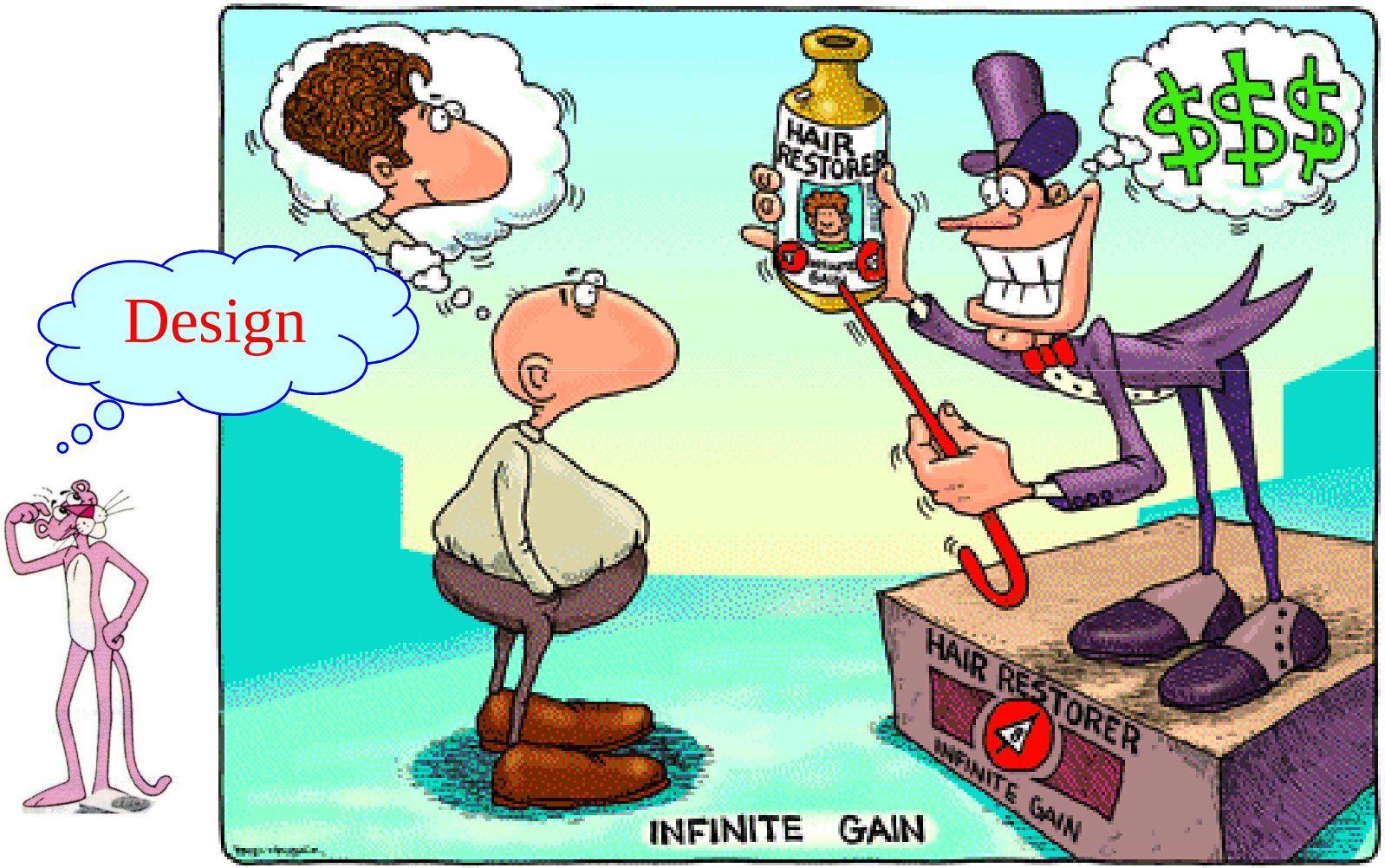
Digital System Design Course

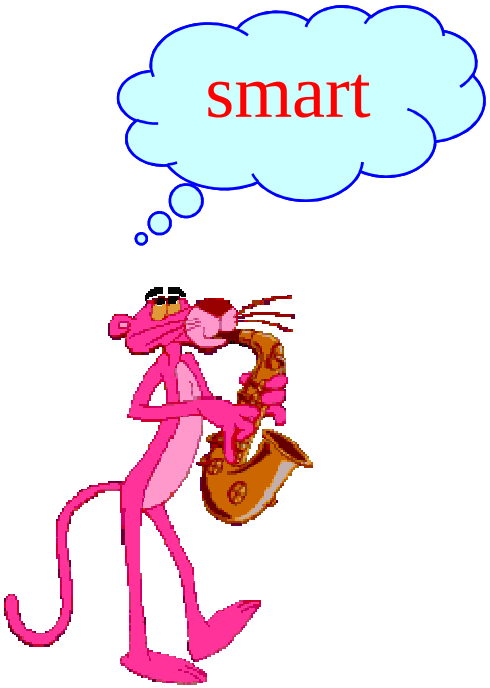
Introduction- Motivation

Sistemas Digitales II



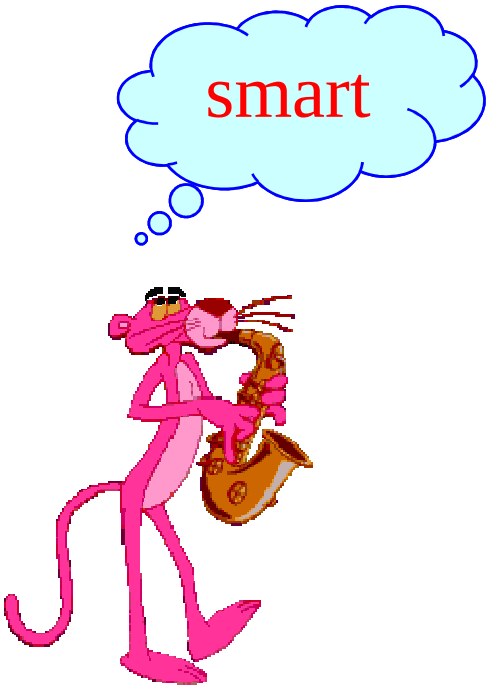
Skills

Flash…..
Smart…



$$\frac{\sqrt{\cancel{2}}}{\cancel{2}} = \sqrt{}$$

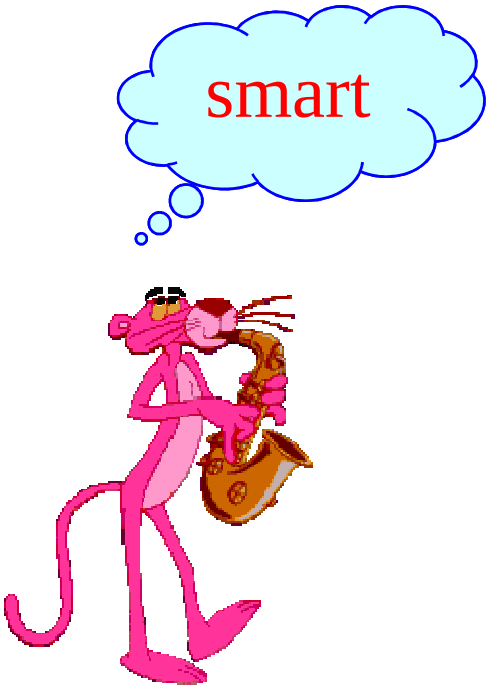
$$\frac{1}{n}\sin x = \ ?$$



$$\frac{1}{\cancel{n}}\sin\cancel{n}\, x =$$

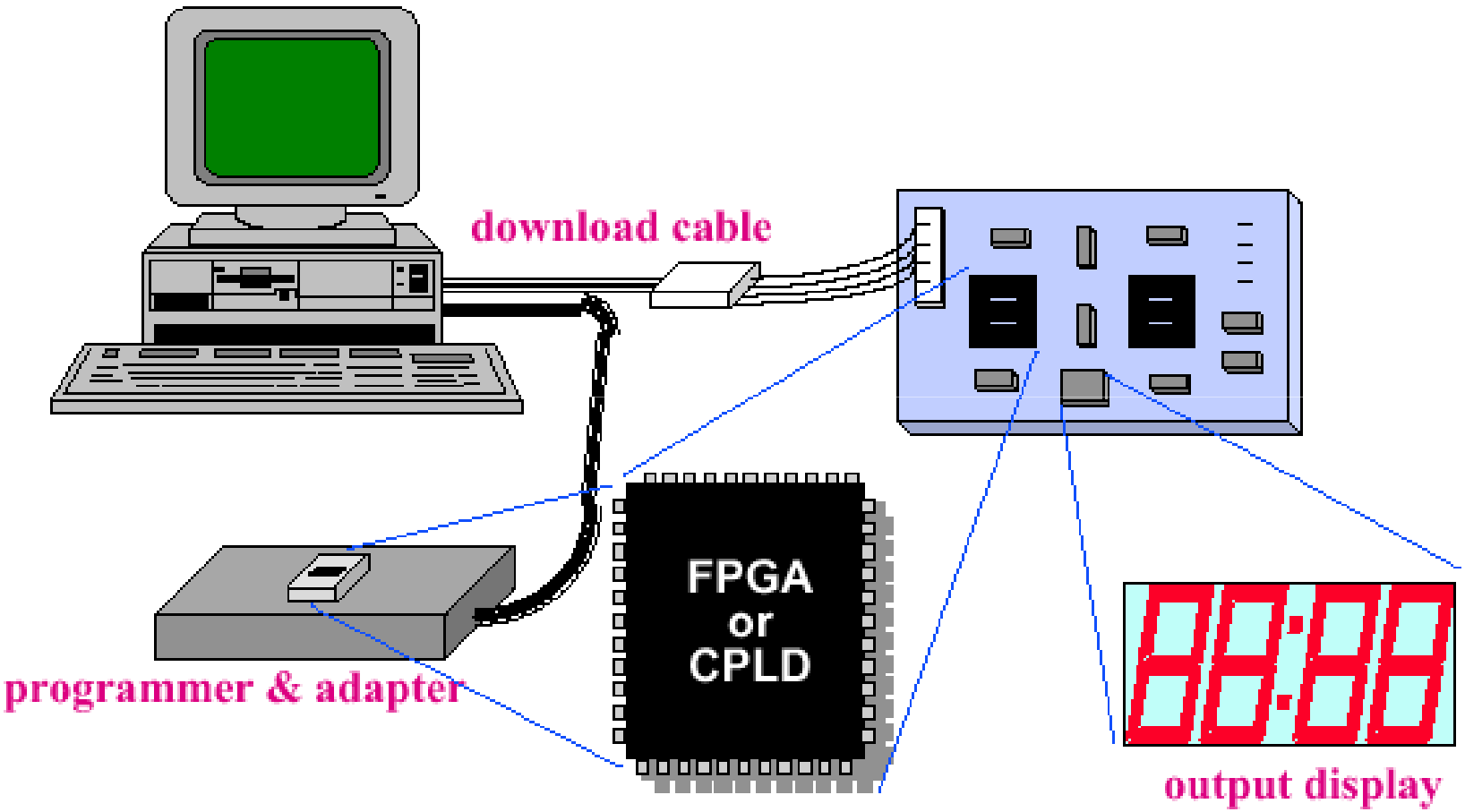
$$\text{si}x = 6$$

$$\lim_{x \to 8} \frac{1}{x - 8} = \infty$$

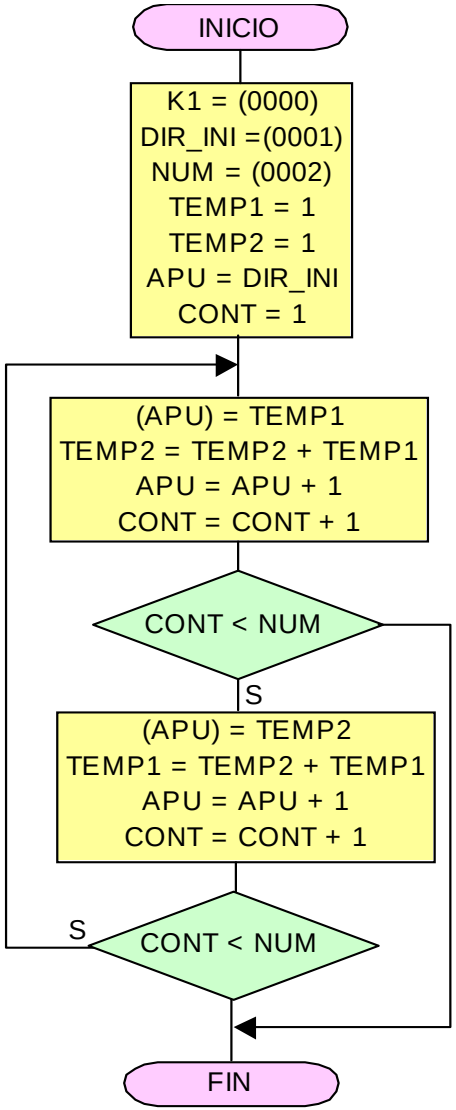


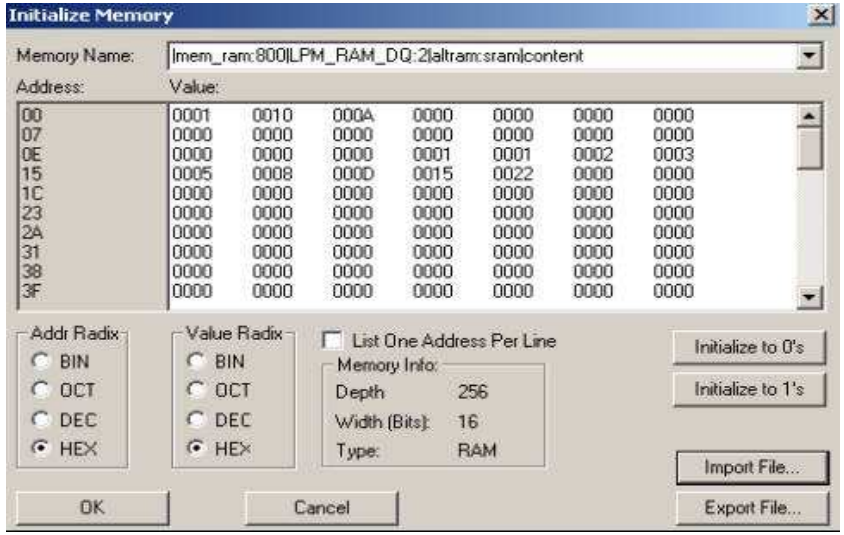
$$\lim_{x \to 5} \frac{1}{x - 5} = \text{ь}$$

