

# Estratégias de descrição de circuitos síncronos

## Tópicos

- Registrador sensível ao nível
- Registrador sensível à borda – inicialização síncrona
- Registrador sensível à borda – inicialização assíncrona
- Registrador sensível à borda com habilitação para sinal de relógio
- Máquinas de estado finito
- Contadores
- Cuidados na descrição



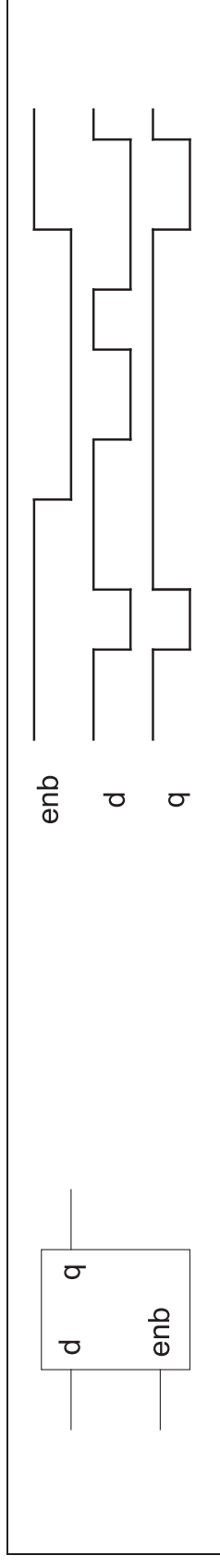
## Estratégias de descrição

- **Descrição visando:** simulação ou teste
  - sem limitações:
    - pode empregar todos os recursos disponíveis na linguagem
- **Descrição visando:** síntese
  - limitada:
    - às estruturas reconhecidas pela ferramenta de síntese
  - procurar utilizar recursos disponíveis na tecnologia-alvo
  - benefícios: circuitos com melhor desempenho  
circuitos com menor custo



## Registrador sensível ao nível – *latch*

- Exemplo:



- **Formato geral:** empregando processo

- sinal **d**: deve estar na lista de sensibilidade

```
PROCESS (enb, d)
BEGIN
  IF (enb = '1' ) THEN  q <= d ;
  END IF ;
END PROCESS;
```

## Registrador sensível ao nível – *latch*

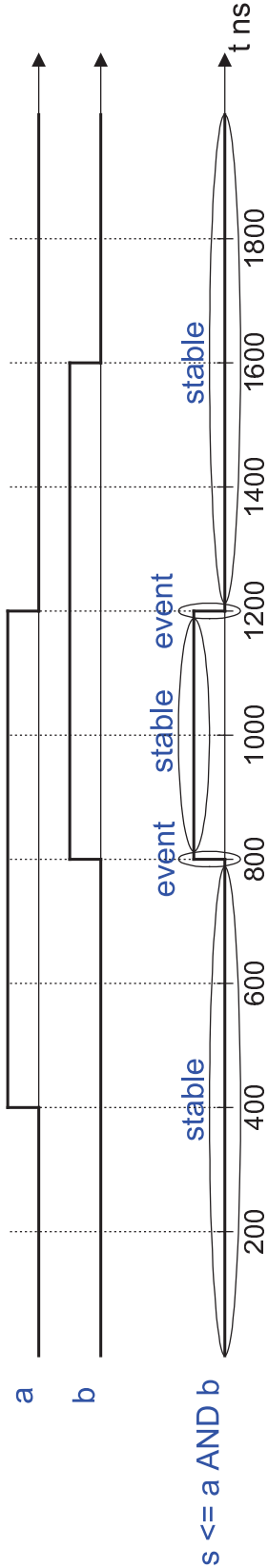
- **Exemplo:** *reset* e *set* assíncronos (devem ser incluídos na lista de sensibilidade)

|    |                                    |
|----|------------------------------------|
| 1  | ENTITY latch3_1 IS                 |
| 2  | PORT (en : IN BIT;                 |
| 3  | rst : IN BIT;                      |
| 4  | set : IN BIT;                      |
| 5  | d : IN BIT_VECTOR(2 DOWNTO 0);     |
| 6  | q : OUT BIT_VECTOR(2 DOWNTO 0));   |
| 7  | END latch3_1;                      |
| 8  |                                    |
| 9  | ARCHITECTURE teste OF latch3_1 IS  |
| 10 | BEGIN                              |
| 11 | PROCESS (en, d, rst, set)          |
| 12 | BEGIN                              |
| 13 | IF (rst = '1') THEN q <= "000";    |
| 14 | ELSIF (set = '1') THEN q <= "111"; |
| 15 | ELSIF (en = '1') THEN q <= d;      |
| 16 | END IF;                            |
| 17 | END PROCESS;                       |
| 18 | END teste;                         |



