



تصميم رقمي متقدم
Advanced Digital Design

Dr.-Eng. Samer Sulaiman

2020-2021

مفردات المنهاج

- أساسيات التصميم الرقمي
- عناصر وتقنيات التصميم الرقمي التوافقي والتعاقبي (المتسلسل)
- **نمذجة التصميم الرقمي باستعمال لغة توصيف الكيان الصلب VHDL**
- المحاكاة الوظيفية والزمنية للأنظمة الرقمية



تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• التعليمة Process:

- إن تعليمة Process من التعليمات التفرعية في لغة VHDL
- تشير إلى أن كل ما يكتب ضمنها يتم تنفيذ بصورة تتابعية وعادة ما تكتب التعليمات IF, WAIT, CASE, LOOP والتي تنفذ بصورة تسلسلية من الأعلى إلى الأسفل وبصورة دورية مادامت تعليمة Process فعالة
- تتضمن تعليمة Process عدد من المتحولات أو الشروط المنطقية تسمى قائمة الحساسية sensitivity list والتي تعتبر مجموعة من المتحولات لا يتم الدخول إلى كتلة التعليمة Process إلا عند حدوث تغيير على هذه المتحولات
- تكتب تعليمة Process في الكود الرئيسي وتنفذ كل مرة يتغير فيها أحد متحولات الحساسية أو الشروط المنطقية
- الصيغة العامة لهذه التعليمة مع ملاحظة أن كل ما يكتب ضمن [] هو اختياري:

- [label:] PROCESS (sensitivity list)
[VARIABLE name type [range] [:= initial_value;]]
BEGIN
(sequential code)
END PROCESS [label];

تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• التعليمة :Process

• مثال: المطلوب تصميم دائرة XOR تعطي خرج في حال تغير احدى مداخلها

```
• library IEEE;
  use IEEE.STD_LOGIC_1164.ALL;
  entity my_xor is
    port ( A,B : in std_logic;
          F : out std_logic);
  end my_xor;
  architecture behav of my_xor is
  begin
    xor_proc: process(A,B) is
    begin
      F <= A XOR B;
    end process xor_proc;
  end behav;
```



تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

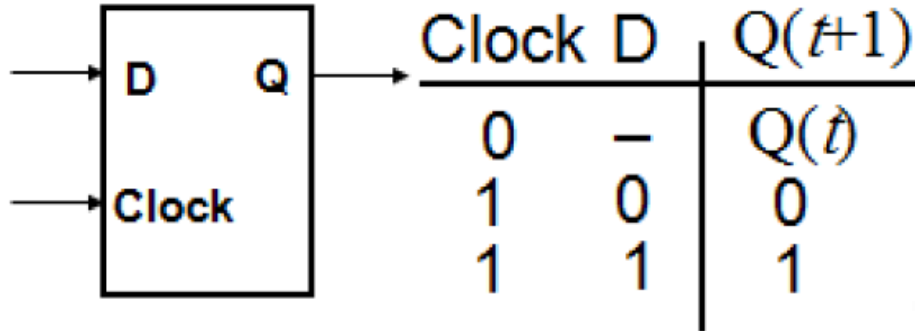
• التعليمة IF:

- أيضاً من التعليمات التتابعية التي تستخدم في حالات التفريع المشروط للبرنامج ،
- تستخدم ضمن تعليمة Process
- الصيغة العامة لها من الشكل :

- IF conditions THEN assignments;
ELSIF conditions THEN assignments;
..
ELSE assignments;
END IF;

- و من أهم التطبيقات والدارات التي توضح عمل هذه التعليمات هي المسجلات و العدادات
- أمثلة:

