

# VHDL

## Circuitos Síncronos

Rafael Corsi Ferrão - IMT



`rafael.corsi@maua.br`  
`http://www.maua.br`

13 de outubro de 2014

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

## 1 Conceitos

## 2 Circuitos Síncronos

- Flip-Flops e Latches

## 3 Declaração Latch

## 4 Descrição síncrona

## 5 exemplos

## Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

### 1 Conceitos

### 2 Circuitos Síncronos

- Flip-Flops e Latches

### 3 Declaração Latch

### 4 Descrição síncrona

### 5 exemplos

Conceitos

**Circuitos  
Síncronos**

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

1 Conceitos

2 Circuitos Síncronos  
■ Flip-Flops e Latches

3 Declaração Latch

4 Descrição síncrona

5 exemplos

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

## Definição

Circuitos síncronos são aqueles que utilizam um sinal de sincronismos para trigar sua execução, esse sinais são conhecidos como *clock*

Em quase toda sua totalidade, projetos digitais são do tipo síncronos. Alguns exemplos:

- ▶ Relógio
- ▶ filtros digitais
- ▶ televisão
- ▶ computador
- ▶ microprocessador
- ▶ ...

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

- ▶ Circuitos síncronos possuem necessariamente em sua formação elementos de memória (flip-flop)
- ▶ o *clock* é o sinal responsável pela sincronização do circuito
- ▶ as mudanças dos estados (variáveis e saídas) ocorrem na transição do *clock* - borda de subida **ou** borda de descida.

## Conceitos

## Circuitos Síncronos

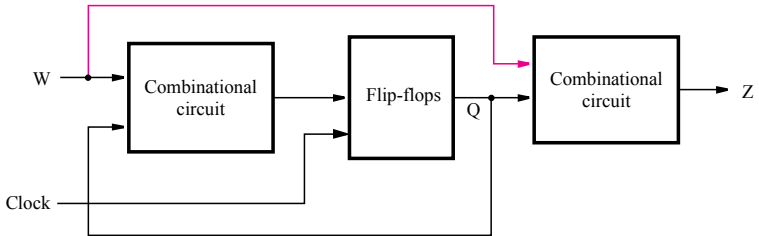
### Flip-Flops e Latches

### Declaração Latch

### Descrição síncrona

### exemplos

- ▶ Para a síntetização de circuitos síncronos, a ferramenta irá utilizar de elementos de memória como Flip-Flop, memórias RAM e ROM.
- ▶ Porém temos que mostrar para o sintetizador que a região descrita deve ser entendida como síncrona.



Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

Principais diferenças entre Flip-Flop e Latches são:

- ▶ Flip-flops são sensíveis a transições, latches são sensíveis a níveis
- ▶ Use preferencialmente em projetos com FPGA os flip-flops;
- ▶ Latches apresentam problema com temporização (veremos mais tarde)



## Conceitos

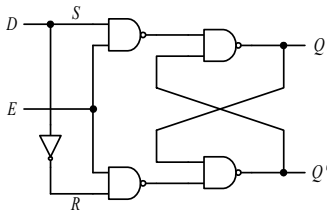
## Circuitos Síncronos

## Flip-Flops e Latches

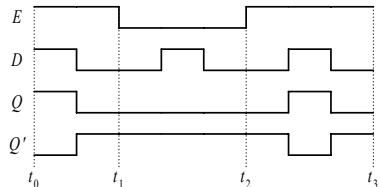
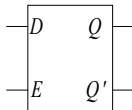
## Declaração Latch

## Descrição síncrona

## exemplos



$E$	$D$	$Q$	$Q_{next}$	$Q_{next}'$
0		0	0	1
0		1	1	0
1	0		0	1
1	1		1	0



Conceitos

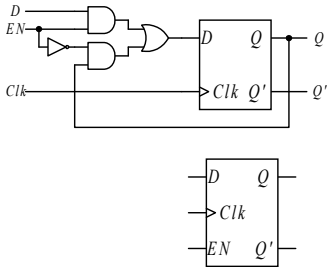
Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos



$Clk$	$EN$	$D$	$Q$	$Q_{next}$	$Q_{next}'$
0			0	0	1
0			1	1	0
1			0	0	1
1			1	1	0
$\uparrow$	0		0	0	1
$\uparrow$	0		1	1	0
$\uparrow$	1	0		0	1
$\uparrow$	1	1		1	0

