

# Componentes

- **Componente:**
  - uma descrição (entidade + arquitetura) empregada por outra entidade
- **Uso:**
  - interligação de múltiplas entidades de projeto
    - projeto hierárquico
- **Declaração de um componente** (primeiro passo para a utilização de um componente)
  - similar à declaração de entidade
  - exemplo:

```
COMPONENT nome_componente
PORT (sinal_a      : modo_a  tipo_sinal_a;
      sinal_b      : modo_b  tipo_sinal_b;
      sinal_c      : modo_c  tipo_sinal_c);
END COMPONENT;
```



## Componentes

- **A solicitação de um componente contém:**

- um rótulo de denominação qualquer (**x1, y2 ...**)
- nome do componente (**nome\_componente**)
- mapa de ligações (**Port Map (...)**)

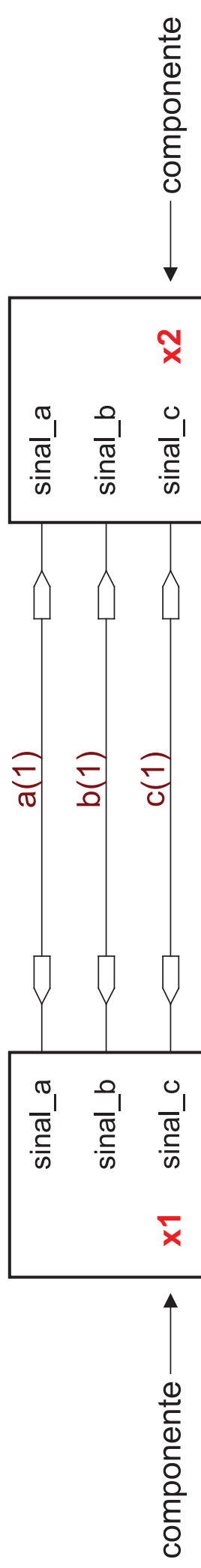
- **Mapa de ligações:**

- pode seguir a mesma ordem estabelecida na declaração do componente:

```
x1: nome_componente PORT MAP (a(1), b(1), c(1));
```

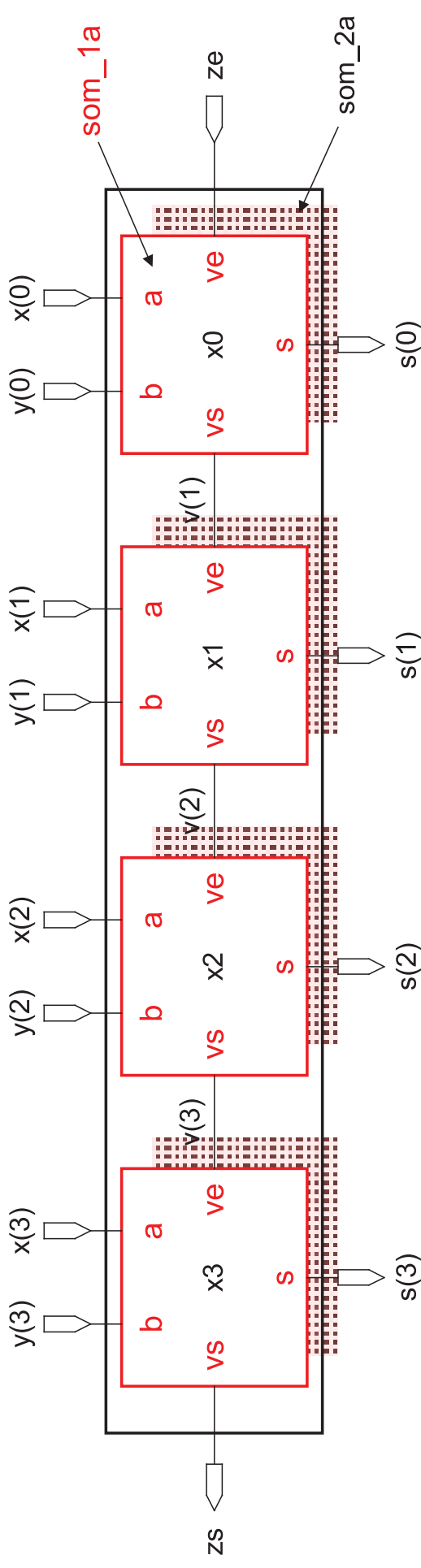
- pode seguir uma nova sequência definida no mapa:

```
x2: nome_componente PORT MAP (sinal_b => b(1), sinal_a => a(1), sinal_c => c(1));
```



## Componentes

- **Exemplo:** Somador de 4 bits formado por somadores de 1 bit



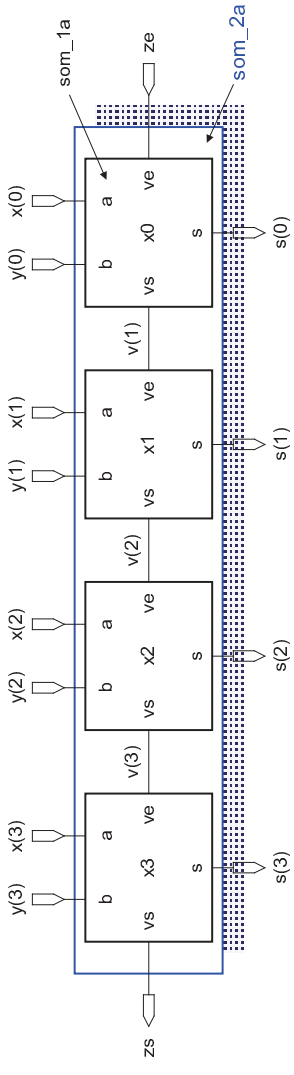
- **Descrição do somador de 1 bit** (componente a ser solicitado)

```
1 ENTITY som_1a IS
2   PORT (a, b, ve : IN BIT;
3         s, vs : OUT BIT);
4 END som_1a;
5
6 ARCHITECTURE teste OF som_1a IS
7
8 BEGIN
9   s <= a XOR b XOR ve;
10  vs <= (a AND b) OR (a AND ve) OR (b AND ve); -- soma
11 END teste;
```



# Componentes

- Descrição do somador empregando o componente: somador de 1 bit



```
1 ENTITY som_2a IS
2   PORT ( x, y : IN BIT_VECTOR (3 DOWNTO 0); -- entradas do somador
3         ze : IN BIT; -- entrada vem um
4         s : OUT BIT_VECTOR (3 DOWNTO 0); -- soma
5         zs : OUT BIT ); -- vai um
6 END som_2a;
7
8 ARCHITECTURE estrutural OF som_2a IS
9
10  COMPONENT som_1a
11  PORT (a, b, ve : IN BIT; s, vs : OUT BIT);
12  END COMPONENT;
13
14  SIGNAL v : BIT_VECTOR (3 DOWNTO 1); -- vai um interno
15
16 BEGIN
17  x0: som_1a PORT MAP ( x(0), ze, s(0), v(1) );
18  x1: som_1a PORT MAP ( x(1), v(1), s(1), v(2) );
19  x2: som_1a PORT MAP (b =>y(2), a =>x(2), s =>s(2), ve =>v(2), vs =>v(3) );
20  x3: som_1a PORT MAP ( x(3), v(3), s(3), zs );
21 END estrutural;
```