## Recordando atributos

|          |                                                                                            |
|----------|--------------------------------------------------------------------------------------------|
| s'STABLE | <div>- verdadeiro: <u>não ocorreu uma troca de valor</u><br/>- falso: caso contrário</div> |
| s'EVENT  | <div>- verdadeiro: <u>ocorreu uma troca de valor</u><br/>- falso: caso contrário</div>     |



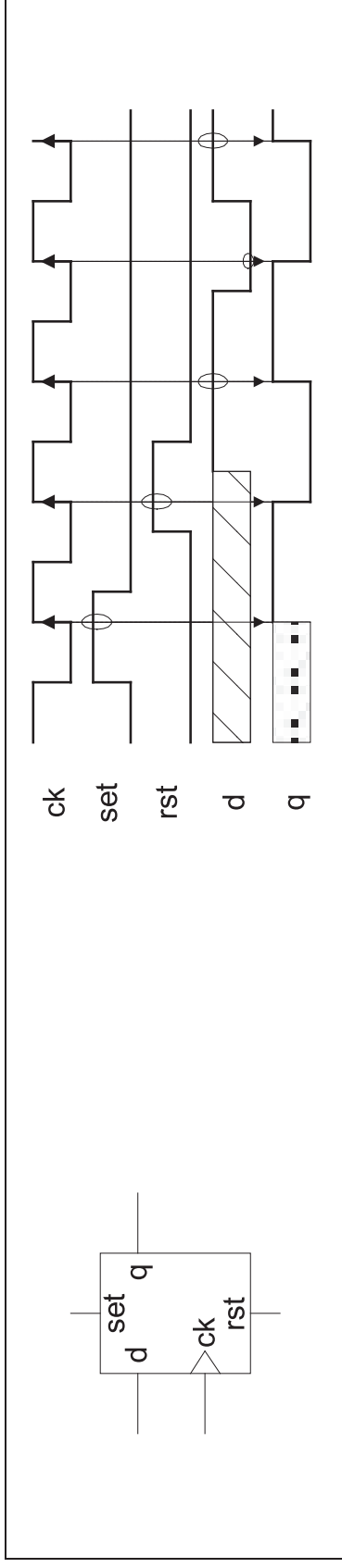
## • Aplicação

- verificação de bordas de subida ou descida em sinais

|                |     |           |                     |
|----------------|-----|-----------|---------------------|
| (ck'EVENT      | AND | ck = '1') | -- borda de subida  |
| (ck'EVENT      | AND | ck = '0') | -- borda de descida |
| (NOT ck'STABLE | AND | ck = '1') | -- borda de subida  |
| (NOT ck'STABLE | AND | ck = '0') | -- borda de descida |



## Registrador sensível à borda – inicialização síncrona (exemplo)



- **Formato geral:** empregando processo

- apenas sinal **ck**: necessita estar na lista de sensibilidade

```
PROCESS (ck)
BEGIN
    IF (ck'EVENT and ck = '1') THEN      -- detecta borda de subida do sinal ck
        IF rst = '1' THEN q <= '0';        -- sequencia de eventos sincronos com ck
        ELSIF set = '1' THEN q <= '1';
        .
        .
        .
        ELSE q <= d;                       -- finaliza com a memorizacao
    END IF;
    END IF;
END PROCESS;
```

