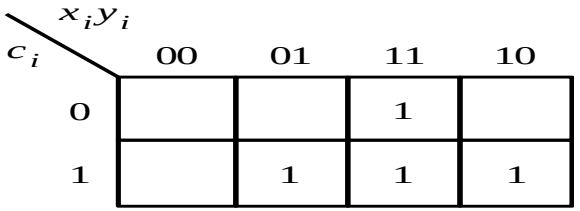
SOMADOR: Full Adder

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

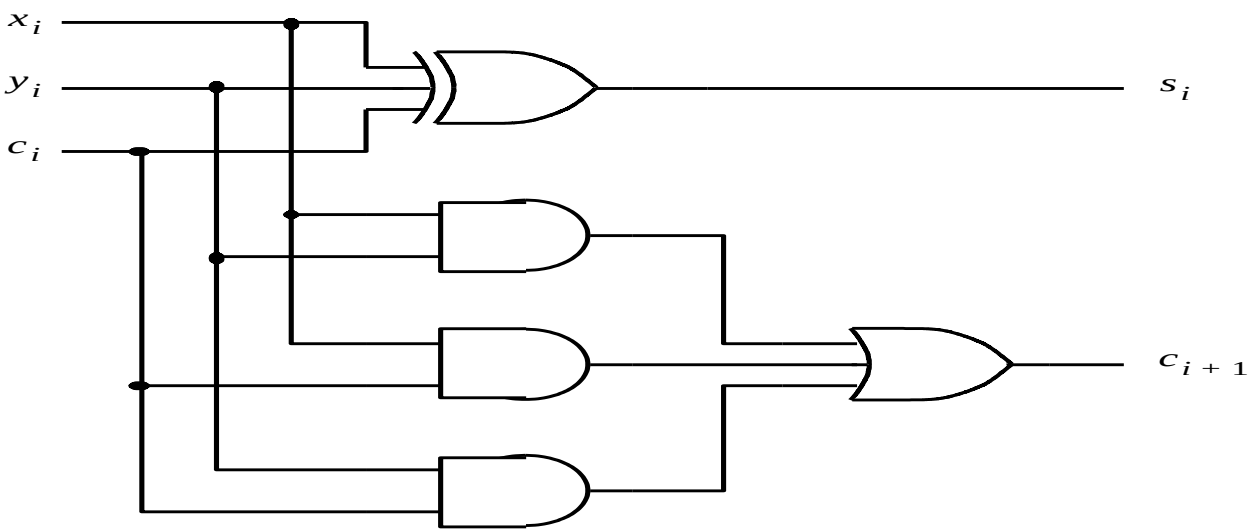


$$s_i = x_i \oplus y_i \oplus c_i$$



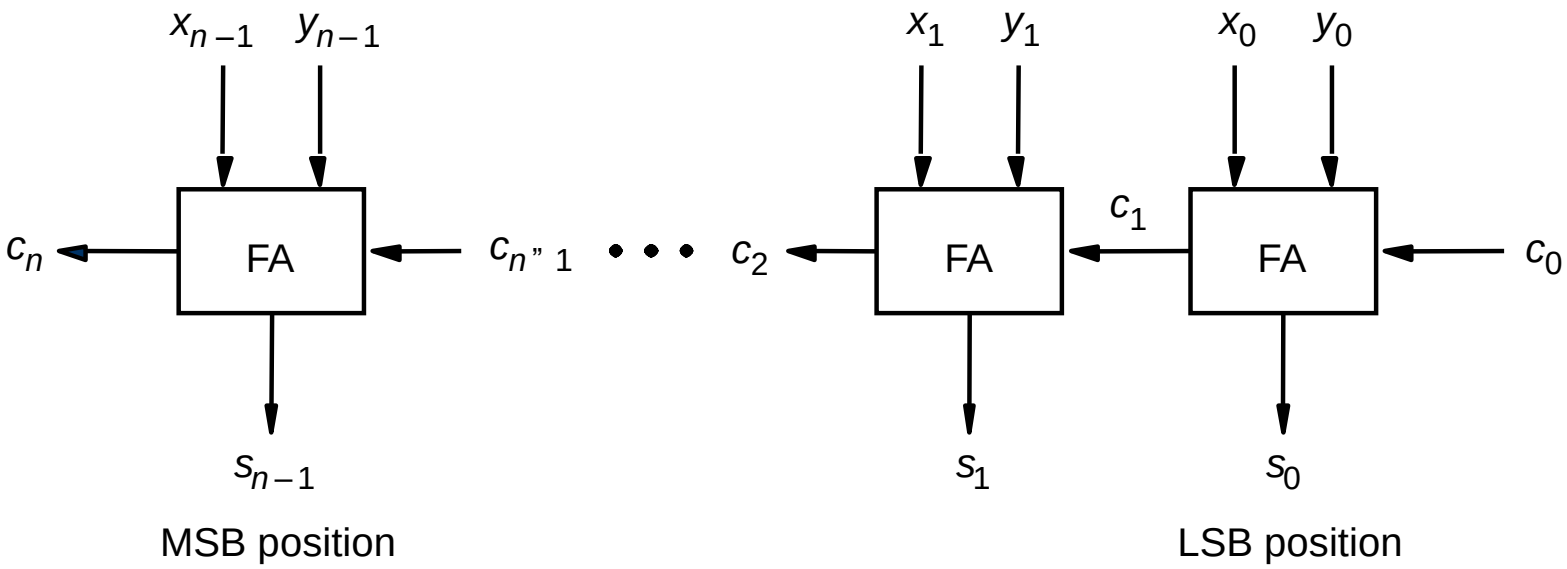
$$c_{i+1} = x_iy_i + x_ic_i + y_ic_i$$

(b) Karnaugh maps



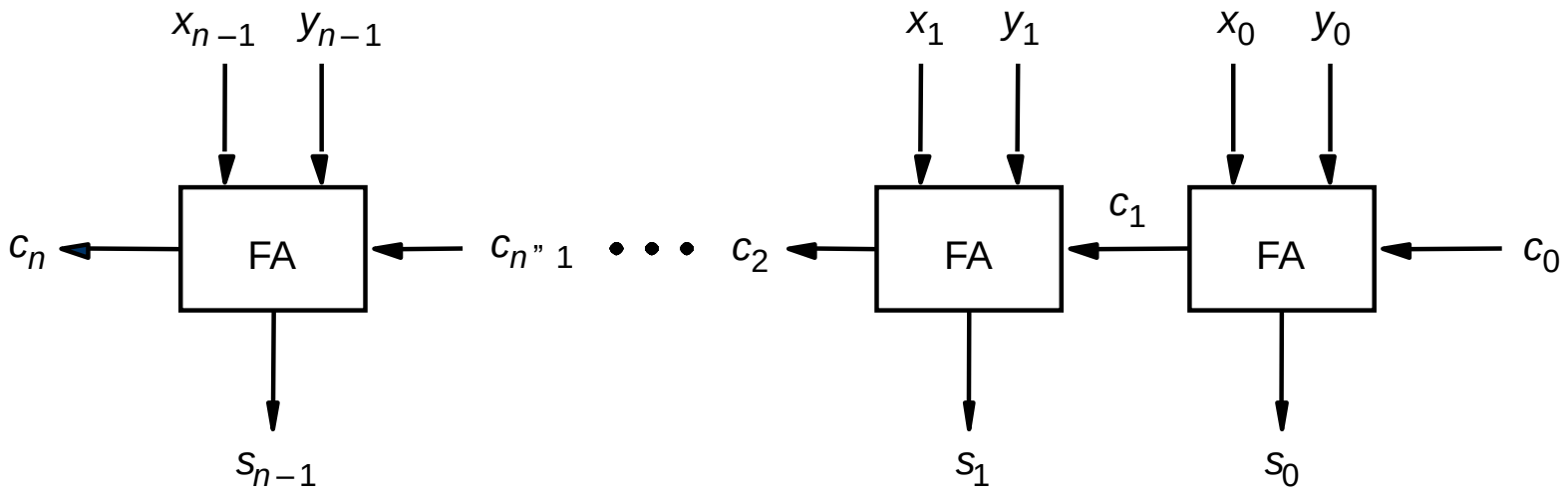
(c) Circuit

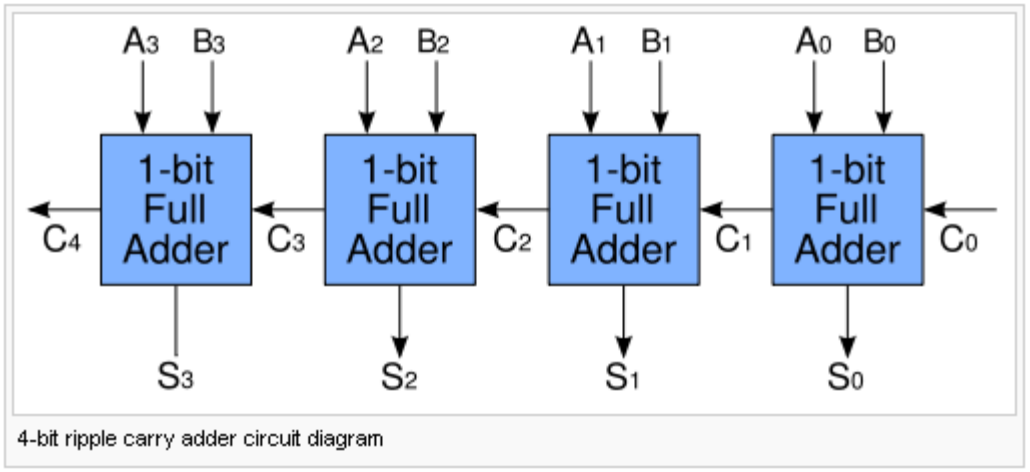