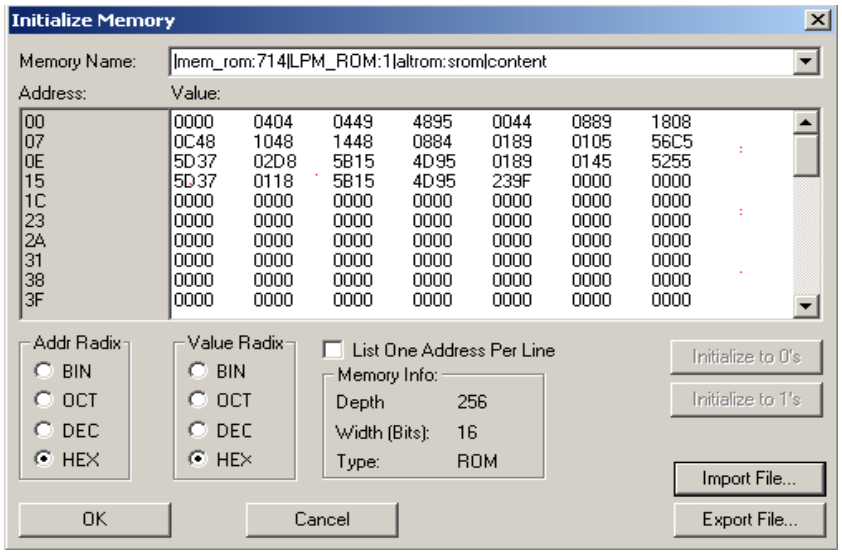
Sistemas Digitales II



Sistemas Digitales II



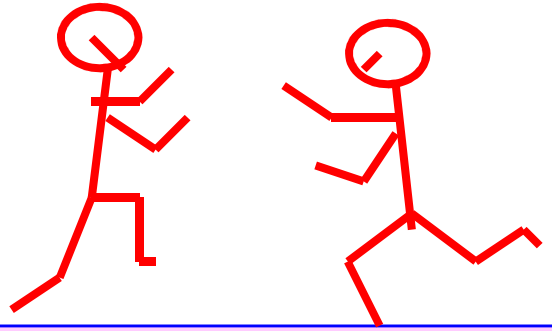




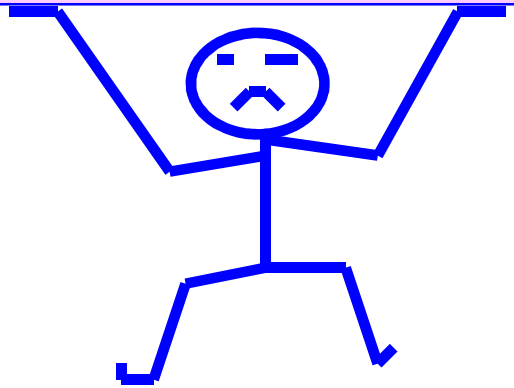
Sistemas Digitales

I

Circuitos combinatorios



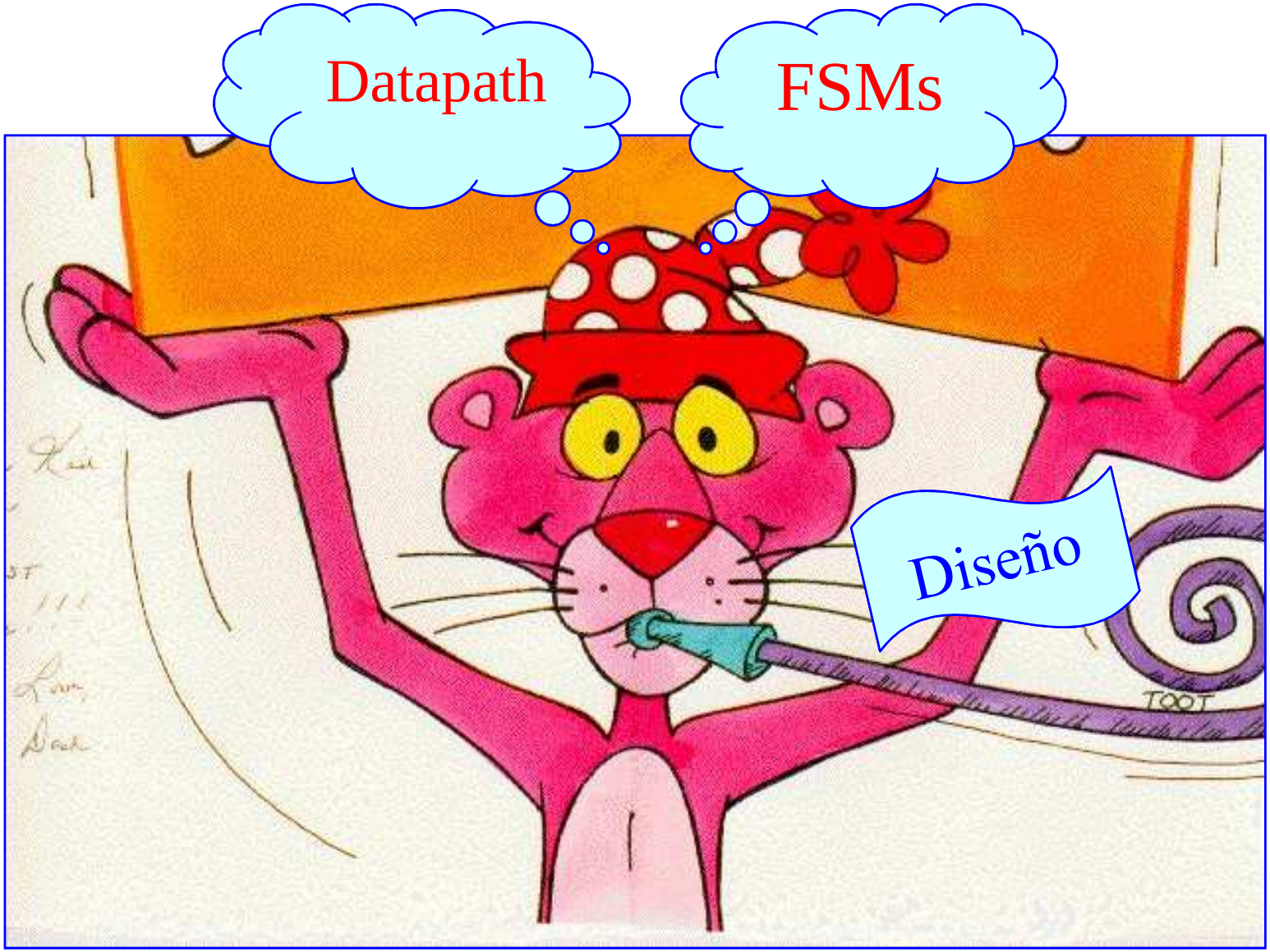
Sistemas Digitales



II

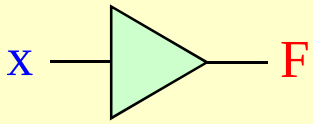
Circuitos secuenciales

Sistemas Digitales




Sincrono
Asincrono
Diseño

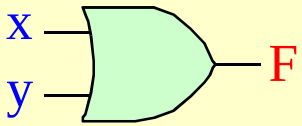
1.1 Basic logic gates



x	F
0	0
1	1

F = x

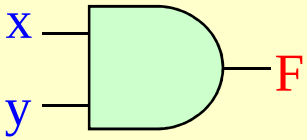
Driver



x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

F = x + y

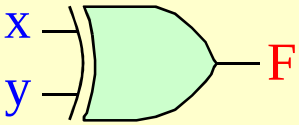
OR



x	y	F
0	0	0
0	1	0
1	0	0
1	1	1

F = x . y

AND

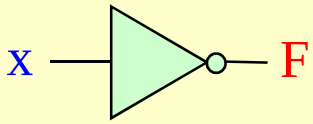


x	y	F
0	0	0
0	1	1
1	0	1
1	1	0

F = x ⊕ y

XOR

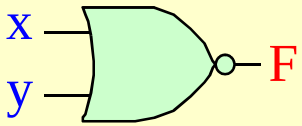
1.1 Basic logic gates



x	F
0	1
1	0

F = x’

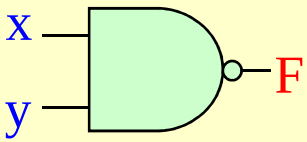
Inverter



x	y	F
0	0	1
0	1	0
1	0	0
1	1	0

F = (x + y)’

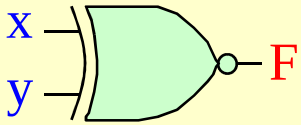
NOR



x	y	F
0	0	1
0	1	1
1	0	1
1	1	0

F = (x . y)’

NAND

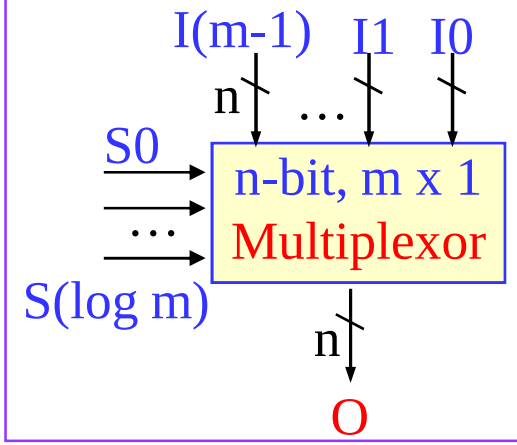


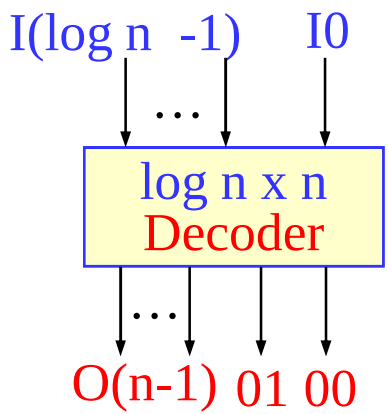
x	y	F
0	0	1
0	1	0
1	0	0
1	1	1

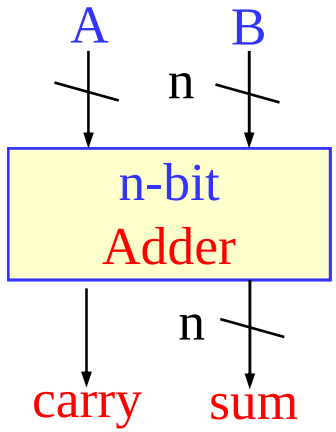
F = x ⊙ y

XNOR

1.2 Combinational components







O =
I0 if S=0..00
I1 if S=0..01
…
I(m-1) if S=1..11

O0 =1 if I=0..00
O1 =1 if I=0..01
…
O(n-1) =1 if I=1..11

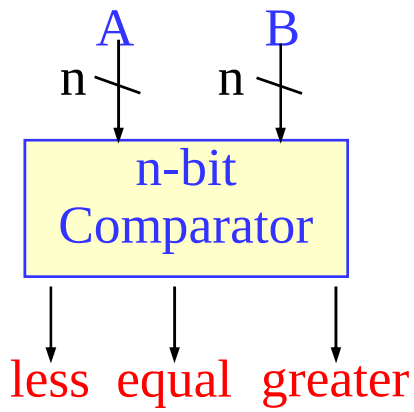
sum = A+B
(first n bits)
carry = (n+1)'th
bit of A+B

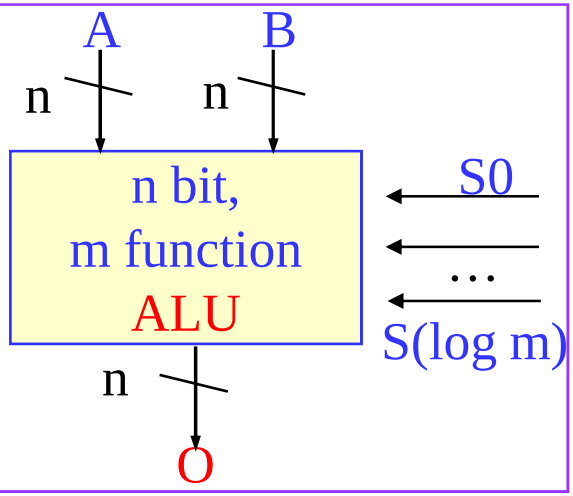
With enable input e →
all O's are 0 if e = 0

With carry-in input
Ci→

sum = A + B + Ci

1.2 Combinational components





less = 1 if A<B
equal =1 if A=B
greater=1 if A>B

O = A *op* B
op determined
by S.