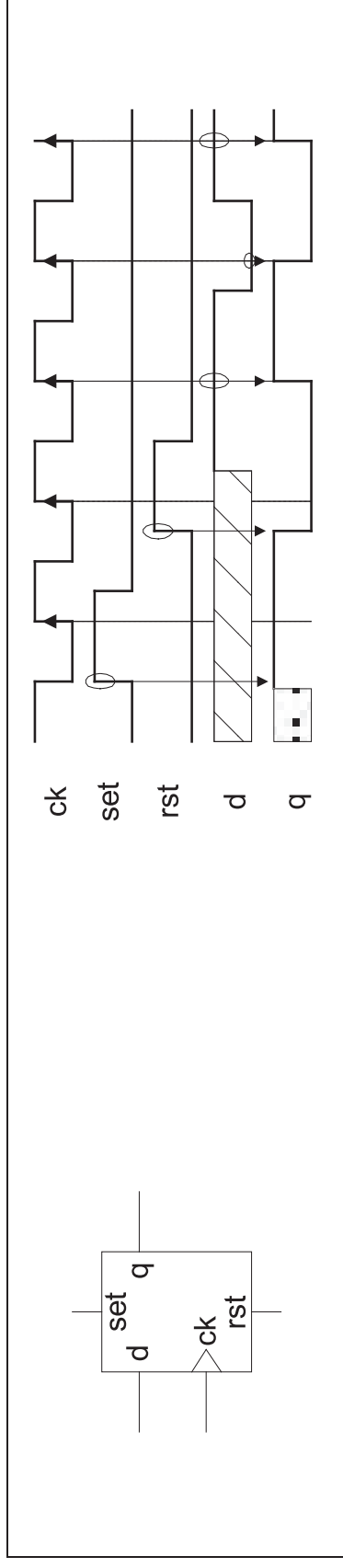
## Registrador sensível à borda – inicialização síncrona (descrição)

- Exemplo:
  - 3 bits
  - *reset* e *set* síncronos (não é necessária a inclusão na lista de sensibilidade)

```
1 ENTITY flip3_2 IS
2   PORT (ck : IN BIT;
3         rst : IN BIT;
4         set : IN BIT;
5         d   : IN BIT_VECTOR(2 DOWNTO 0);
6         q   : OUT BIT_VECTOR(2 DOWNTO 0));
7 END flip3_2;
8
9 ARCHITECTURE teste OF flip3_2 IS
10 BEGIN
11   PROCESS (ck)
12   BEGIN
13     IF (ck'EVENT AND ck = '1') THEN
14       IF (rst = '1') THEN q <= "000";
15       ELSEIF (set = '1') THEN q <= "111";
16       ELSE
17         q <= d;
18       END IF;
19     END IF;
20   END PROCESS;
21 END teste;
```



## Registrador sensível à borda – inicialização assíncrona (exemplo)



- **Formato geral:** empregando processo

- sinais **ck**, **rst** e **set** necessitam estar na lista de sensibilidade

```
PROCESS (ck, rst, set)
BEGIN
    IF      (rst = '1')      THEN q <= '0'; -- eventos assíncronos
    ELSIF   (set = '1')      THEN q <= '1'; --
                                .          .
                                .          .
                                .          .
    ELSEIF  (ck'EVENT AND ck = '1') THEN q <= d; -- detecta borda de subida de ck
    END IF;
END PROCESS;
```

## Registrador sensível à borda – inicialização assíncrona

- Exemplo:
  - 3 bits
  - **reset set** assíncronos (é necessária a inclusão na lista de sensibilidade)

```
1 ENTITY flip3_3 IS
2   PORT (ck : IN BIT;
3         rst : IN BIT;
4         set : IN BIT;
5         d   : IN BIT_VECTOR(2 DOWNTO 0);
6         q   : OUT BIT_VECTOR(2 DOWNTO 0));
7 END flip3_3;
8
9 ARCHITECTURE teste OF flip3_3 IS
10 BEGIN
11   PROCESS (ck, rst, set)
12   BEGIN
13     IF      (rst = '1')      THEN q <= "000"; -- q=000 independente de ck
14     ELSIF   (set = '1')      THEN q <= "111"; -- q=111 independente de ck
15     ELIF    (ck'EVENT AND ck = '1') THEN q <= d; -- condicao do sinal relógio
16     END IF;
17   END PROCESS;
18 END teste;
```