SOMADOR: n-bit Adder



An *n*-bit ripple-carry adder

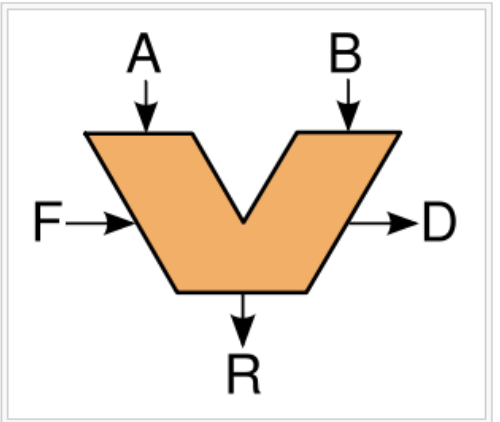
SOMADOR: n-bit Adder





4-bit ripple carry adder circuit diagram

...



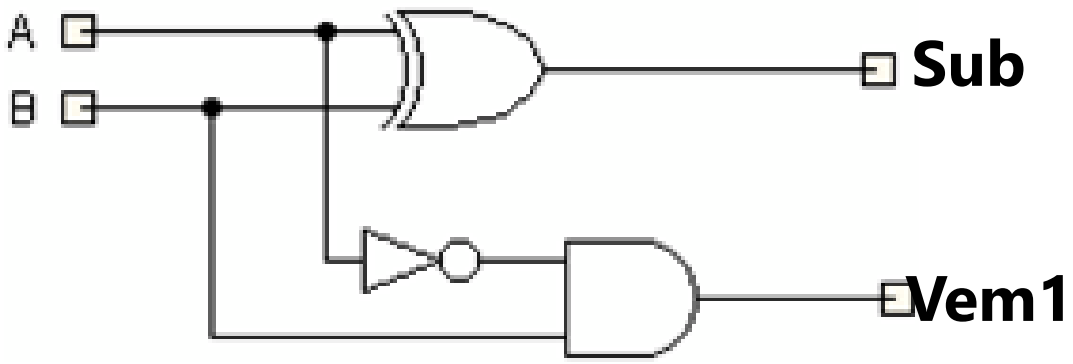
A typical schematic symbol for an ALU: A & B are the data (registers); R is the output; F is the Operand (instruction) from the Control Unit; D is an output status