Conceitos

Circuitos  
Síncronos

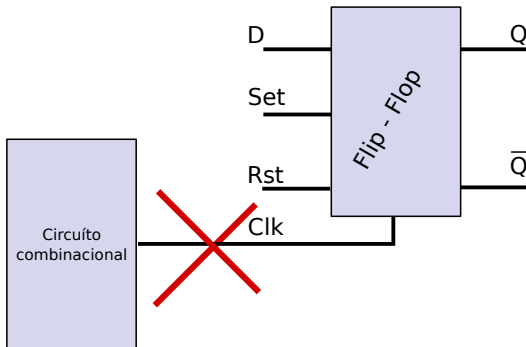
Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

Nunca deve-se usar uma lógica para gerar o sinal para trigar um clock:



Conceitos

Circuitos  
Síncronos

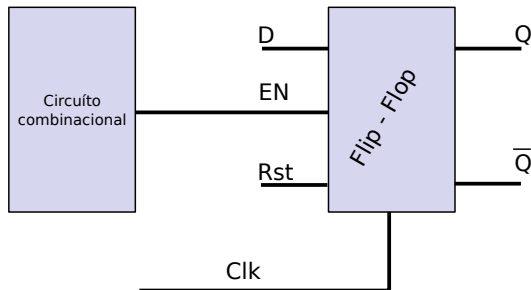
Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

Porém, deve-se utilizar o circuito combinacional como enable ou rst para o flip-flop



Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

**Declaração  
Latch**

Descrição  
síncrona

exemplos

1 Conceitos

2 Circuitos Síncronos  
■ Flip-Flops e Latches

**3 Declaração Latch**

4 Descrição síncrona

5 exemplos

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

```
ENTITY latch IS
    PORT(
        rst : IN  STD_LOGIC; -- entradas
        set  : IN  STD_LOGIC;
        d    : IN  STD_LOGIC;
        q    : OUT STD_LOGIC
    );
END latch;

ARCHITECTURE bhv OF latch IS
BEGIN

    PROCESS(rst, en, d)
    BEGIN
        IF rst = '0' THEN
            q <= '0';
        ELSIF set = '1' THEN
            q <= d;
        END IF;
    END PROCESS;
END bhv;
```

Conceitos

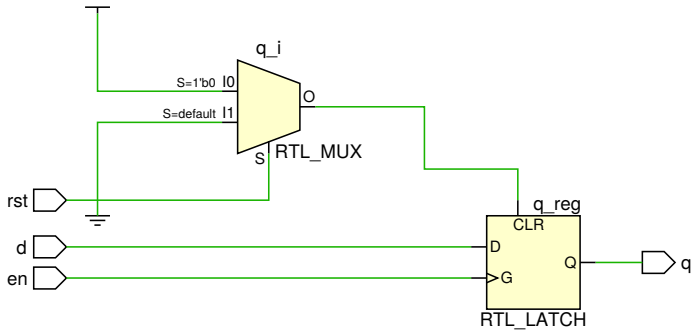
Circuitos  
Síncronos

Flip-Flops e  
Latches

**Declaração  
Latch**

Descrição  
síncrona

exemplos



Conceitos

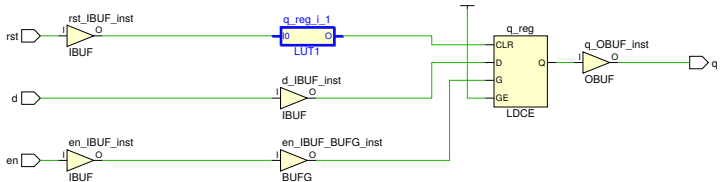
Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos



LDCE : Transparent Data Latch with Asynchronous Clear and Gate Enable



Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

1 Conceitos

2 Circuitos Síncronos  
■ Flip-Flops e Latches

3 Declaração Latch

4 Descrição síncrona

5 exemplos

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

Circuitos síncronos em VHDL são implementados em regiões de códigos sequenciais (**process**).