تصميم الأنظمة الرقمية باستخدام VHDL



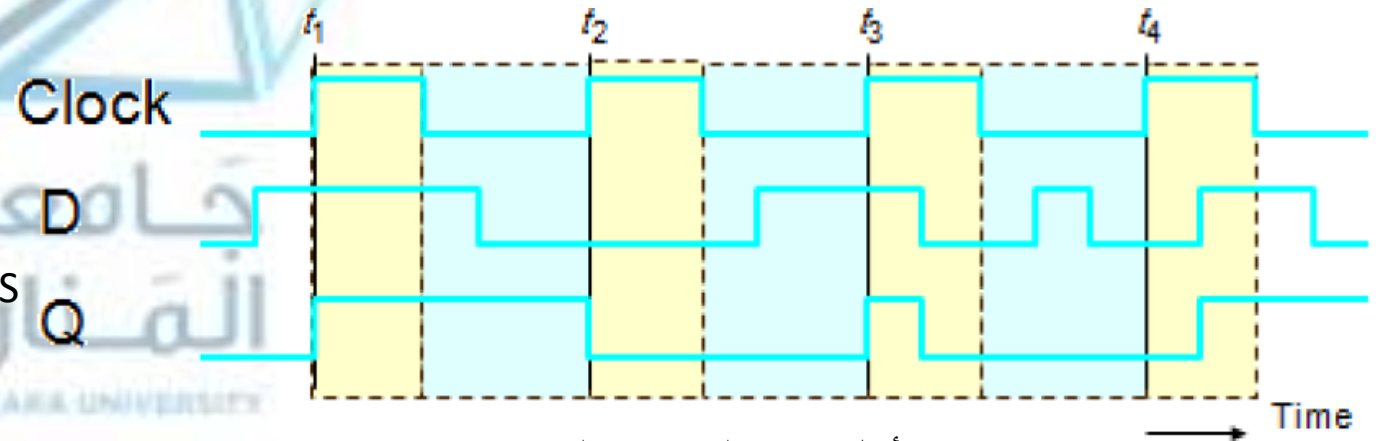
• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• التعليمة IF:

• مثال: المطلوب تصميم دائرة الماسك (D latch) الموضحة بالشكل التالي:

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY latch IS
    PORT (D, Clock: IN STD_LOGIC;
          Q: OUT STD_LOGIC);
END latch;
ARCHITECTURE behavioral OF latch IS
BEGIN
    PROCESS (D, Clock)
    BEGIN
        IF Clock = '1' THEN
            Q <= D;
        END IF;
    END PROCESS;
END behavioral;
    
```



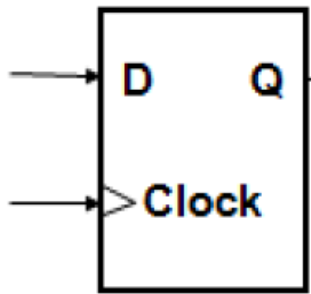
من أجل تحديد التغير في النبضة ('1') - - if(Clock'event and Clock='1')

END IF ;

END PROCESS ;

END behavioral;

تصميم الأنظمة الرقمية باستخدام VHDL



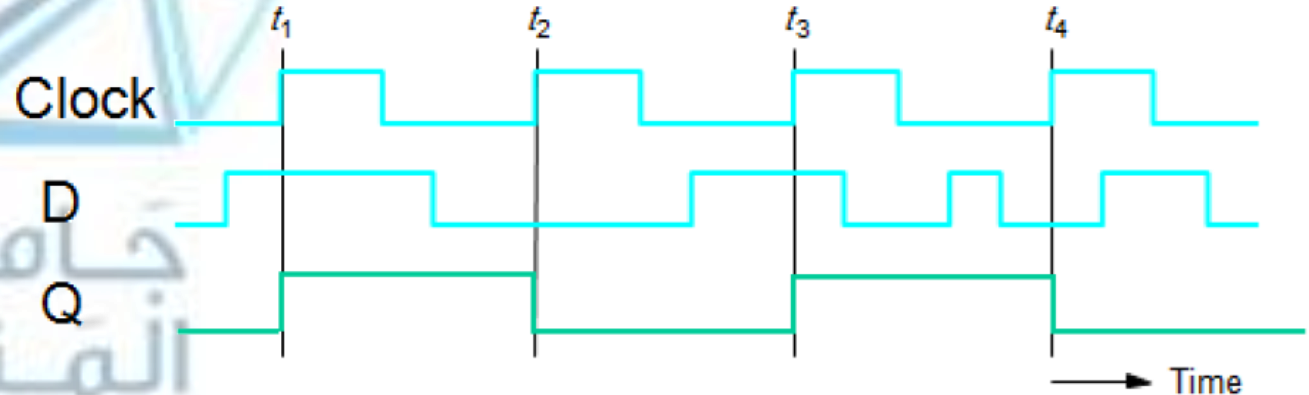
Clk	D	Q(t+1)
↑	0	0
↑	1	1
0	—	Q(t)
1	—	Q(t)

- تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:
- التعليمة IF:

• مثال: المطلوب تصميم دائرة القلاب (D flip-flop) الموضحة بالشكل التالي:

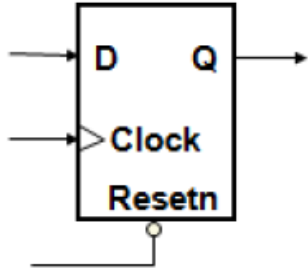
```

LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
ENTITY flipflop IS
    PORT (D, Clock: IN STD_LOGIC ;
          Q: OUT STD_LOGIC);
END flipflop ;
ARCHITECTURE behavioral OF flipflop IS
BEGIN
    PROCESS ( Clock )
    BEGIN
        IF Clock'EVENT AND Clock = '1' THEN -- IF rising_edge(Clock) THEN
            Q <= D ;
        END IF ;
    END PROCESS ;
END behavioral ;
    
```



طريقة أخرى لتوصيف النبضة الصاعدة IF rising_edge(Clock) THEN

تصميم الأنظمة الرقمية باستخدام VHDL

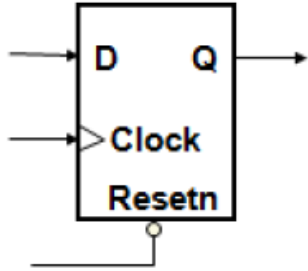


- تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:
- التعليمة IF:

• مثال: المطلوب تصميم دائرة القلاب (D flip-flop) السابق مع قطب تصفير اللامتزامن (asynchronous reset):

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
ENTITY flipflop_ar IS
    PORT (D, Resetn, Clock: IN STD_LOGIC ;
          Q: OUT STD_LOGIC) ;
END flipflop_ar ;
ARCHITECTURE behavioral OF flipflop_ar IS
BEGIN
    PROCESS ( Resetn, Clock )
    BEGIN
        IF Resetn = '0' THEN
            Q <= '0' ;
        ELSIF rising_edge(Clock) THEN
            Q <= D ;
        END IF ;
    END PROCESS ;
END behavioral ;
```


تصميم الأنظمة الرقمية باستخدام VHDL



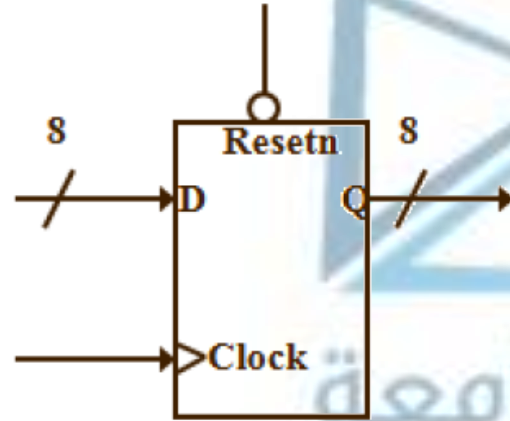
- تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:
- التعليمة IF:

• مثال: المطلوب تصميم دائرة القلاب (D flip-flop) السابق مع قطب تصفير متزامن (synchronous):