SUBTRATOR

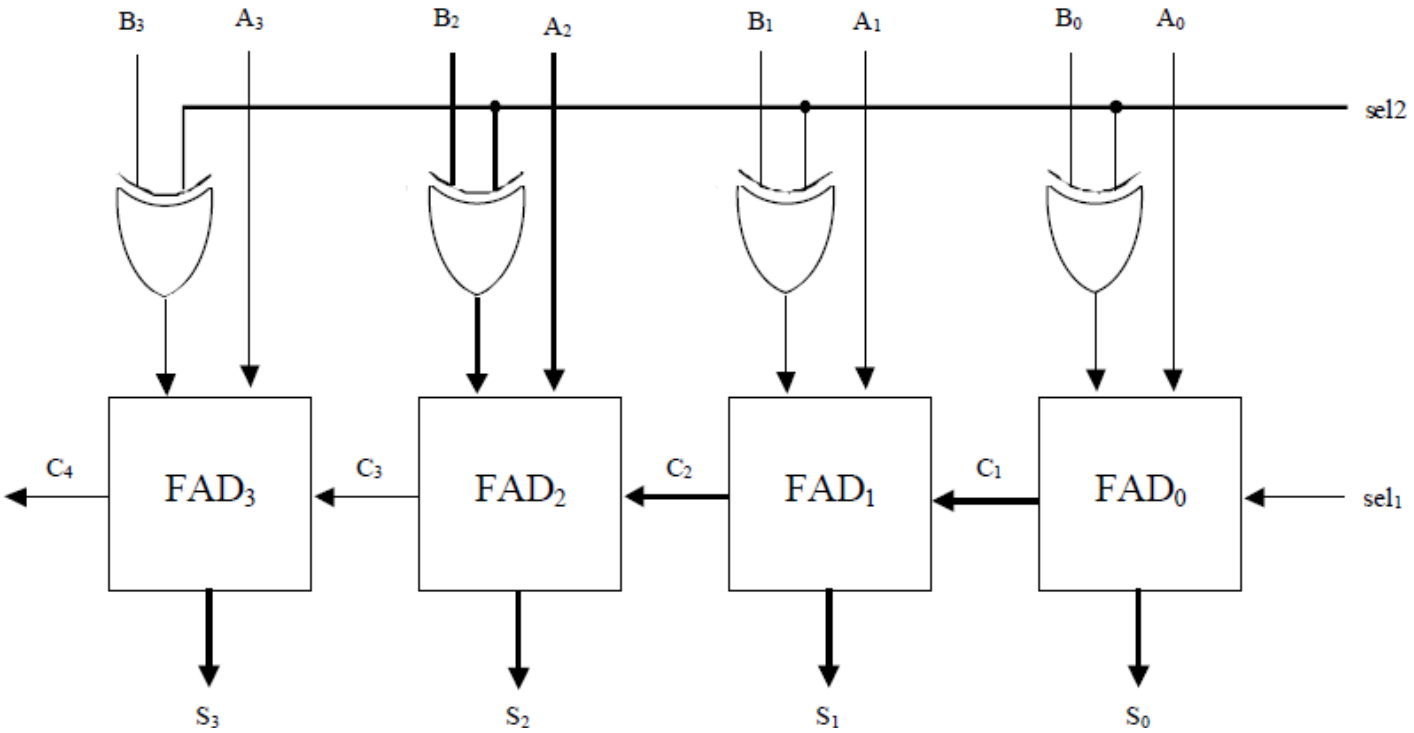
Subtração entre 2 bits: meio-subtrator

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

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



SOMA e SUBTRAÇÃO



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 ($S = A - B$)