May have status outputs
carry, zero, etc.

1.3 Combinational logic design

A) Problem description

- *y is 1 if a is to 1, or b and c are 1.*
- *z is 1 if b or c is to 1, but not both, or if all are 1.*

B) Truth table

Inputs			Outputs	
a	b	c	y	z
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

1.3 Combinational logic design

C) Output equations

y = a'bc + ab'c' + ab'c + abc' + abc

z = a'b'c + a'bc' + ab'c + abc' + abc

D) Minimized output equations

y

a \ bc	00	01	11	10
0	0	0	1	0
1	1	1	1	1

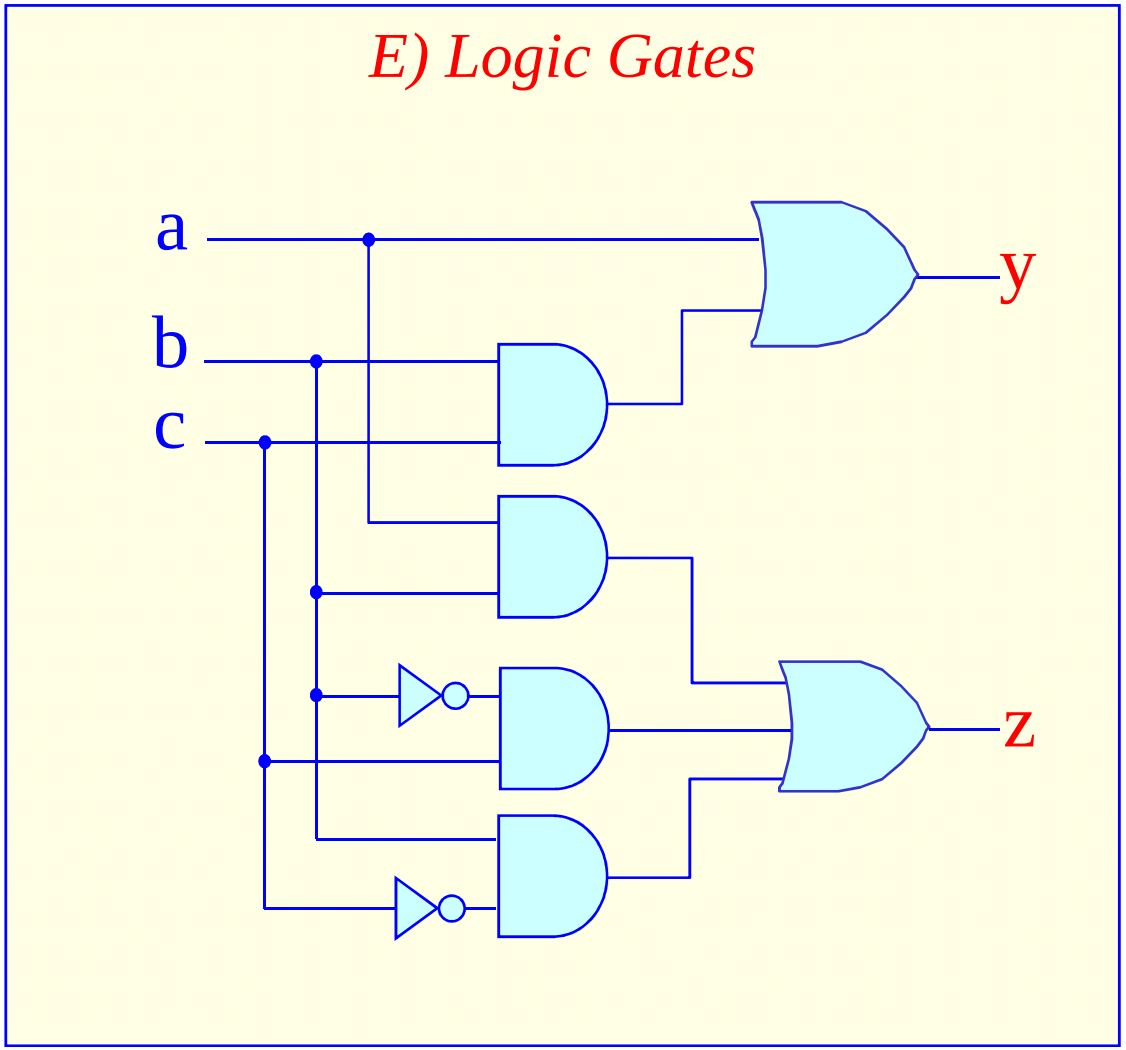
y = a + bc

z

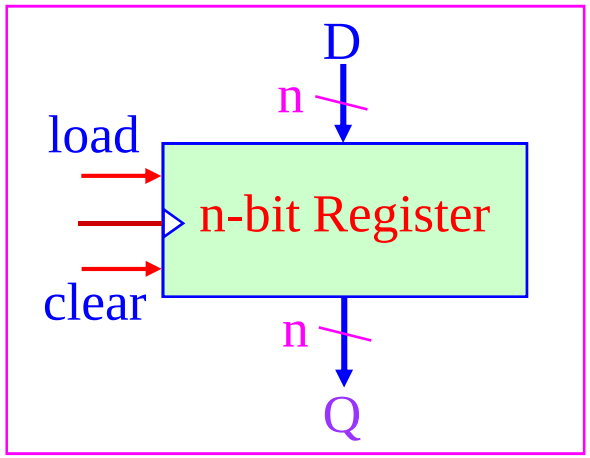
a \ bc	00	01	11	10
0	0	1	0	1
1	0	1	1	1

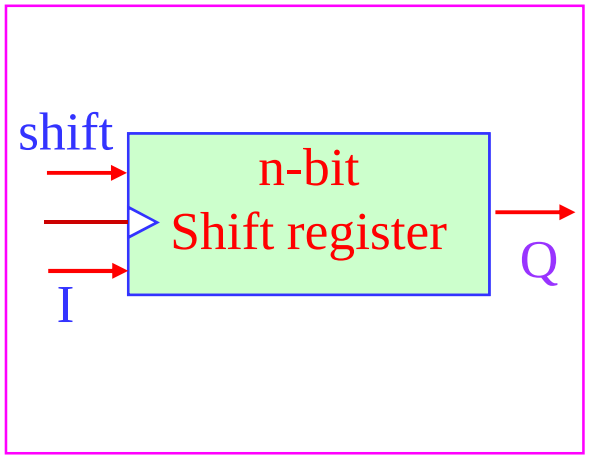
z = ab + b’c + bc’

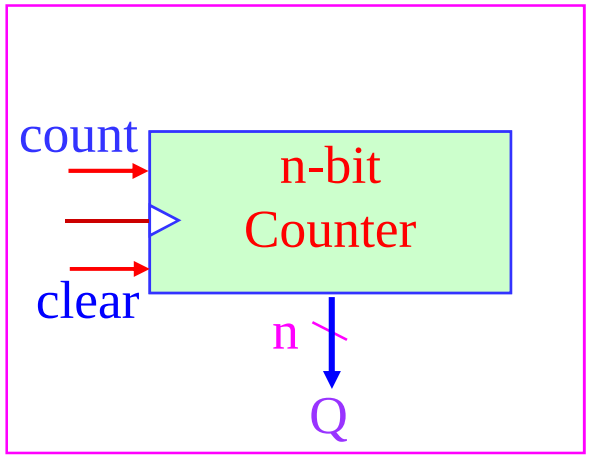
1.3 Combinational logic design



2.1 Sequential components







Q =
0 if clear = 1,
D if load = 1 and clock =1,
Q (previous) otherwise

Q = lsb
- Content shifted
- I stored in msb

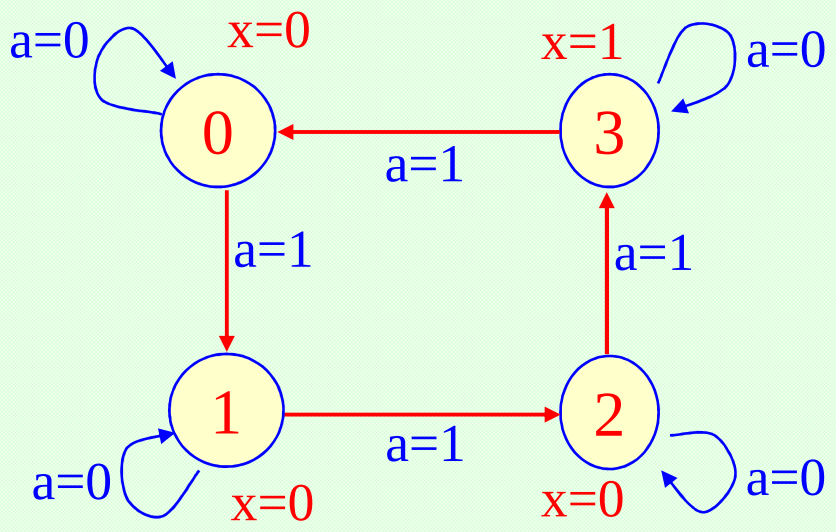
Q =
0 if clear = 1,
Q (prev) + 1 if count =1
and clock = 1.

2.2 Sequential logic design

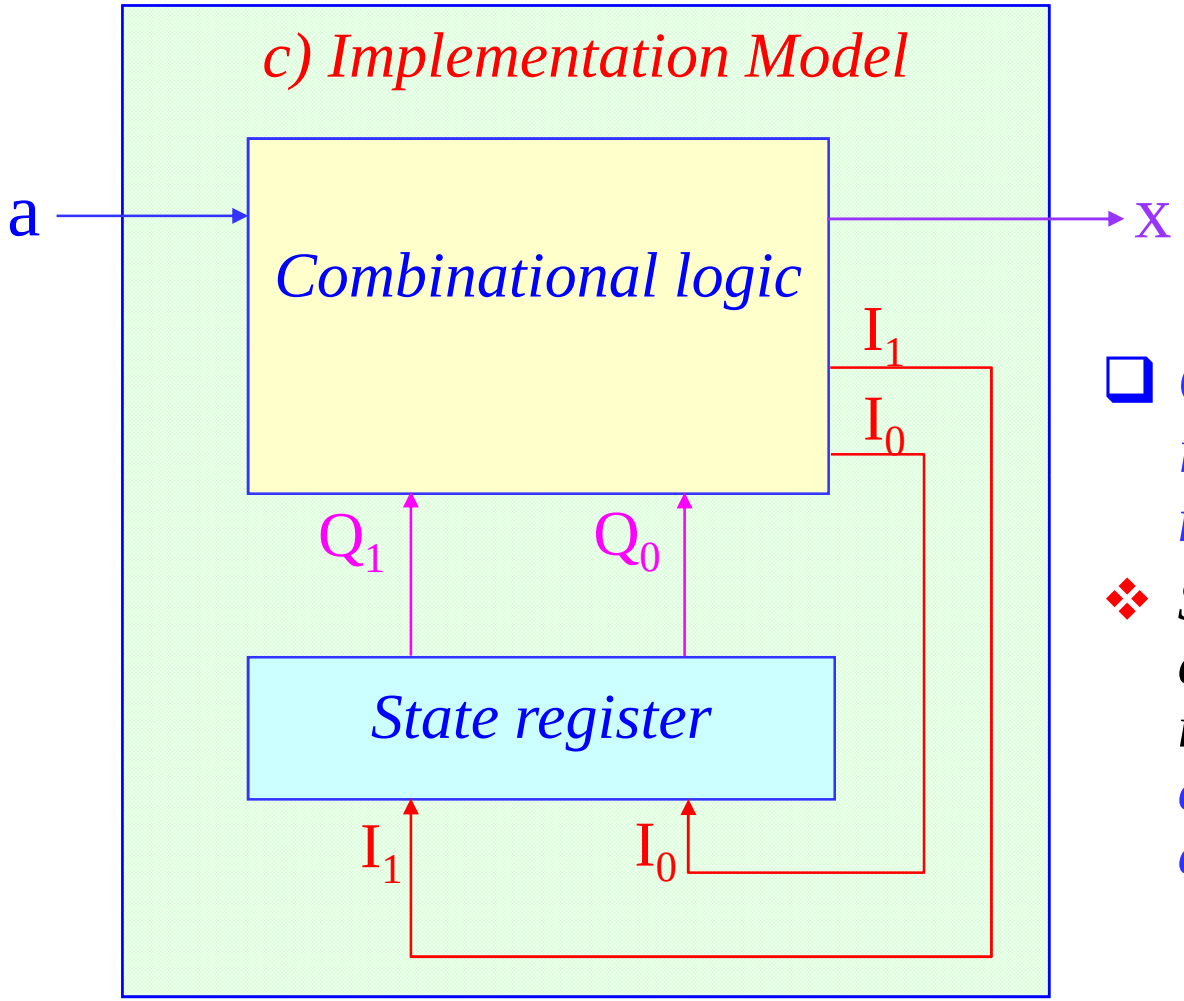
A) Problem Description

- *You want to construct a clock divide*
- *Slow down your pre-existing clock so that you output a1 for every four clock cycles*

B) State Diagram



2.2 Sequential logic design



- *Given this implementation model*
- *Sequential logic design quickly reduces to combinational logic design*

2.2 Sequential logic design

D) State Table (Moore-type)

Inputs			Outputs		
Q1	Q0	a	I1	I0	x
0	0	0	0	0	0
0	0	1	0	1	
0	1	0	0	1	0
0	1	1	1	0	
1	0	0	1	0	0
1	0	1	1	1	
1	1	0	1	1	1
1	1	1	0	0	