Uma transição em VHDL pode ser detectada de duas maneiras diferentes :

```
(ck'EVENT and CK = '1') -- Borda de subida  
(ck'EVENT and CK = '0') -- Borda de descida  
(rising_edge(ck))       -- Borda de subida  
(falling_edge(ck))      -- Borda de descida
```

Onde **ck** é o um sinal/entrada que propaga o **clock** .

```
entity sincrono is
    Port ( clk : in  std_logic;
          rst : in  std_logic;
          d   : in  std_logic;
          q   : out std_logic
        );
end sincrono;

architecture Behavioral of sincrono is

begin
    process(clk)
    begin
        if(rising_edge(clk)) then
            if(rst = '1') then
                q <= '0';
            else
                q <= d;
            end if;
        end if;
    end process;
end Behavioral;
```

Conceitos

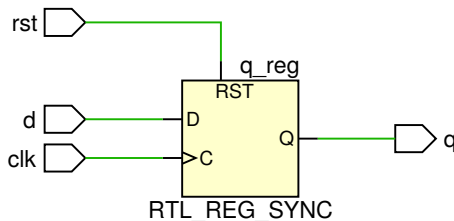
Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

[Conceitos](#)[Circuitos  
Síncronos](#)[Flip-Flops e  
Latches](#)[Declaração  
Latch](#)[Descrição  
síncrona](#)[exemplos](#)

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

```
entity sincrono is
    Port ( clk : in  std_logic;
          rst : in  std_logic;
          d   : in  std_logic;
          q   : out std_logic
        );
end sincrono;

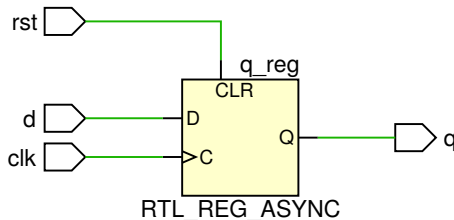
architecture Behavioral of sincrono is

begin
    process(clk, rst)
    begin
        if(rst = '1') then
            q <= '0';
        elsif(clk'EDGE = '1' and clk = '1') then
            q <= d;
        end if;
    end process;
end Behavioral;
```

Conceitos

Circuitos  
SíncronosFlip-Flops e  
LatchesDeclaração  
LatchDescrição  
síncrona

exemplos



Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

```
entity sincrono is
    Port ( clk : in  std_logic;
          rst : in  std_logic;
          set : in  std_logic;
          d   : in  std_logic;
          q   : out std_logic
        );
end sincrono;

architecture Behavioral of sincrono is

begin
    process(clk, rst)
    begin
        if(rst = '1') then
            q <= '0';
        elsif(set = '1') then
            q <= '1';
        elsif(clk'EDGE = '1' and clk = '1') then
            q <= d;
        end if;
    end process;
end Behavioral;
```

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

**exemplos**

1 Conceitos

2 Circuitos Síncronos  
■ Flip-Flops e Latches

3 Declaração Latch

4 Descrição síncrona

**5 exemplos**



Conceitos

Circuitos

Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

```
entity contador is
    Port ( clk : in  std_logic;
          rst : in  std_logic;
          d   : in  std_logic;
          q   : out std_logic
        );
end contador;

architecture Behavioral of contador is

    signal cnt : integer range 0 to 100;

begin
    process(clk, rst)
    begin
        if(rst = '1') then
            cnt <= 0;
        elsif(rising_edge(clk)) then
            cnt <= cnt + 1;
        end if;
    end process;
end Behavioral;
```

# Pisca led a cada 1 segundo com clk de 100Mhz

Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

```
entity contador_led is
    Port ( clk : in  std_logic; -- 100Mhz
          rst : in  std_logic;
          led : out std_logic
        );
end contador_led;

architecture Behavioral of contador_led is

    signal cnt    : integer range 0 to 100_000_000 := 0;
    signal led_i  : std_logic := '0';

begin
    process(clk, rst)
    begin
        if(rst = '1') then
            cnt <= 0;
        elsif(rising_edge(clk)) then
            if (cnt = 100_000_000) then
                cnt <= 0;
                led_i <= not led_i;
            else
                cnt <= cnt + 1;
            end if;
        end if;
    end process;

    led <= led_i;

end Behavioral;
```



Conceitos

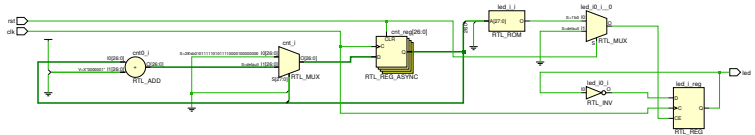
Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos



Conceitos

Circuitos  
Síncronos

Flip-Flops e  
Latches

Declaração  
Latch

Descrição  
síncrona

exemplos