SOMA e SUBTRAÇÃO

Representação Numérica: Complemento de 2

Dados representados em complemento de 1: Inverte bits
Dados representados em complemento de 2: Inverte + 1

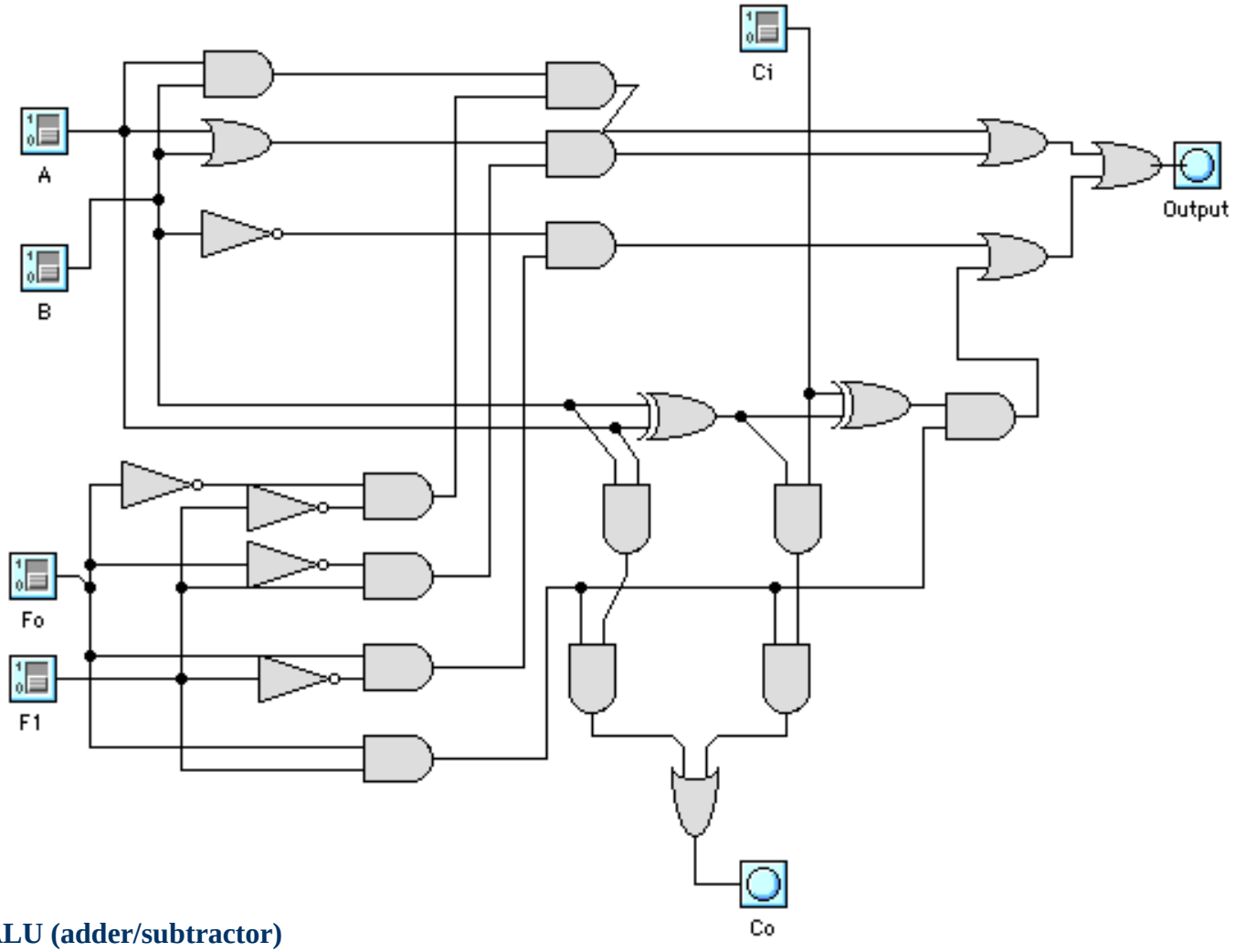
Valores Negativos:
-1 => 0001 em C2 => 1111

