

زبان توصیف سخت افزار

Hardware Description Language

عملکرد همروند و عملکرد ترتیبی

□ عملکرد ترتیبی:

- اجرا به ترتیب از بالا به پایین
- مانند اکثر زبان‌های نرم‌افزاری
- برای عملکرد ترتیبی، در بدنه‌های ترتیبی
 - process
 - procedure
 - function

□ عملکرد همروند:

- عملکرد واقعی سخت‌افزار
- عملکرد پیش‌فرض در architecture
-

عملکرد همروند و عملکرد ترتیبی

□ دستورهای ترتیبی:

– فقط در بدنه ترتیبی

– ایجاد یک پودمان (module instantiation)

– انتساب به سیگنال

– فرایند (process)

– if/for generate

– when-else

– with-select-when

– فراخوانی روال

– فراخوانی تابع

عملکرد همروند و عملکرد ترتیبی

دستورهای همروند: 

- فقط در بدنه همروند

- if-then-else

- حلقه for

- حلقه while

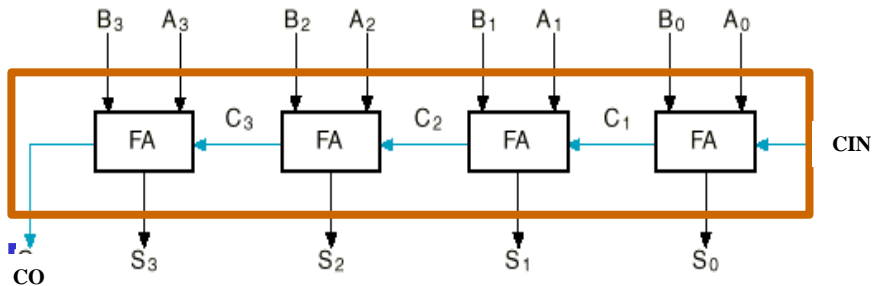
- case-when

- فراخوانی روال

- فراخوانی تابع

ساختارهای منظم

• for-generate



□ برای توصیف طرح‌های منظم

```
entity FOUR_BIT_ADDER is
    port (A , B : in bit_vector (3 downto 0);
          CIN : in bit;
          S : out bit_vector (3 downto 0);
          COUT : out bit);
end entity FOUR_BIT_ADDER;
--
architecture STRUCT_FA of FOUR_BIT_ADDER is
    component FULL_ADDER port (X , Y , CI : in bit ; Z , CO : out bit);
    end component;

    signal C : bit_vector (3 downto 1);
begin
    FA0: FULL_ADDER port map (A(0), B(0), CIN, S(0), C(1));
    FA1: FULL_ADDER port map (A(1), B(1), C(1), S(1), C(2));
    FA2: FULL_ADDER port map (A(2), B(2), C(2), S(2), C(3));
    FA3: FULL_ADDER port map (A(3), B(3), C(3), S(3), COUT);
end architecture STRUCT_FA;
```

ساختارهای منظم

```
FA0: FULL_ADDER port map (A(0), B(0), CIN, S(0), C(1));  
FA1: FULL_ADDER port map (A(1), B(1), C(1), S(1), C(2));  
FA2: FULL_ADDER port map (A(2), B(2), C(2), S(2), C(3));  
FA3: FULL_ADDER port map (A(3), B(3), C(3), S(3), COUT);
```

```
entity FOUR_BIT_ADDER is  
...  
end entity FOUR_BIT_ADDER;  
--  
architecture STRUCT_FA of FOUR_BIT_ADDER is  
...  
begin  
    FA0: FULL_ADDER port map (A(0), B(0), CIN, S(0), C(1));  
    FA3: FULL_ADDER port map (A(3), B(3), C(3), S(3), COUT);  
    F: for I in 2 downto 1 generate  
        FAS: FULL_ADDER port map(A(I), B(I), C(I), S(I), C(I+1));  
    end generate F;  
end architecture STRUCT_FA;
```