2.2 Sequential logic design

E) Minimized Output Equations

I1

a \ Q1Q0	00	01	11	10
0	0	0	1	1
1	0	1	0	1

I1 = Q1'Q0a + Q1a' + Q1Q0'

I0

a \ Q1Q0	00	01	11	10
0	0	1	1	0
1	1	0	0	1

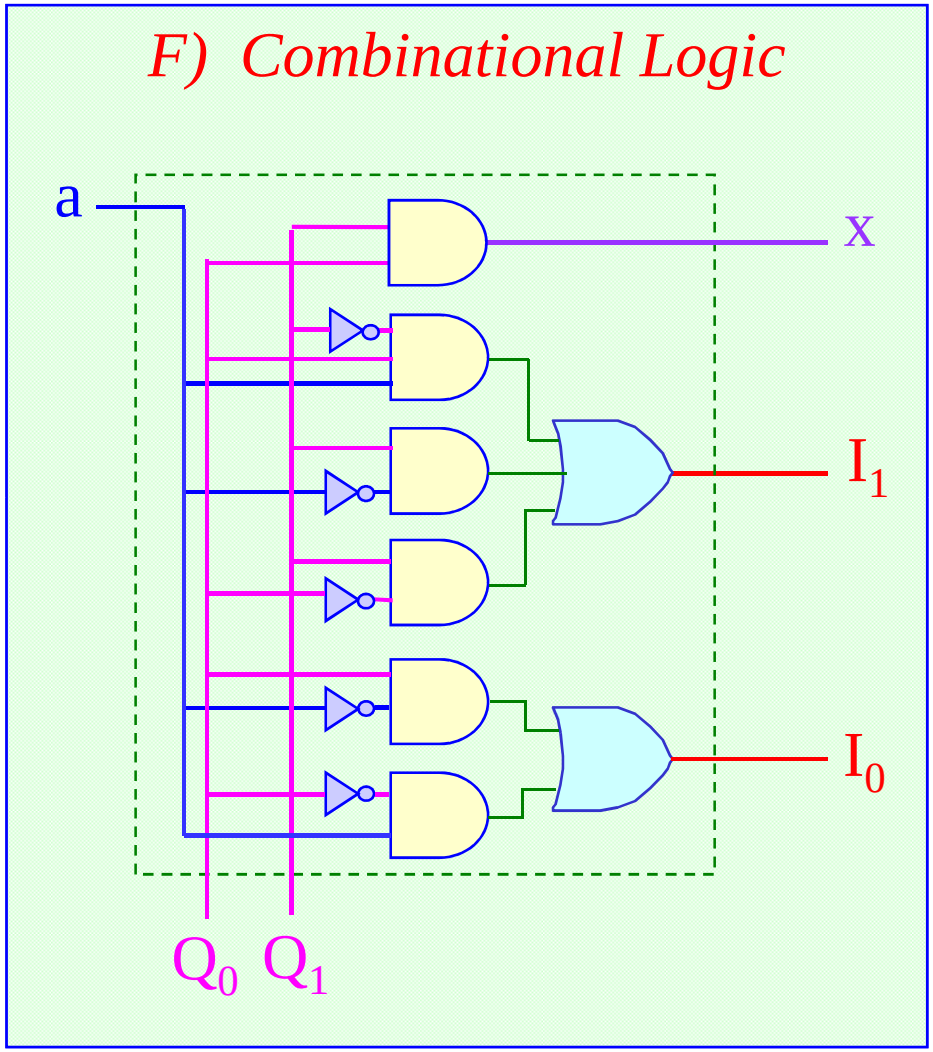
I0 = Q0a' + Q0'a

x

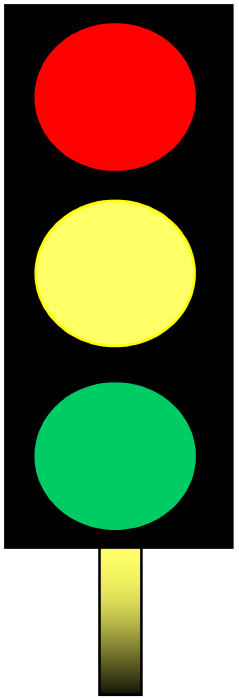
a \ Q1Q0	00	01	11	10
0	0	0	1	0
1	0	0	1	0

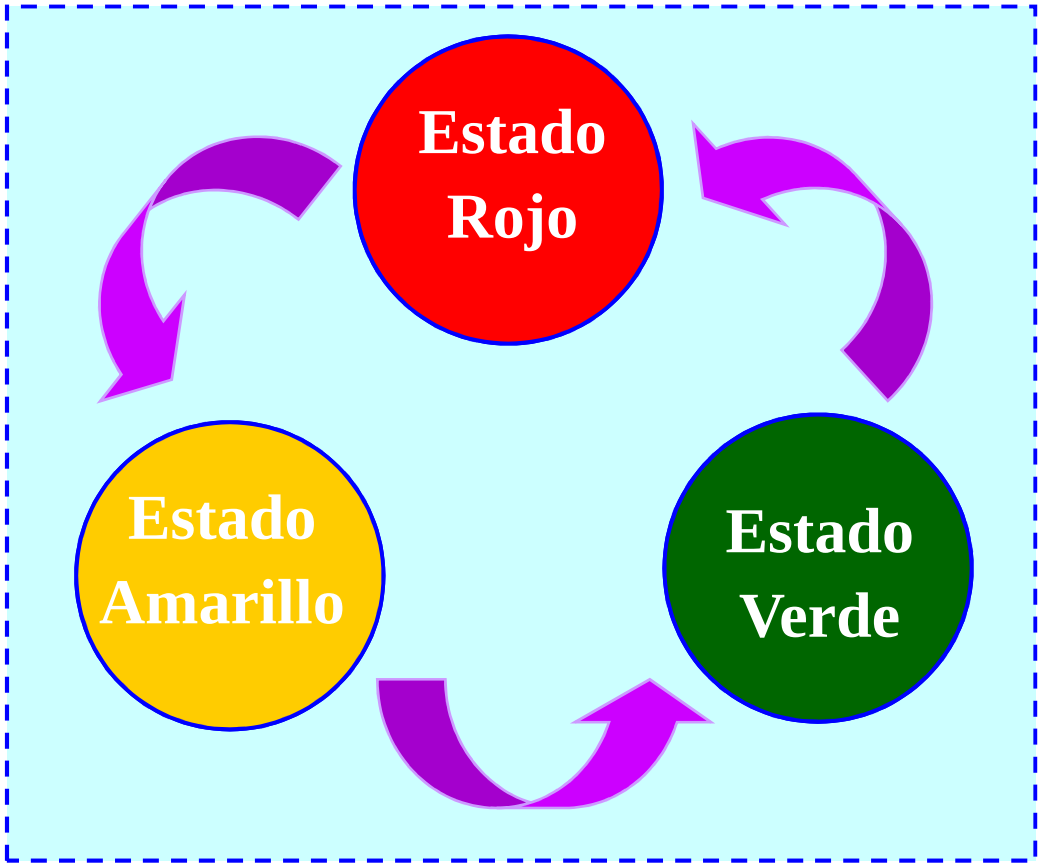
x = Q1Q0

2.2 Sequential logic design



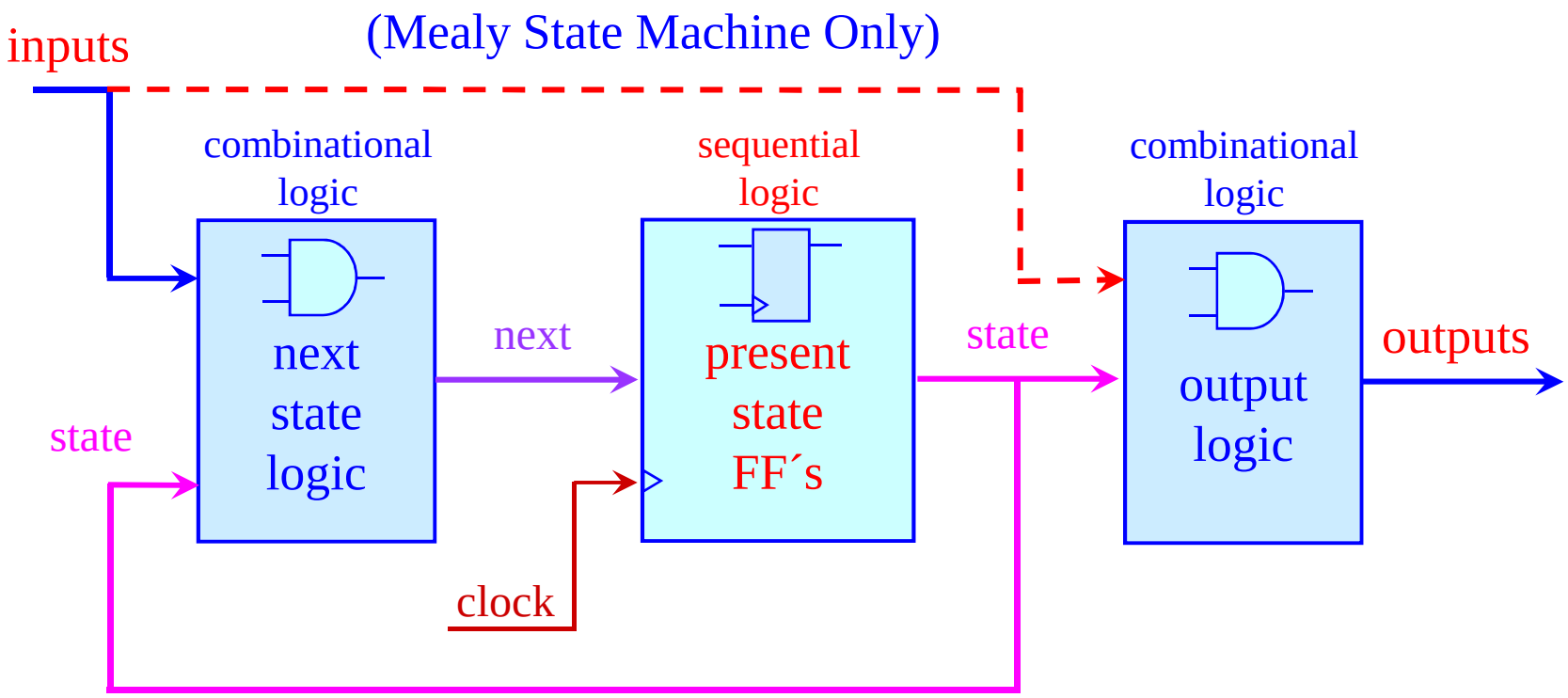
2.2 Sequential logic design: FSM





2.2 Sequential logic design: FSM

- *Basic FSM Structure*



FSM Block Diagram

2.2 Sequential logic design: VHDL

LENGUAJES DE DESCRIPCIÓN DE HARDWARE

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_SIGNED.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_1164.ALL;