Subtração:
8 - 3 = 8 + (-3) = 1000 + 1101 = 0101
3 - 5 = 3 + (-5) = 0011 + 1011 = 1110

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 ($S = A - B$)

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

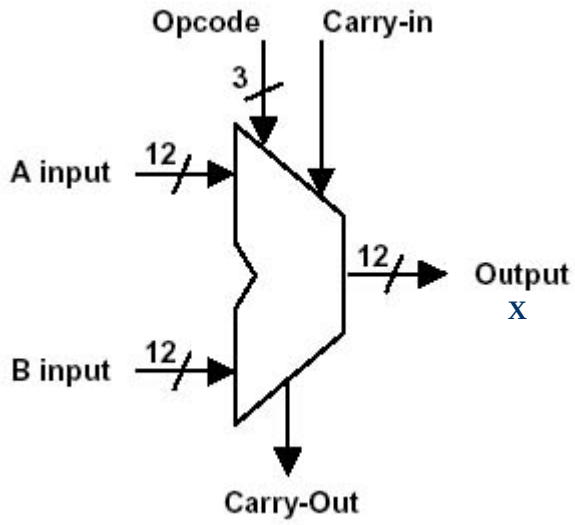
ULA: SOMA e SUBTRAÇÃO



1-Bit ALU (adder/subtractor)

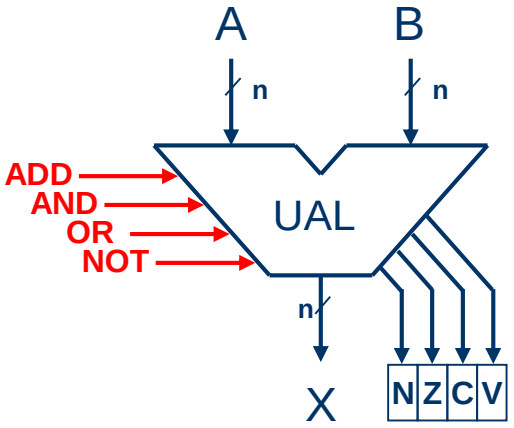
Unidade Lógica e Aritmética (ULA)

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)

Unidade Lógica e Aritmética (ULA)

Unidade Lógico Aritmética - ULA

Exemplo: TTL 74181 ALU

Signetics

74181, LS181, S181
Arithmetic Logic Units

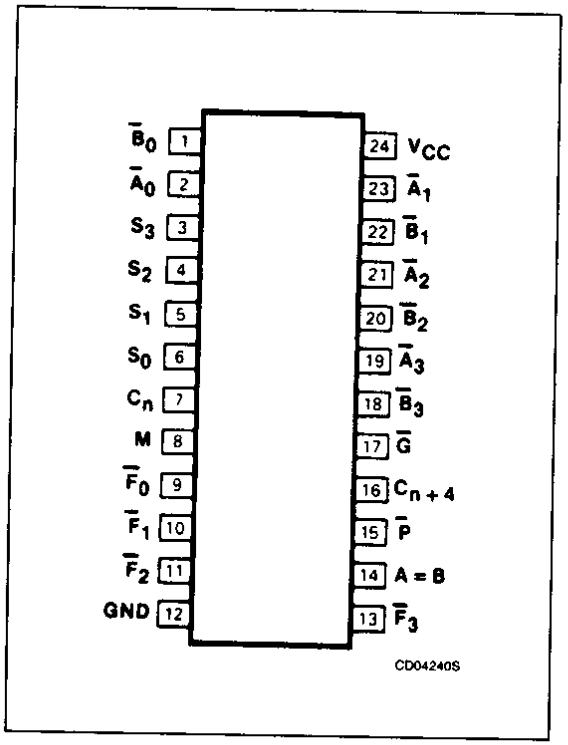
4-Bit Arithmetic Logic Unit
Product Specification