ساختارهای منظم

```
FA0: FULL_ADDER port map (A(0), B(0), CIN, S(0), C(1));
FA1: FULL_ADDER port map (A(1), B(1), C(1), S(1), C(2));
FA2: FULL_ADDER port map (A(2), B(2), C(2), S(2), C(3));
FA3: FULL_ADDER port map (A(3), B(3), C(3), S(3), COUT);
```

```
entity FOUR_BIT_ADDER is
...
end entity FOUR_BIT_ADDER;
--
```

```
architecture STRUCT_FOR of FOUR_BIT_ADDER is
```

```
signal C : bit_vector (4 downto 0);
```

```
begin
```

```
  F: for I in 3 downto 0 generate
```

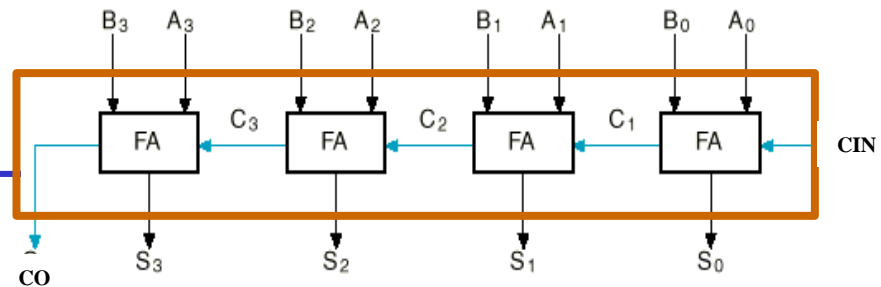
```
    FAS: FULL_ADDER port map(A(I), B(I), C(I), S(I), C(I+1));
```

```
  end generate F;
```

```
  COUT <= C(4);
```

```
  C(0) <= CIN;
```

```
end architecture STRUCT_FOR;
```



ساختارهای منظم

• if-generate

```
FA0: FULL_ADDER port map (A(0), B(0), CIN, S(0), C(1));  
FA1: FULL_ADDER port map (A(1), B(1), C(1), S(1), C(2));  
FA2: FULL_ADDER port map (A(2), B(2), C(2), S(2), C(3));  
FA3: FULL_ADDER port map (A(3), B(3), C(3), S(3), COUT);
```

```
entity FOUR_BIT_ADDER is  
...  
end entity FOUR_BIT_ADDER;  
--  
architecture STRUCT_FOR of FOUR_BIT_ADDER is  
    signal C : bit_vector (2 downto 1);  
begin  
    F: for I in 3 downto 0 generate  
        F_IF0: if I=0 generate  
            L0:FULL_ADDER port map(A(0), B(0), CIN, S(0), C(1));  
        end generate F_IF0;  
        F_IF3 : if I=3 generate  
            L3:FULL_ADDER port map(A(3), B(3), C(3), S(3), COUT);  
        end generate F_IF3;  
        F_IF_1_2 : if I/=3 and I/=0 generate  
            L_1_2:FULL_ADDER port map(A(I), B(I), C(I), S(I), C(I+1));  
        end generate F_IF_1_2;  
    end generate F;  
end architecture STRUCT_FOR;
```


طراحی به صورت کلی

• طراحی کلی:

□ محدود نبودن به حالت خاص

- شمارنده صفر تا $n-1$

- جمع کننده n -بیتی

- مقایسه کننده n -بیتی

طراحی به صورت کلی

• جمع کننده n -بیتی:

با ثابت‌ها 

```
constant N : integer :=4;
entity N_BIT_ADDER IS
    port( A , B : in bit_vector (N-1 downto 0);
          CIN : in bit;
          S : out bit_vector(N-1 downto 0);
          COUT : out bit);
end entity N_BIT_ADDER;

architecture STRUCT_FOR of N_BIT_ADDER is
    component FULL_ADDER port (X , Y , CI : in bit ; Z , CO : out bit);
    end component;
    signal C : bit_vector (N-1 downto 0);
begin
    FA1: FULL_ADDER port map (A(0), B(0), CIN, S(0), C(1));
    FA_N: FULL_ADDER port map (A(N-1), B(N-1), C(N-1), S(N-1), COUT );
    L: for I in N-2 downto 1 generate
        FAS: FULL_ADDER port map(A(I), B(I), C(I), S(I), C(I+1));
    end generate L;
end architecture STRUCT_FOR;
```

طراحی به صورت کلی

- جمع کننده n -بیتی:

□ با ثابت‌ها

- اشکال:

- برای هر مورد، نیاز به کامپایل مجدد

- برای چند مورد، نیاز به کپی کد

طراحی به صورت کلی

با پارامتر generic 


```
entity N_BIT_ADDER IS
  generic (N : integer :=4);
  port(A ,B : in bit_vector (N-1 downto 0);
        CIN : in bit;
        S : out bit_vector(N-1 downto 0);
        COUT : out bit);

end entity N_BIT_ADDER;

architecture STRUCT_FOR of N_BIT_ADDER is
  component FULL_ADDER port (X , Y , CI : in bit ; Z , CO : out bit);
  end component;
  signal C : bit_vector (N-1 downto 0);
begin
  FA1: FULL_ADDER port map (A(0) , B(0) , CIN, S(0) , C(1));
  FA_N: FULL_ADDER port map (A(N-1) ,B(N-1) ,C(N-1) ,S(N-1) ,COUT );
  L: for I in N-2 downto 1 generate
    FAS: FULL_ADDER port map(A(I) ,B(I) ,C(I) ,S(I) ,C(I+1));
  end generate L;
end architecture STRUCT_FOR;

...
begin
  INST1:N_BIT_ADDER generic map (8) port map (...);
  INST2:N_BIT_ADDER generic map (64) port map (...);
end architecture STRUCT_FOR;
```

طراحی به صورت کلی

- بدون مشخص کردن بازه درگاه‌ها (با) 
- چگونه اندیس حلقه for-generate را بدهیم؟
- چگونه در instantiation مشخص کنیم کدام (attribute)
- است؟
- برای سیگنال‌ها نمی‌توان نام مشخص گذاشت.

```
entity GENERAL_ADDER IS
    port(A ,B : in bit_vector (N-1 downto 0);
          CIN : in bit;
          S : out bit_vector (N-1 downto 0);
          COUT : out bit);
end entity GENERAL_ADDER;

architecture ARCH1 of GENERAL_ADDER is
    component FULL_ADDER port (X , Y , CI : in bit ; Z , CO : out bit);
    end component;
    signal C : bit_vector (A'range);
begin
    FA1: FULL_ADDER port map (A(A'low), B(A'low), CIN, S(A'low), C(A'low+1));
    FA_N: FULL_ADDER port map (A(A'high), B(A'high), C(A'high), S(A'high), COUT
    );
    L: for I in high-1 downto A'low+1 generate
        FAS: FULL_ADDER port map(A(I), B(I), C(I), S(I), C(I+1));
    end generate L;
end architecture ARCH1;
```

طراحی به صورت کلی

بدون مشخص کردن بازه
درگاهها (با attribute)

```
architecture ...  
signal S1, S2, S3: bit_vector (7 downto 0);  
signal S4, S5, S6: bit_vector (3 downto 0);  
signal CIN1, CIN2, COUT1, COUT2: bit;  
begin  
...  
    EIGHTBIT:GENERAL_ADDER port map (S1, S2, CIN1, S3, COUT1);  
    FOURBIT:GENERAL_ADDER port map (S4, S5, CIN2, S6, COUT2);  
end architecture STRUCT_FOR;
```

اگر درگاهها به درگاههای سطح بالاتر وصل بود چطور؟