ENTITY MUX IS
PORT(
        A,B,S: IN STD_LOGIC;
        Z:      OUT STD_LOGIC);
END;
ARCHITECTURE  FIRST OF MUX IS
BEGIN
        CASE S IS
                WHEN 0 => Z<=A;
                WHEN 1 => Z<=B;
        END CASE;
END FIRST;
```

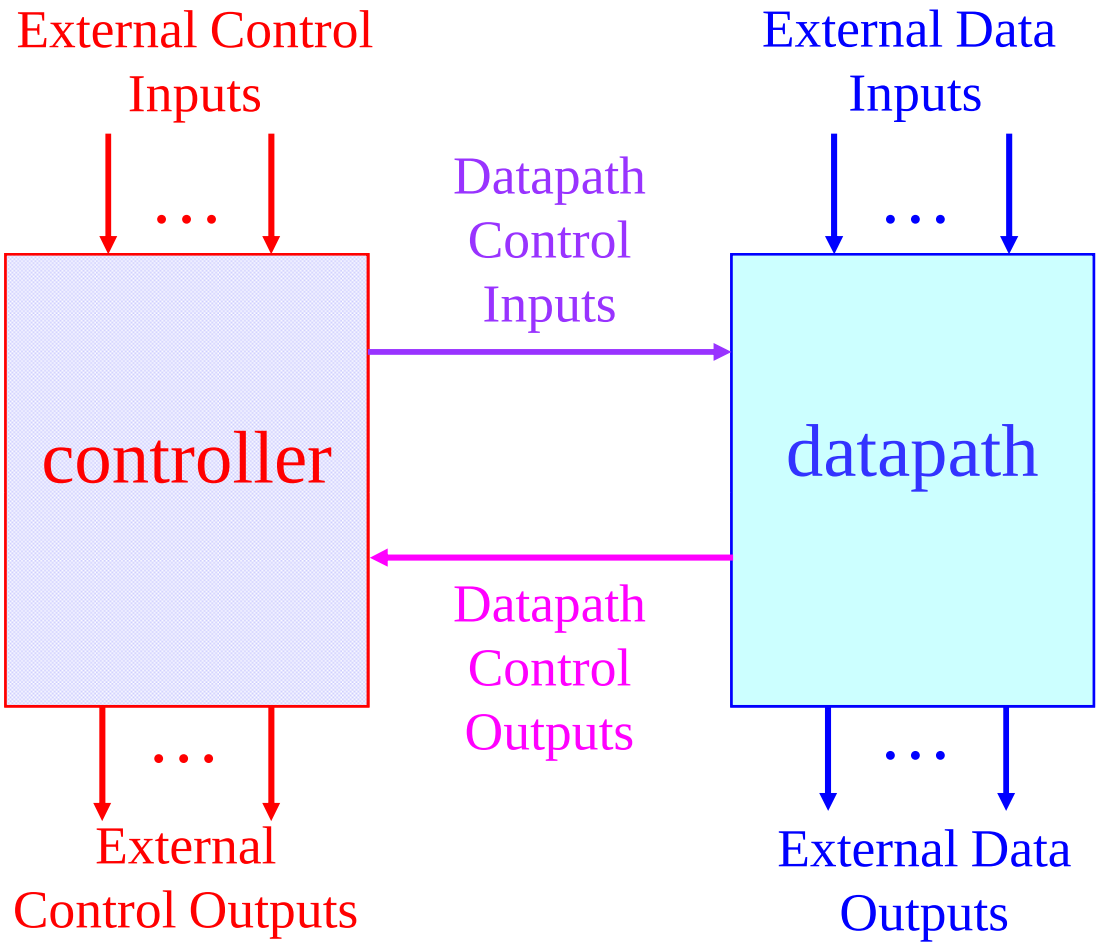
Verilog HDL:

Verilog Hardware Description Language

VHDL:

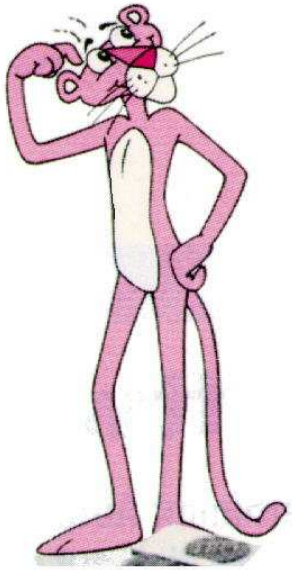
Very High Speed Integrated Circuit HDL

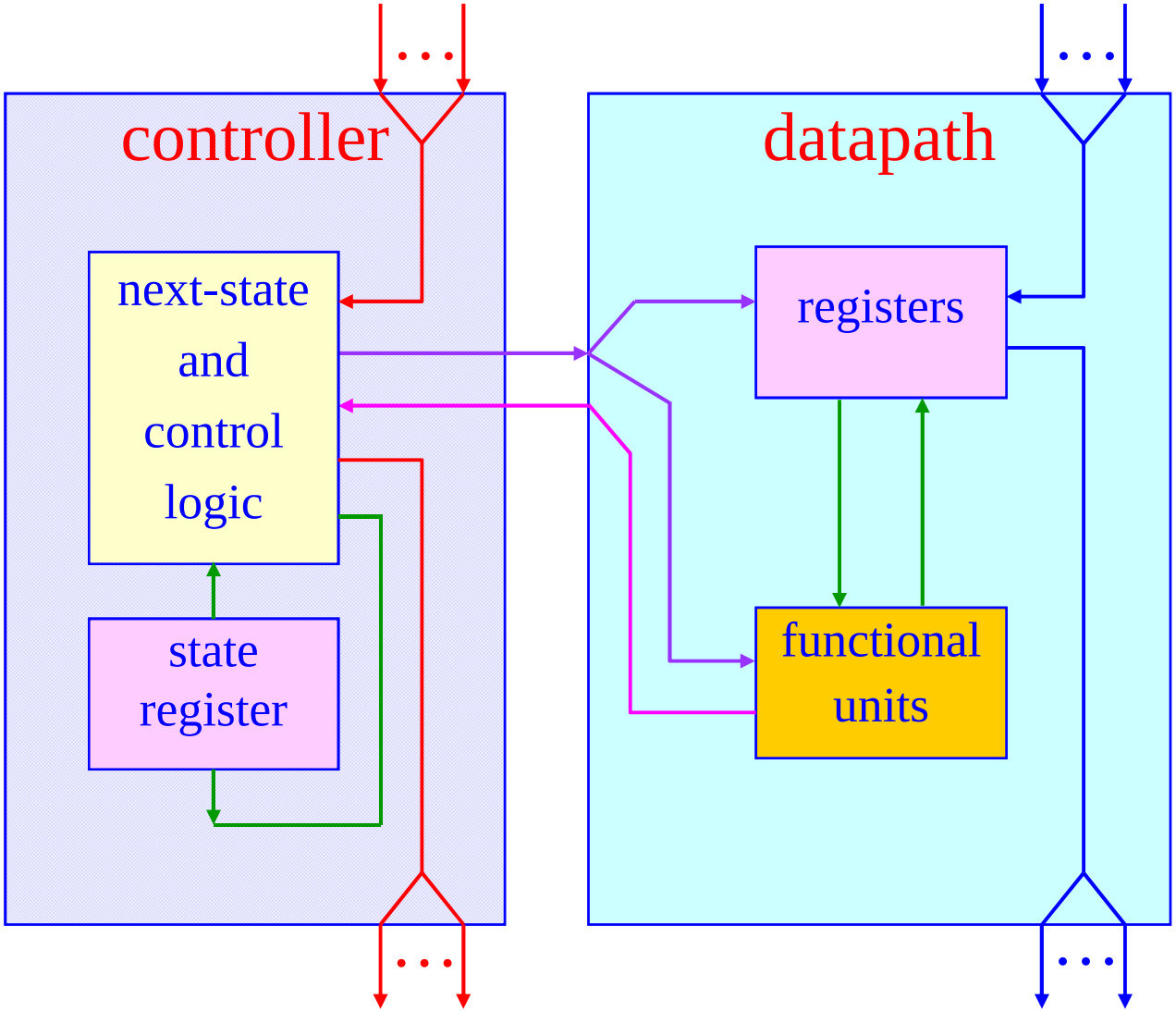
3. Digital System Basic Model



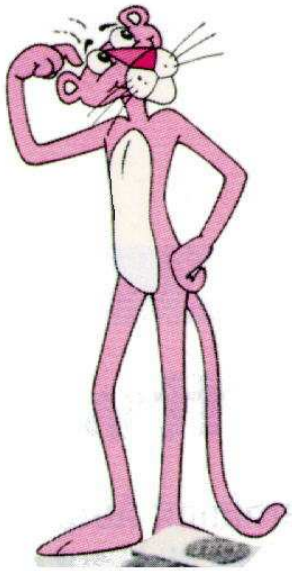
Controller and Datapath