Logic Products

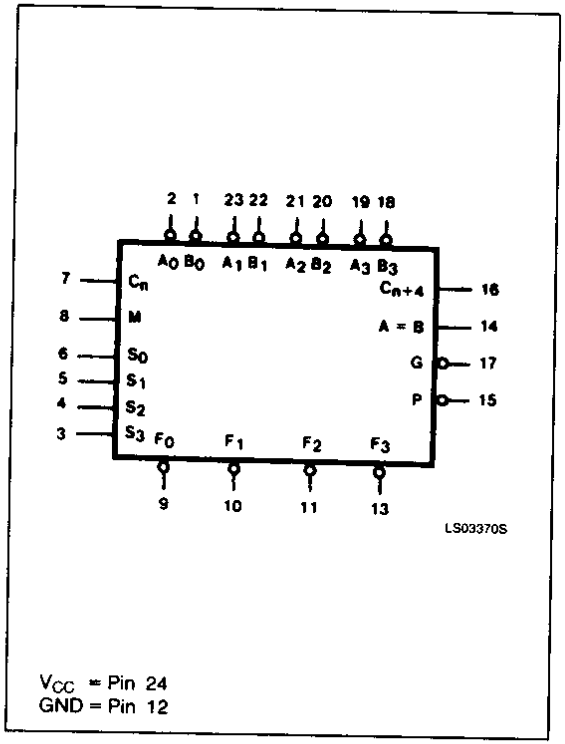
FEATURES

- 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



LOGIC SYMBOL



December 4, 1985

5-350

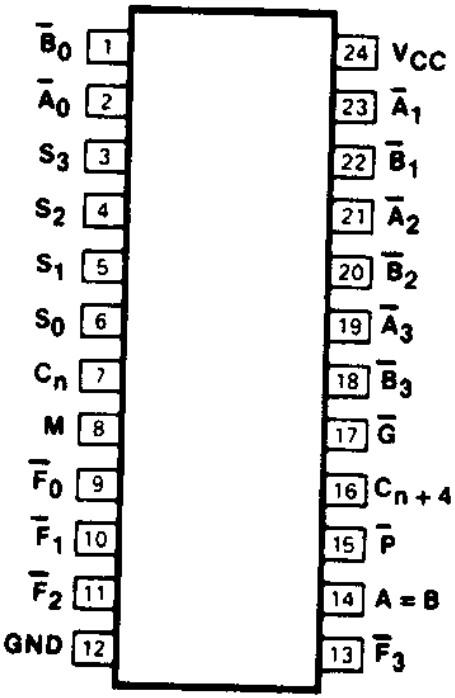
Unidade Lógica e Aritmética (ULA)

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	$\overline{A}$	A
L	L	L	H	$\overline{A + B}$	A + B
L	L	H	L	$\overline{A}B$	A + $\overline{B}$
L	L	H	H	Logical 0	minus 1
L	H	L	L	$\overline{AB}$	A plus A$\overline{B}$
L	H	L	H	$\overline{B}$	(A + B) plus A$\overline{B}$
L	H	H	L	A ⊕ B	A minus B minus 1
L	H	H	H	A$\overline{B}$	AB minus 1
H	L	L	L	$\overline{A}$ + B	A plus AB
H	L	L	H	$\overline{A \cdot B}$	A plus B
H	L	H	L	B	(A + $\overline{B}$) plus AB
H	L	H	H	AB	AB minus 1
H	H	L	L	Logical 1	A plus A*
H	H	L	H	A + $\overline{B}$	(A + B) plus A
H	H	H	L	A + B	(A + $\overline{B}$) plus A
H	H	H	H	A	A minus 1



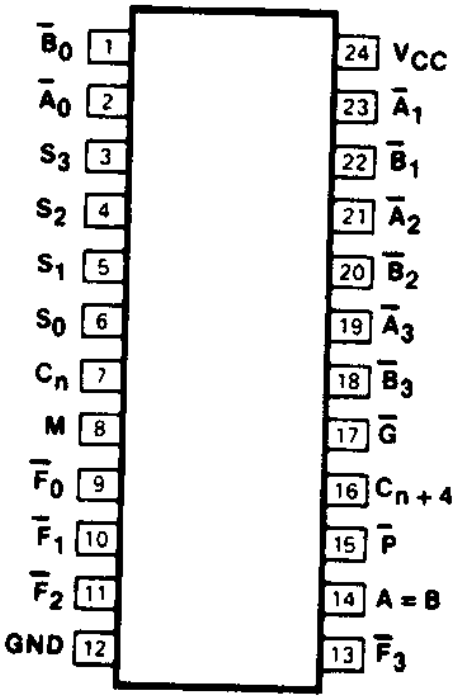
Unidade Lógica e Aritmética (ULA)

Unidade Lógico Aritmética - ULA

Exemplo: TTL 74181 ALU

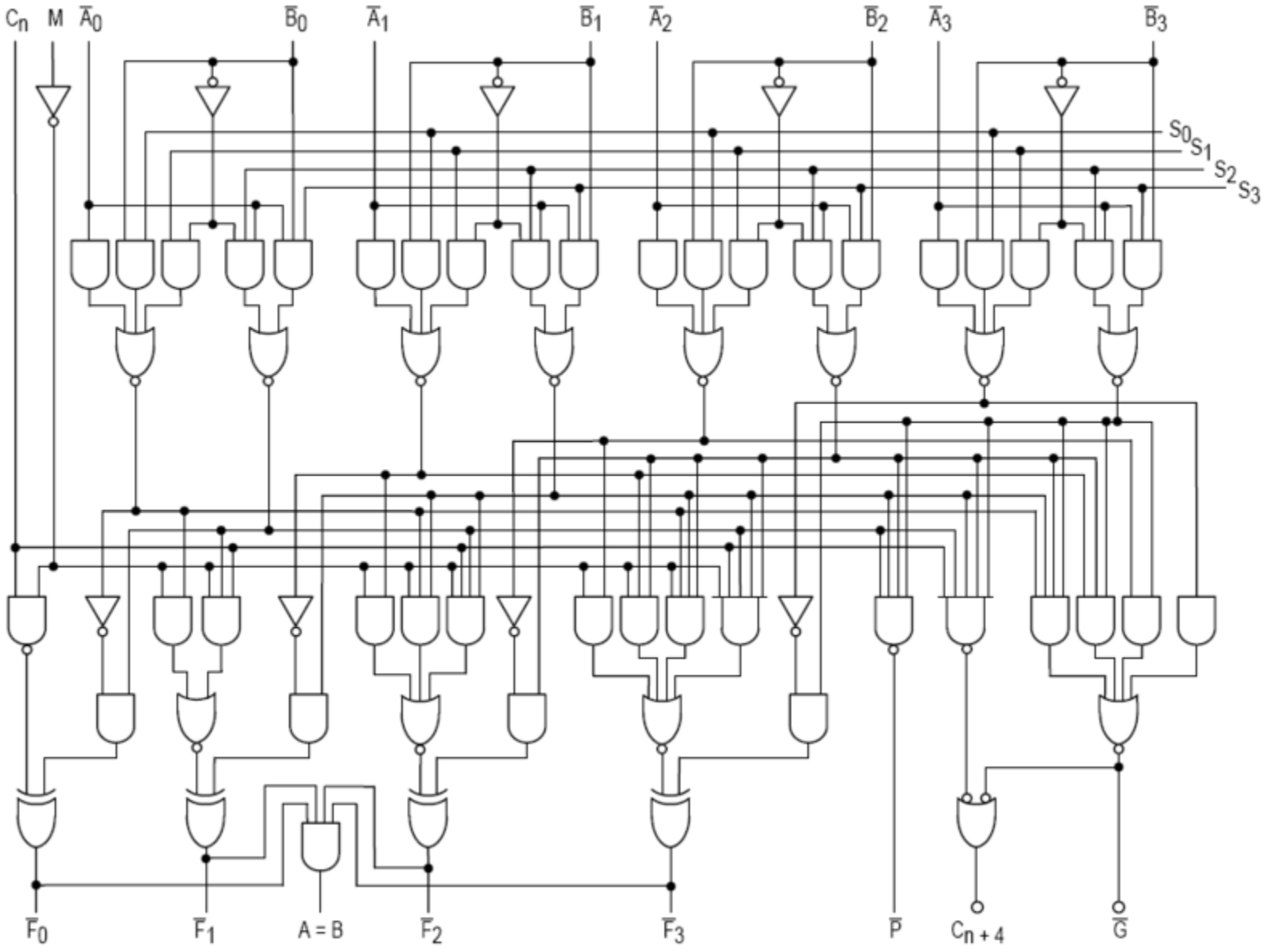
MODE SELECT INPUTS				ACTIVE LOW INPUTS & OUTPUTS	
S_3	S_2	S_1	S_0	Logic (M = H)	Arithmetic** (M = L) (C_n = L)
L	L	L	L	$\overline{A}$	A minus 1
L	L	L	H	$\overline{AB}$	AB minus 1
L	L	H	L	$\overline{A} + B$	$A\overline{B}$ minus 1
L	L	H	H	Logical 1	minus 1
L	H	L	L	$\overline{A + B}$	A plus (A + $\overline{B}$)
L	H	L	H	$\overline{B}$	AB plus (A + $\overline{B}$)
L	H	H	L	$\overline{A \bullet B}$	A minus B minus 1
L	H	H	H	$A + \overline{B}$	$A + \overline{B}$
H	L	L	L	$\overline{AB}$	A plus (A + B)
H	L	L	H	$\overline{A \bullet B}$	A plus B
H	L	H	L	B	$A\overline{B}$ (A + B)
H	L	H	H	A + B	A + B
H	H	L	L	Logical 0	A plus A*
H	H	L	H	$A\overline{B}$	AB plus A
H	H	H	L	AB	$A\overline{B}$ 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 2s complement notation.