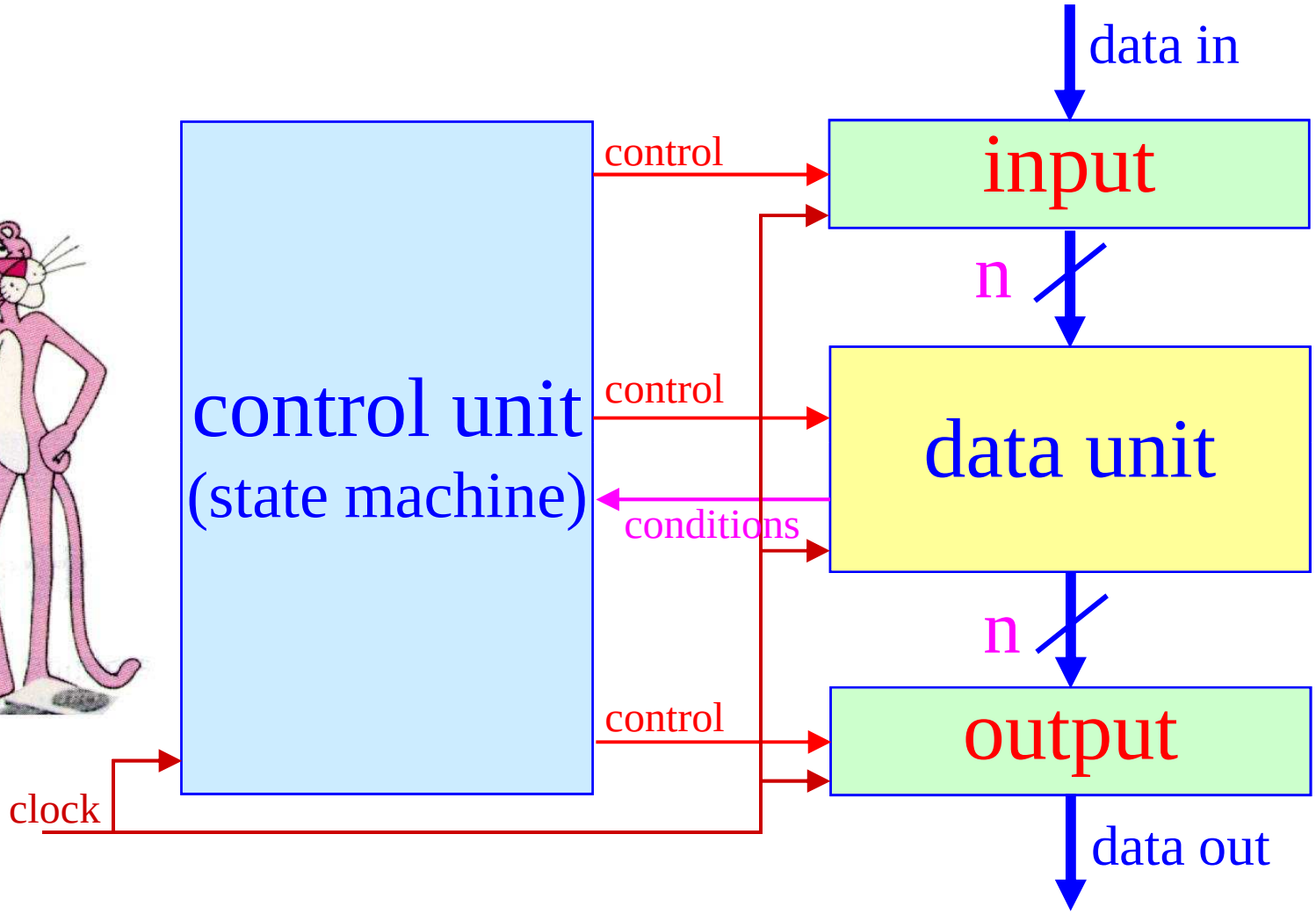
3. View inside: controller and datapath





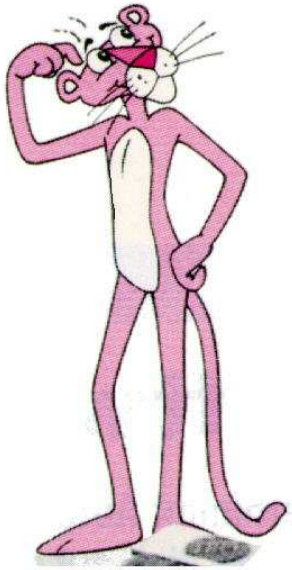
3. Synchronous System Structure

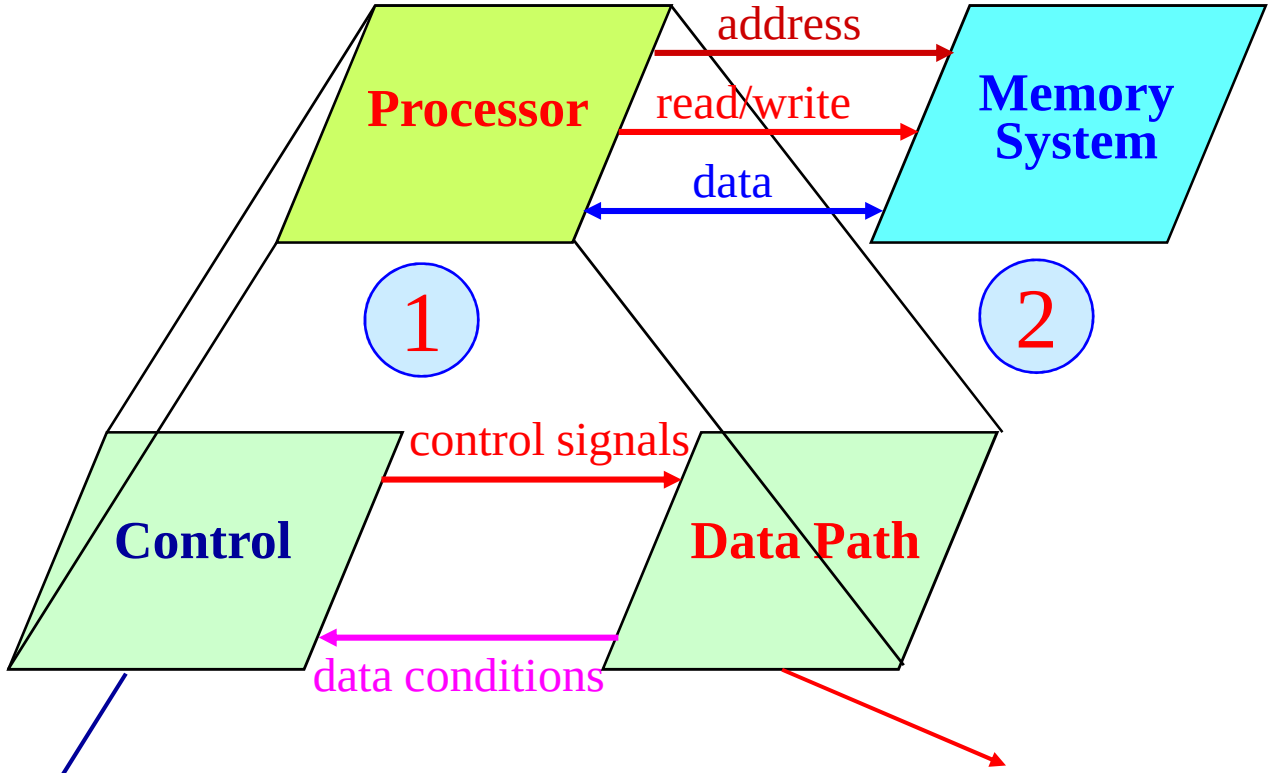




4. Anatomy of a Computer

- *Block diagram view: processor*



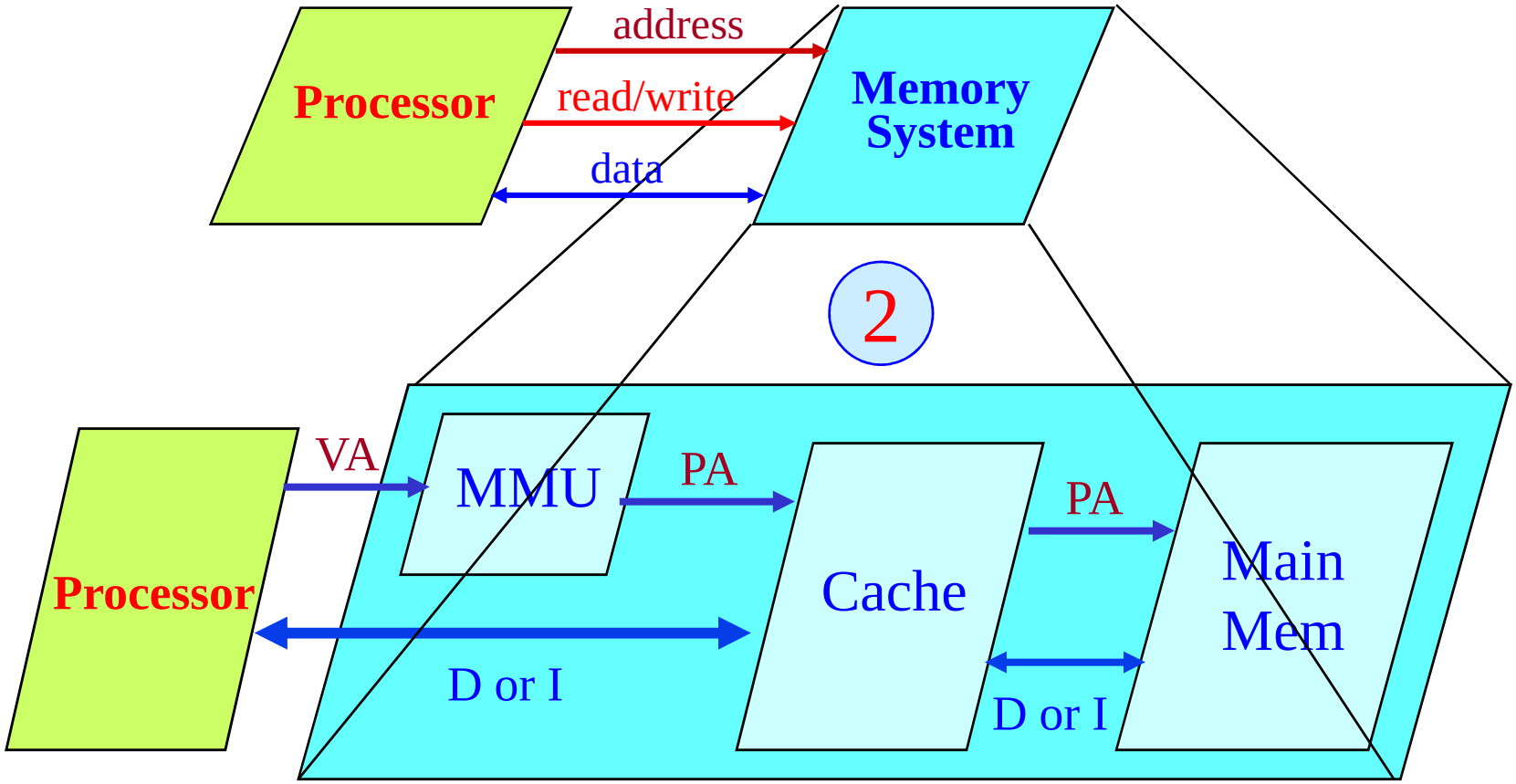


Instruction Unit:
instruction fetch and interpretation FSM

Execution Unit:
functional units and registers

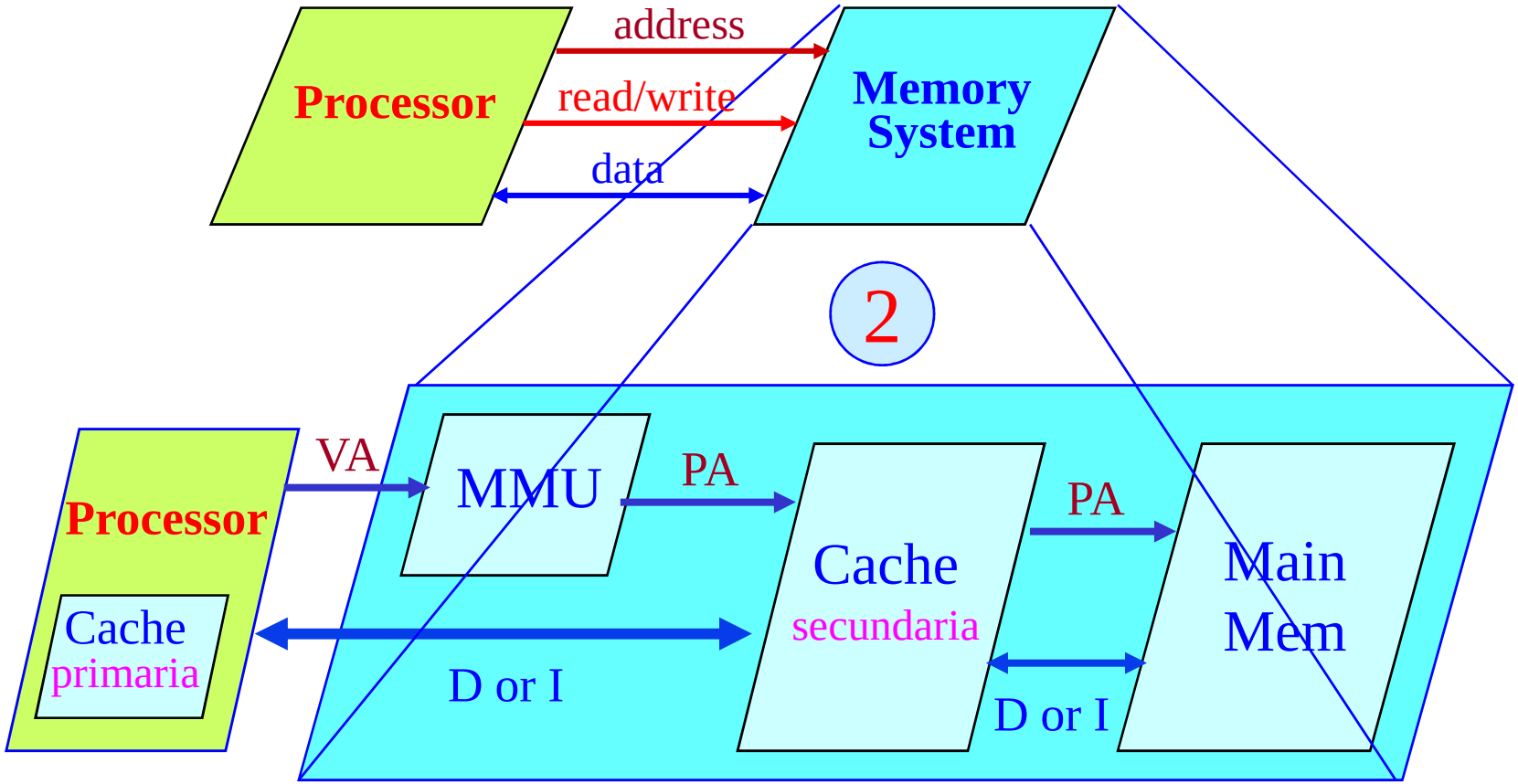
4. Anatomy of a Computer

- *Block diagram view: memory system*



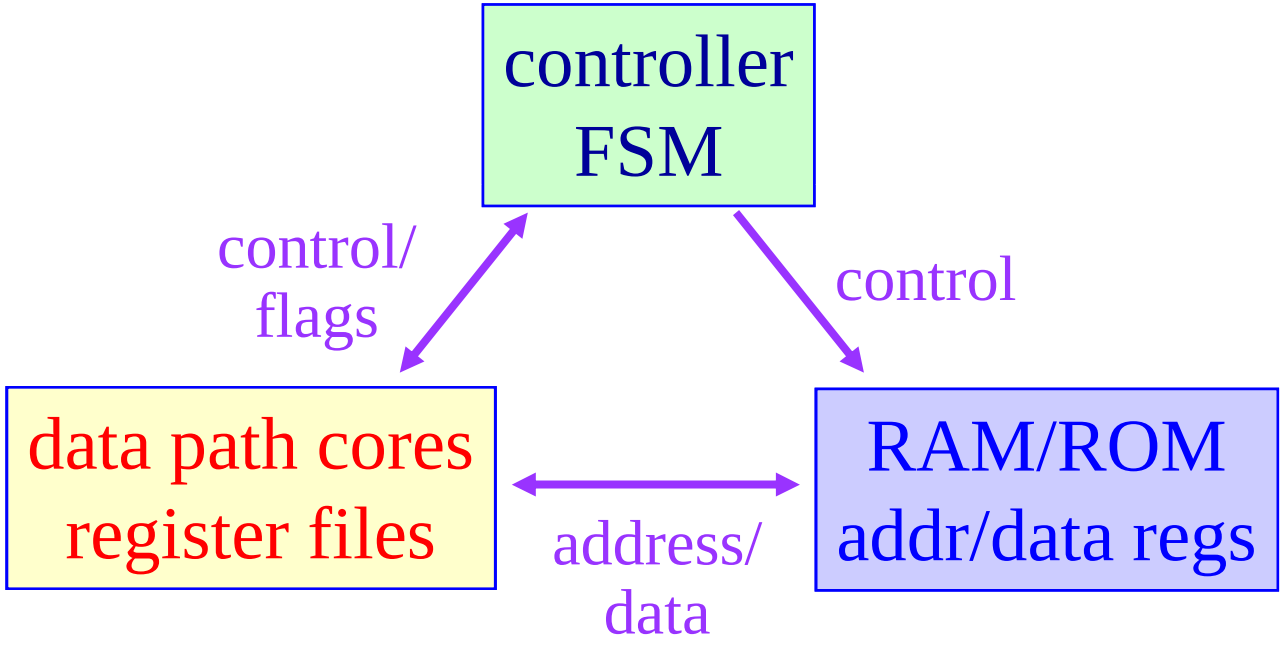
4. Anatomy of a Computer

- *Block diagram view: memory system*



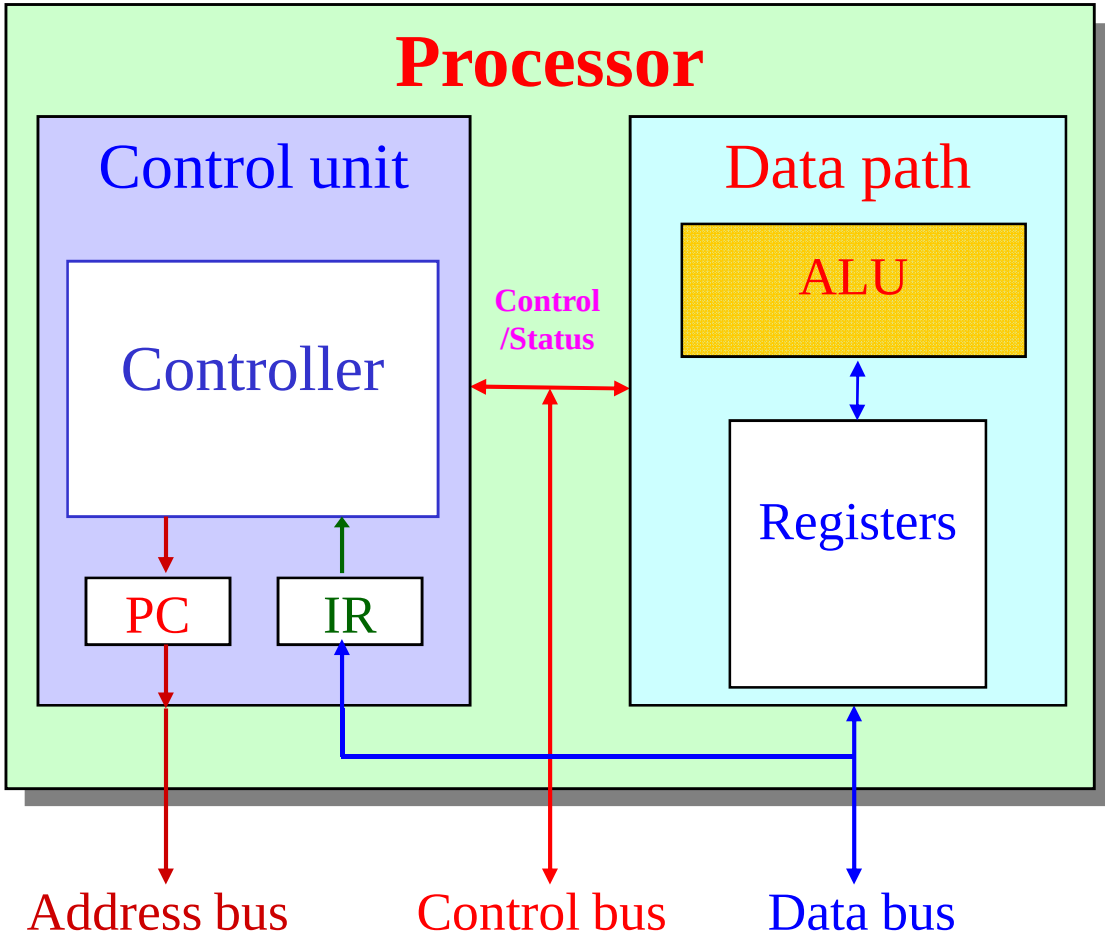
4.1 Anatomy of a Digital System

- *Block diagram view*
  - *Data path*: fuctional building blocks are involved
  - *Control path*: circuits of decision are involved

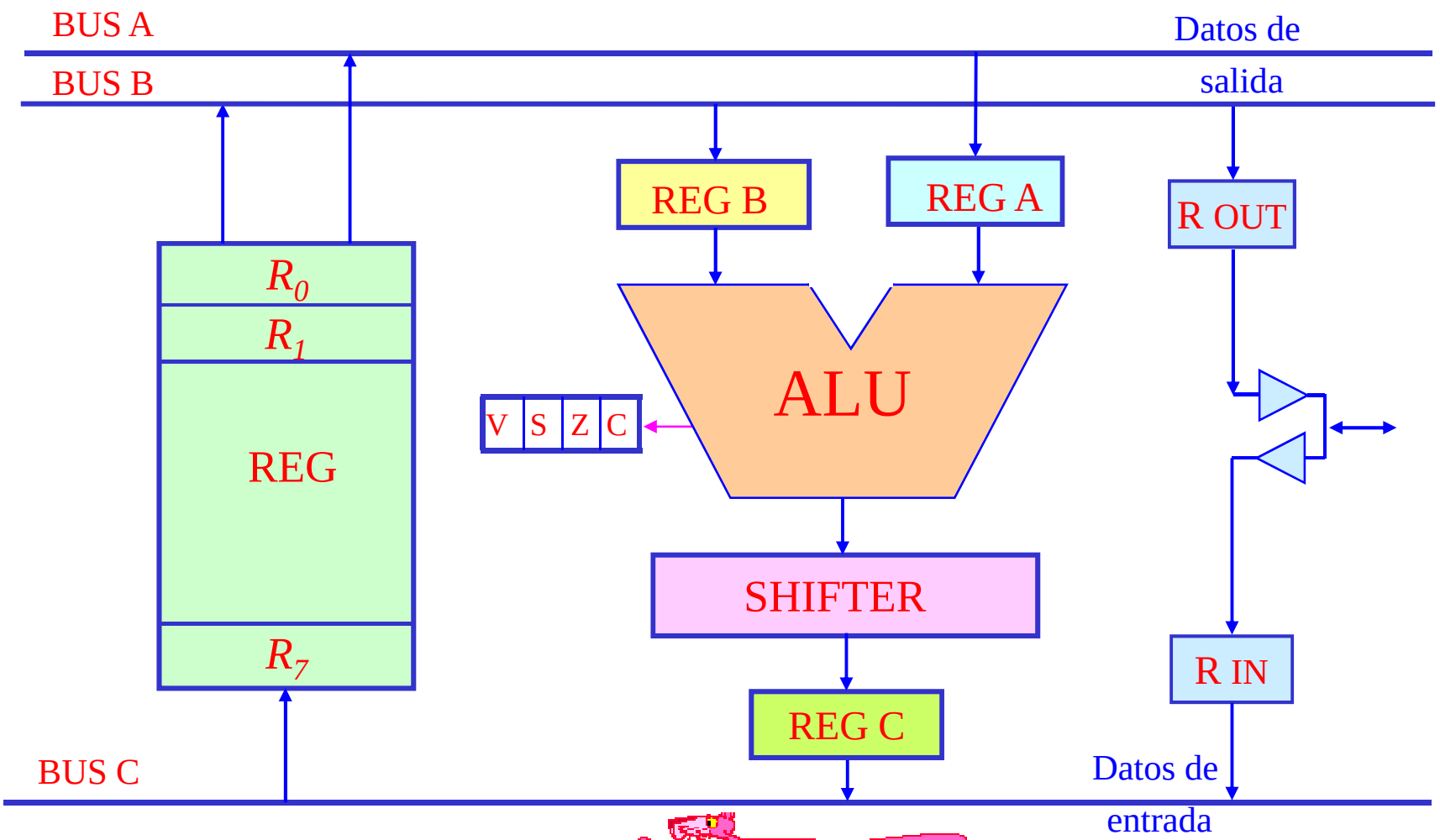


4.2 Anatomy of a Basic Processor

- *Block diagram view*
  - *Data path*: ALU and registers
  - *Control path*: control unit

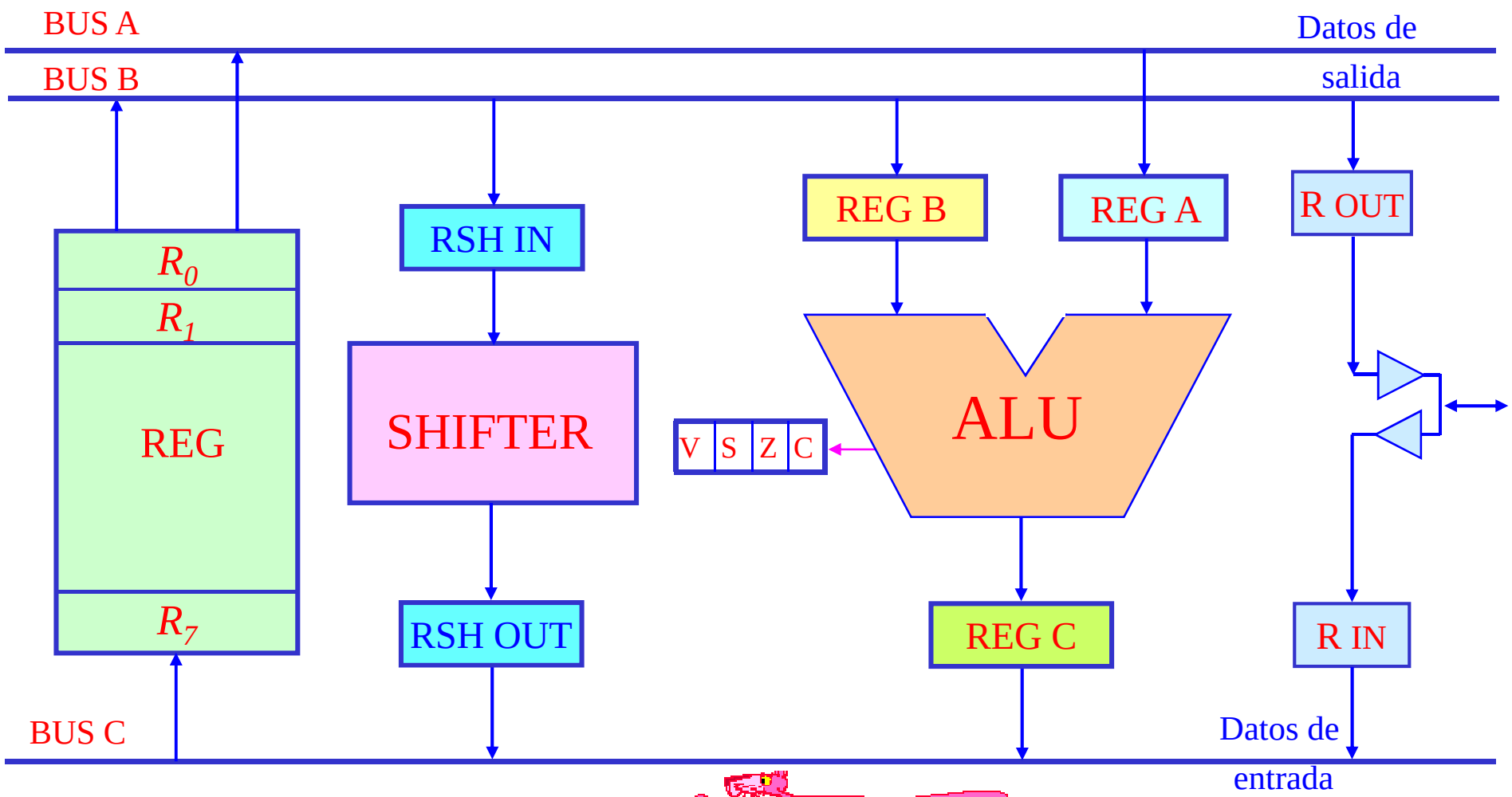


Data-Path Unit Design: UV1-2005



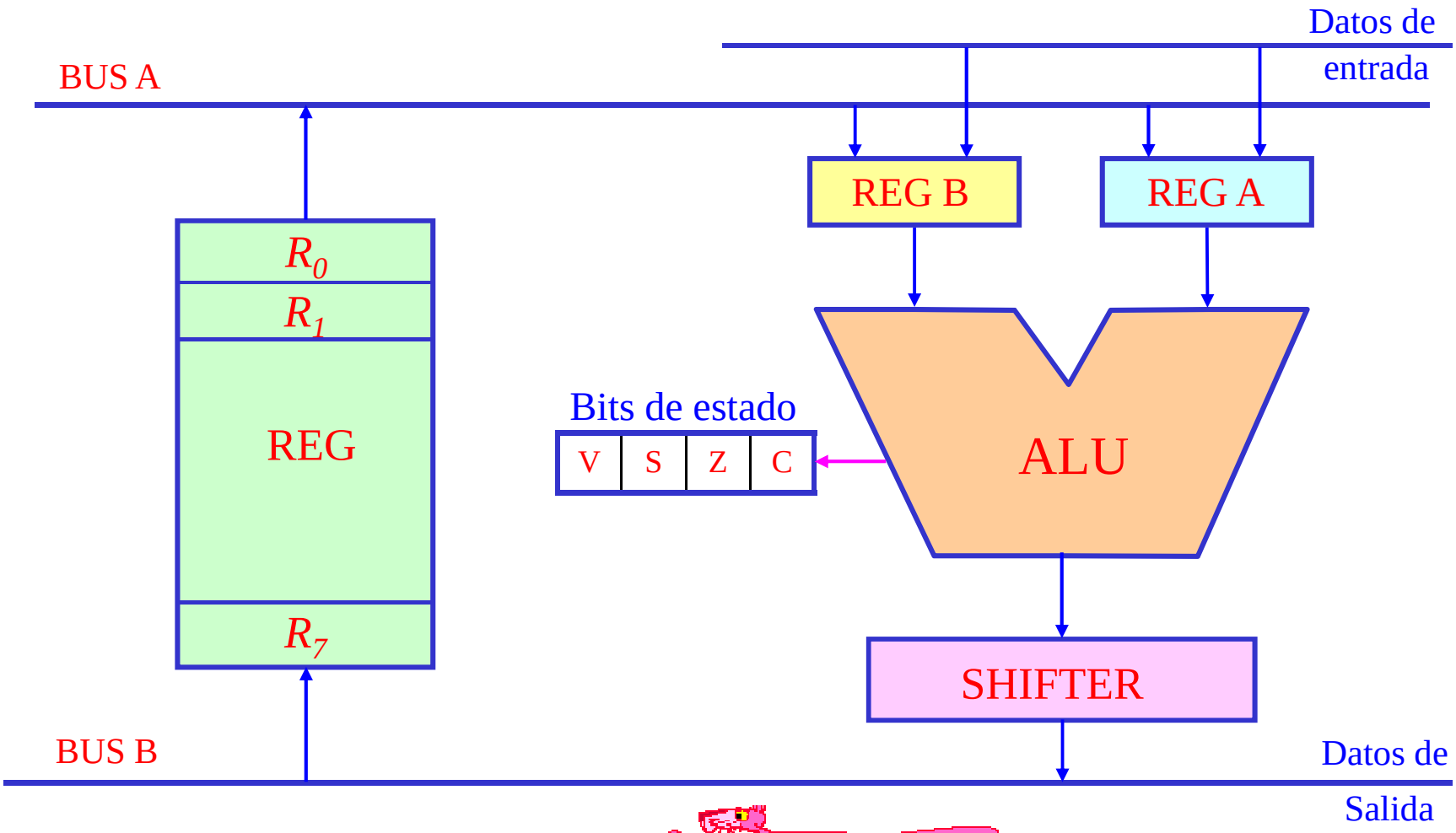
Load/store architecture