Unidade Lógica e Aritmética (ULA)

Schematic
74181
ALU



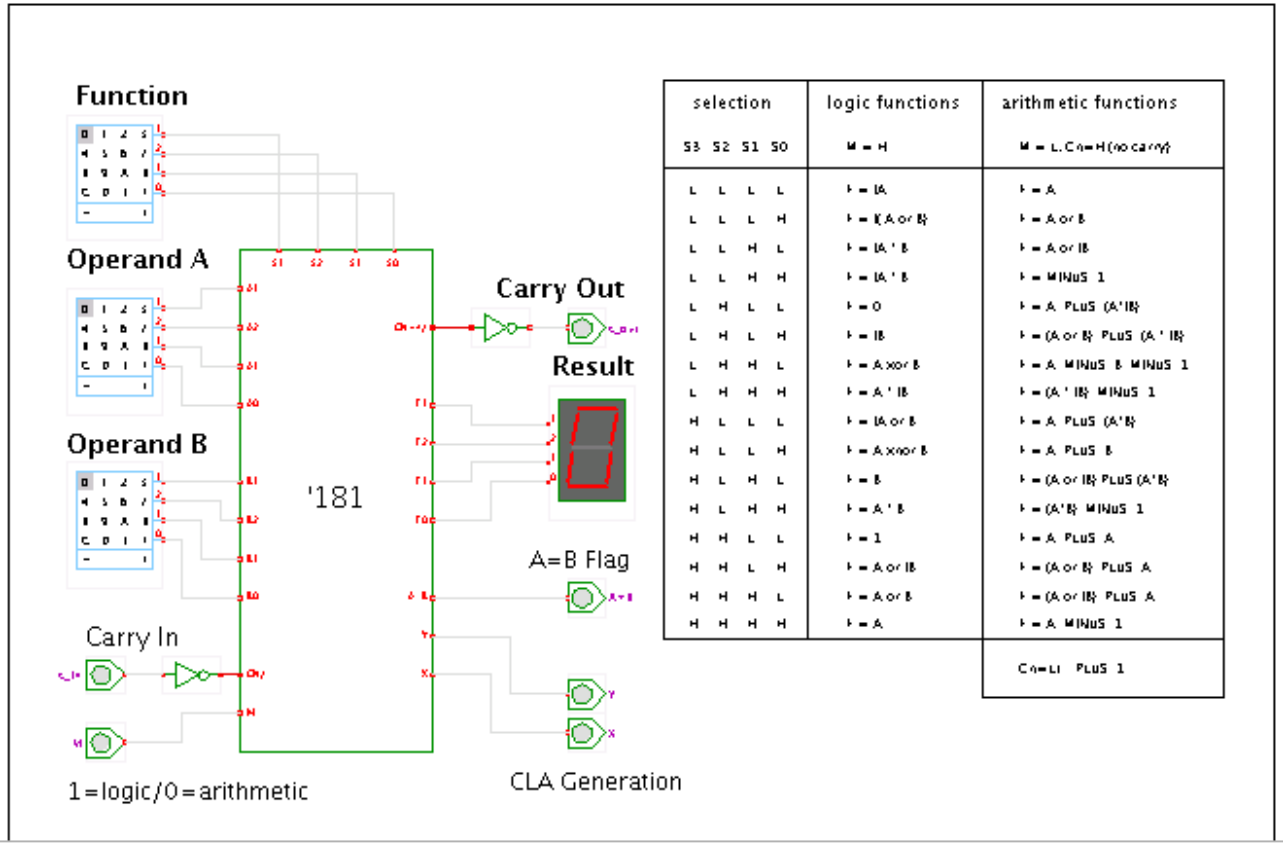
Unidade Lógica e Aritmética (ULA)

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

HADES - JAVA Intercative Demo - Logic Circuits [

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

TTL-series 74181 ALU demonstration



selection S3 S2 S1 S0	logic functions M = H	arithmetic functions M = L, Cn=H (no carry)
L L L L	F = /A	F = A
L L L H	F = /(A or B)	F = A or B
L L H L	F = /A · B	F = A or /B
L L H H	F = /A · B	F = MINUS 1
L H L L	F = 0	F = A PLUS (A·/B)
L H L H	F = /B	F = (A or B) PLUS (A · /B)
L H H L	F = A xor B	F = A MINUS B MINUS 1
L H H H	F = A · /B	F = (A · /B) MINUS 1
H L L L	F = /A or B	F = A PLUS (A·B)
H L L H	F = /A xnor B	F = A PLUS B
H L H L	F = B	F = (A or /B) PLUS (A·B)
H L H H	F = A · B	F = (A·B) MINUS 1
H H L L	F = 1	F = A PLUS A
H H L H	F = A or /B	F = (A or B) PLUS A
H H H L	F = A or B	F = (A or /B) PLUS A
H H H H	F = A	F = A MINUS 1
		Cn=L PLUS 1

Fonte: http://tams-www.informatik.uni-hamburg.de/applets/hades/webdemos/20-arithmetic/50-74181/demo-74181-ALU_print.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 Digitais / BSI
Web disciplina: Wiki ICMC - Http://wiki.icmc.usp.br
> Programa, Material de Aulas, Critérios de Avaliação,
> Lista de Exercícios, Trabalhos Práticos, Datas das Provas