Data-Path Unit Design: UV2-2005



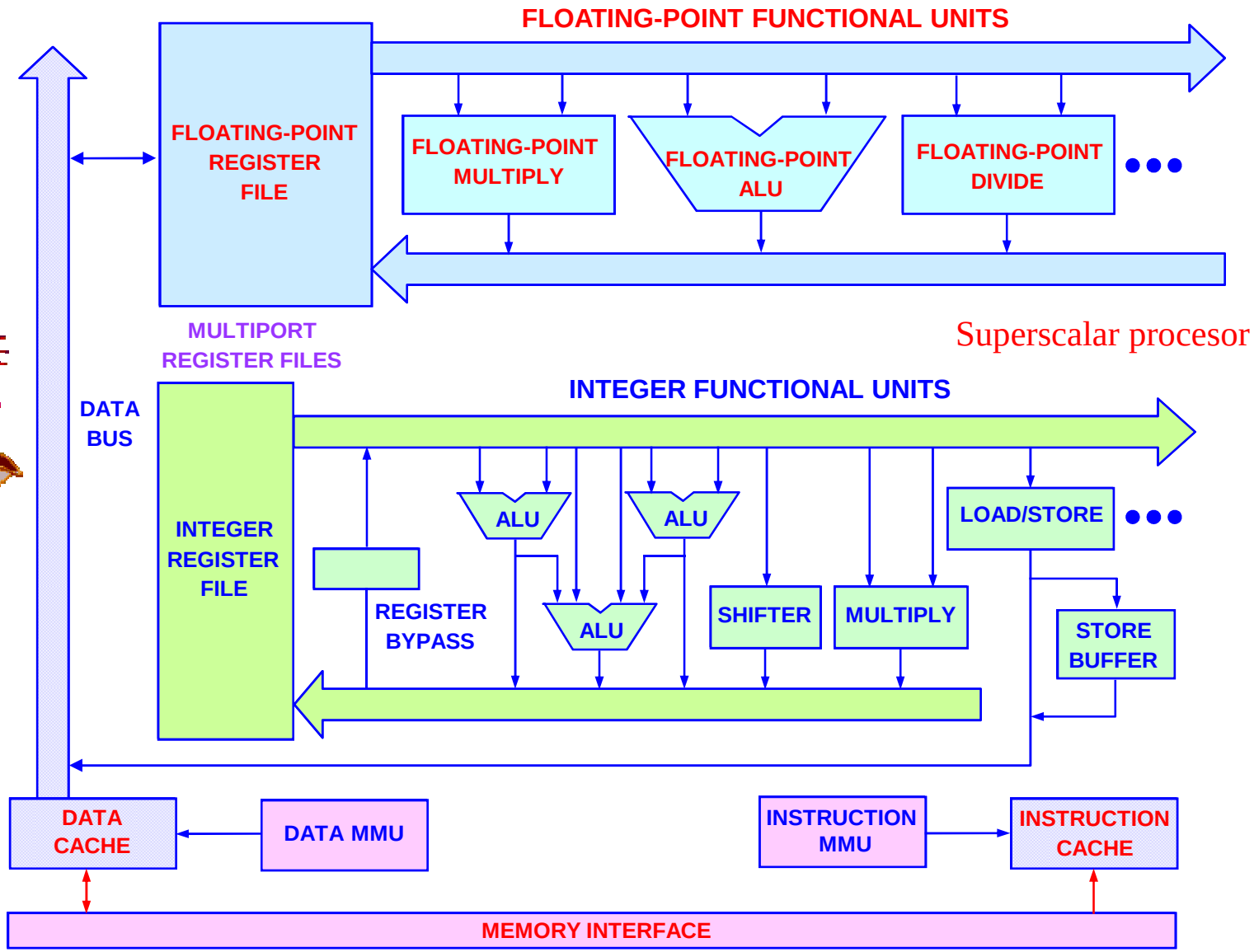
Load/store architecture

Data-Path Unit Design: UV2-2005



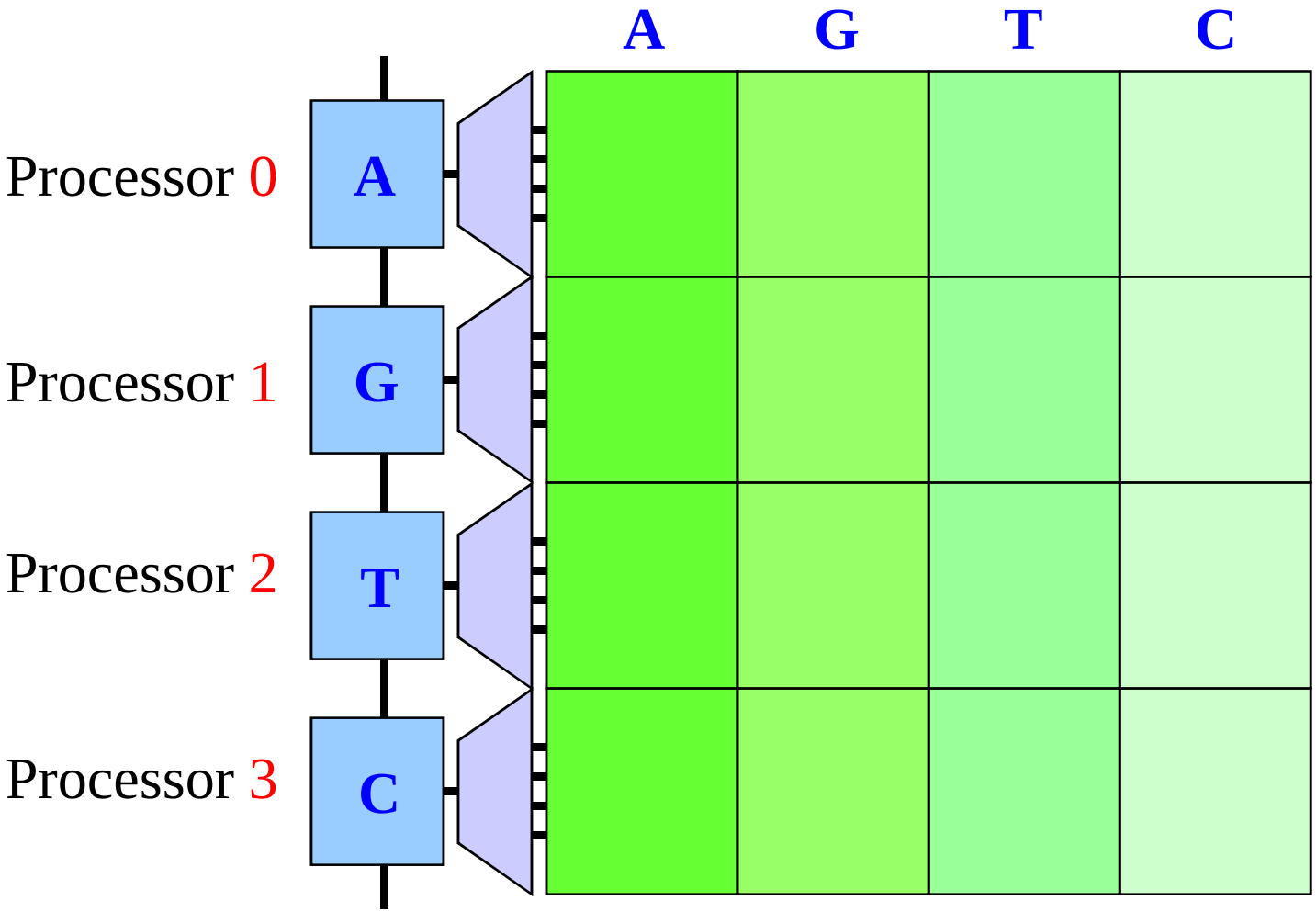
Load/store architecture !!

Superscalar procesor



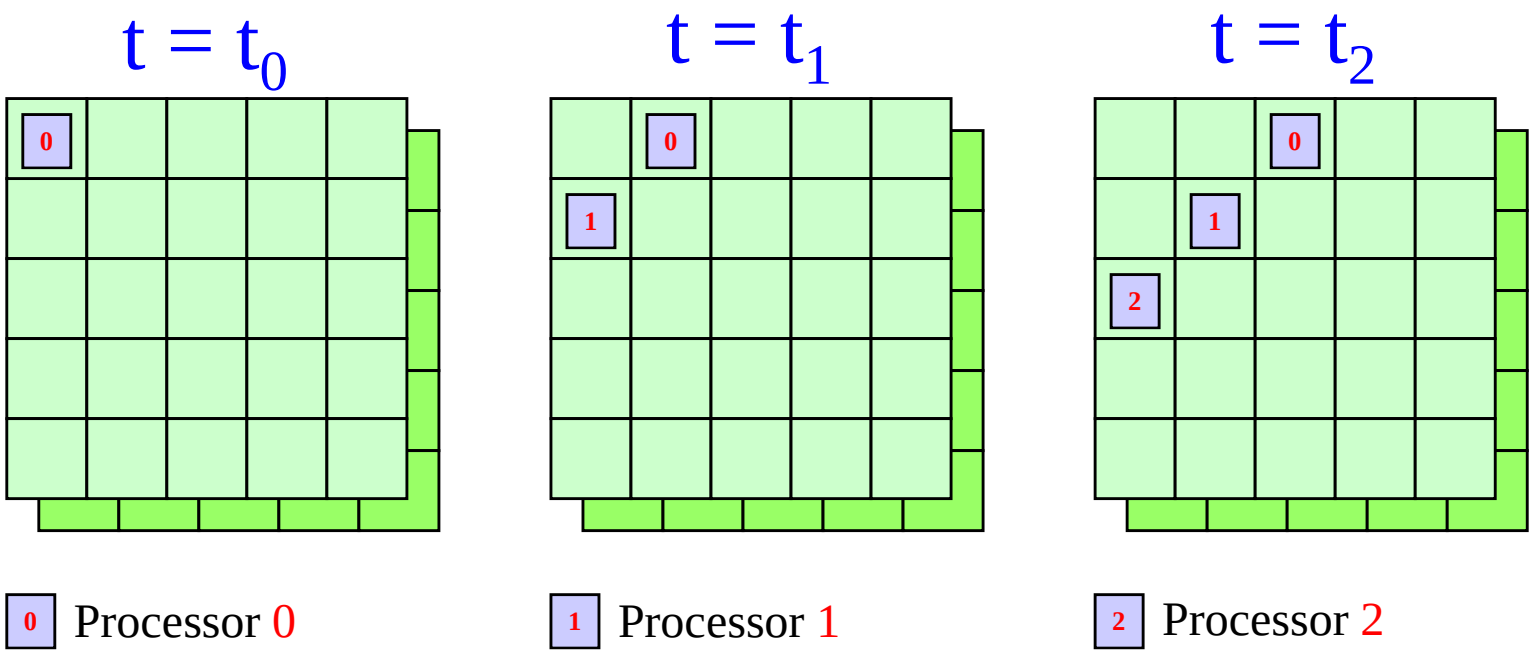
Systolic Implementation: DNA Processor

- *DNA processor* → uni-dimensional architecture

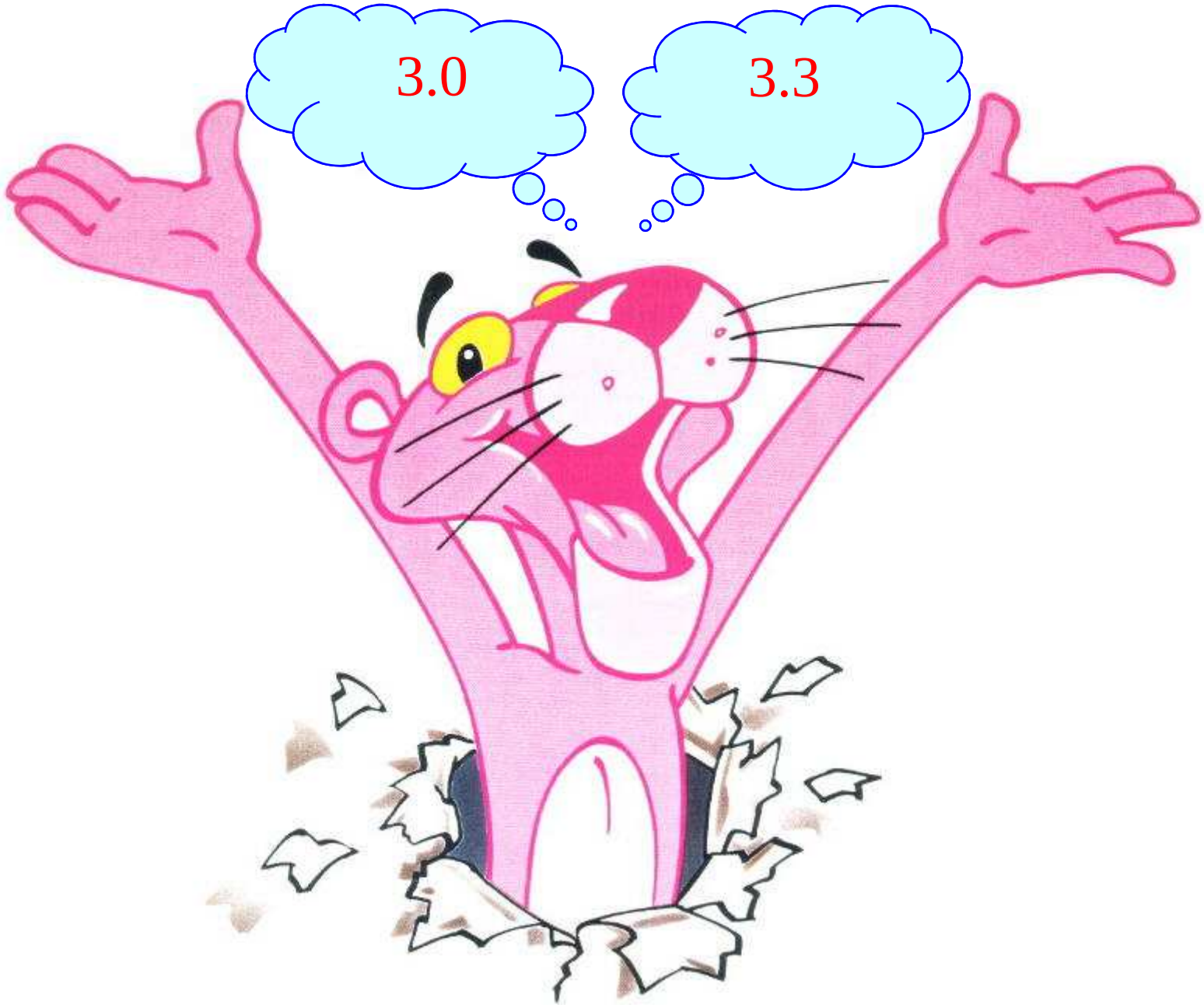


Systolic Implementation: DNA Processor

- *DNA processor* → uni-dimensional architecture




Univalle
the best


3.0
3.3

2.89

FSM-AFSM

MIKE

DATA PATH