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
ENTITY flipflop_sr IS
    PORT (D, Resetn, Clock: IN STD_LOGIC ;
          Q: OUT STD_LOGIC) ;
END flipflop_sr ;
ARCHITECTURE behavioral OF flipflop_sr IS
BEGIN
    PROCESS(Clock)
    BEGIN
        IF rising_edge(Clock) THEN
            IF Resetn = '0' THEN
                Q <= '0' ;
            ELSE
                Q <= D ;
            END IF ;
        END IF ;
    END PROCESS ;
END behavioral ;
```

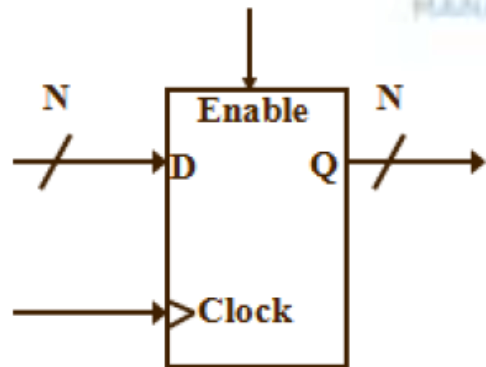
تصميم الأنظمة الرقمية باستخدام VHDL

- تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:
• التعليمة IF:



- وظيفة: المطلوب تصميم دائرة المسجل (Register) مع قطب تصفير لامترامن ولـ 8 خانات والموضح بالشكل:

- وظيفة: المطلوب تصميم دائرة المسجل (Register) السابق مع قطب تصفير لامترامن ولـ n خانة



- وظيفة: المطلوب تصميم دائرة المسجل (Register) مع قطب تفعيل لامترامن ولـ n خانات والموضح بالشكل:

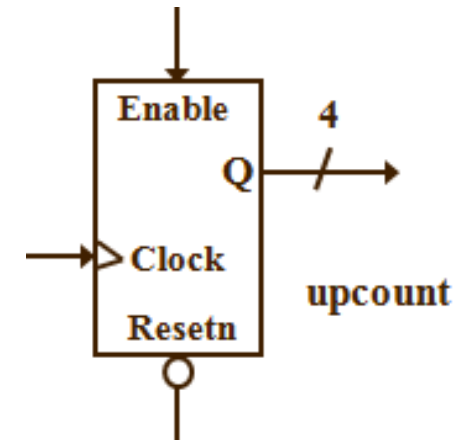
تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• التعليمة IF:

• مثال: المطلوب تصميم دائرة عداد تصاعدي بأربع خانات مع قطب تصفير لامتزامن:

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
USE ieee.std_logic_unsigned.all ;
ENTITY upcount_ar IS
    PORT (Clock, Resetn, Enable : IN STD_LOGIC ;
          Q: OUT STD_LOGIC_VECTOR (3 DOWNTO 0));
END upcount_ar ;
ARCHITECTURE behavioral OF upcount_ar IS
    SIGNAL Count : STD_LOGIC_VECTOR(3 DOWNTO 0);
BEGIN
    PROCESS ( Clock, Resetn )
    BEGIN
        IF Resetn = '0' THEN
            Count <= "0000" ;
        ELSIF rising_edge(Clock) THEN
            IF Enable = '1' THEN
                Count <= Count + 1 ;
            END IF ;
        END IF ;
    END PROCESS ;
    Q <= Count ;
END behavioral ;
```



تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• التعليمة IF:

• مثال: المطلوب تصميم دائرة عداد رقمي صاعد حلقي من القيمة 0 إلى القيمة 9 بحيث يزداد خرجة قيمة واحدة عند كل جبهة صاعدة لنبضات الساعة clk (عداد عشري حلقي):

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY counter IS
    PORT (clk : IN STD_LOGIC;
          digit : OUT INTEGER RANGE 0 TO 9);
END counter;
ARCHITECTURE counter OF counter IS
BEGIN
    count: PROCESS(clk)
        VARIABLE temp : INTEGER RANGE 0 TO 10;
    BEGIN
        IF (clk'EVENT AND clk='1') THEN
            temp := temp + 1;
            IF (temp=10) THEN
                temp := 0;
            END IF;
        END IF;
        digit <= temp;
    END PROCESS count;
END counter;
```

تصميم الأنظمة الرقمية باستخدام VHDL

- تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:
- مصادر نبضات الساعة Clock Sources في لوحة الـ FPGA:
- التعليمة Wait:
- من التعليمات التتابعية التي لها عدة أشكال تستخدم لضبط عمليات التنفيذ للتعليمات ضمن كتلة Process،
 - وبوجودها تفقد لائحة الحساسية لتعليمة Process فاعليتها،
 - بالتالي تصبح لائحة الحساسية غير ضرورية.
 - الصيغة العامة لهذه التعليمة لها عدة أشكال :
 - الشكل الأول :
 - الانتظار حتى يتحقق شرط منطقي ما متعلق بإشارة واحدة فقط ،
 - يستعمل كبديل عن لائحة الحساسية ويجب ذكره في بداية كتلة Process،
 - تنفذ كتلة التعليمات في Process في كل مرة يتحقق فيها الشرط
 - مثال:
 - المطلوب تصميم دائرة مسجل بثمانية خانات متزامن مع منفذ تصفير:
- ```
PROCESS -- no sensitivity list
BEGIN
 WAIT UNTIL (clk'EVENT AND clk='1');
 IF (rst='1') THEN
 output <= "00000000";
 ELSIF (clk'EVENT AND clk='1') THEN
 output <= input;
 END IF;
END PROCESS;
```

# تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• التعليمة Wait:

• الصيغة العامة لهذه التعليمة لها عدة أشكال :

• الشكل الثاني:

• الانتظار حتى يطرأ تغيير ما على قيمة إحدى الإشارات لتنفيذ التعليمات

• مثال:

• المطلوب تصميم دائرة مسجل بثمانية خانات متزامن مع منفذ تصفير:

• PROCESS  
BEGIN

WAIT ON clk, rst;

IF (rst='1') THEN

output <= "00000000";

ELSIF (clk'EVENT AND clk='1') THEN

output <= input;

END IF;

END PROCESS;

• الشكل الثالث:

• يستخدم فقط لأغراض المحاكاة

• وظيفته الانتظار لفترة زمنية ما .

• مثال عن الصيغة المستخدمة ; WAIT FOR 5ns

# تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• التعليمة Wait:

• مثال:

• المطلوب تصميم دائرة قلاب من النوع D باستخدام تعليمة WAIT:

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY dff IS
 PORT (d, clk, rst: IN STD_LOGIC;
 q: OUT STD_LOGIC);
END dff;
ARCHITECTURE dff1 OF dff IS
BEGIN
 PROCESS
 BEGIN
 WAIT ON rst, clk;
 IF (rst='1') THEN
 q <= '0';
 ELSIF (clk'EVENT AND clk='1') THEN
 q <= d;
 END IF;
 END PROCESS;
END dff1;
```

# تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• تعليمة Case:

- من التعليمات التتابعية التي تستخدم ضمن كتلة Process
- تقوم بعمل مشابه للتعليمة When التفرعية،
- مهمتها إسناد قيمة ما لمتحول بحسب شرط ما
- خلافاً للتعليمة When التي لا تسمح بالتعامل إلا مع متحول وحيد، يمكن لهذه التعليمة أن تسند قيمة لعدة إشارات معاً
- مثال:

- CASE control IS  
WHEN "00" => x<=a; y<=b;  
WHEN "01" => x<=b; y<=c;  
WHEN OTHERS => x<="0000"; y<="ZZZZ";  
END CASE;



# تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• تعليمة Case:

• مثال:

• المطلوب تصميم دائرة قلاب من النوع D باستخدام تعليمة CASE:

```
• ENTITY dff IS
 PORT (d, clk, rst: IN BIT;
 q: OUT BIT);
END dff;
ARCHITECTURE dff3 OF dff IS
BEGIN
 PROCESS (clk, rst)
 BEGIN
 CASE rst IS
 WHEN '1' => q<='0';
 WHEN '0' =>
 IF (clk'EVENT AND clk='1') THEN
 q <= d;
 END IF;
 WHEN OTHERS => NULL; -- Unnecessary, rst is of type – BIT
 END CASE;
 END PROCESS;
END dff3;
```

# تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• تعليمة Case:

• مثال:

• ماذا يمثل التوصيف التالي؟

```
• ENTITY test IS
 PORT (clk, reset : IN STD_LOGIC);
END test;
ARCHITECTURE test OF test IS
BEGIN
 PROCESS(clk, reset)
 VARIABLE temp1: INTEGER RANGE 0 TO 10;
 VARIABLE temp2: INTEGER RANGE 0 TO 10;
 BEGIN
 IF (reset='1') THEN
 temp1 := 0;
 temp2 := 0;
 ELSIF (clk'EVENT AND clk='1') THEN
 temp1 := temp1 + 1;
 IF (temp1=10) THEN
 temp1 := 0;
 temp2 := temp2 + 1;
 IF (temp2=10) THEN
 temp2 := 0;
 END IF;
 END IF;
 END IF;
 END IF;
END PROCESS;
END test;
```

# تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• تعليمة Loop:

- من التعليمات التتابعية التي تستخدم ضمن كتلة Process
  - تقوم بتكرار تنفيذ قسم من البرنامج عدة مرات
  - تستخدم تعليمة Exit من داخل الحلقة للخروج من داخل حلقة التكرار عند تحقق شرط ما
  - تستخدم تعليمة Next من داخل الحلقة لتجاوز بعض التعليمات من داخل حلقة التكرار
  - عند تحقق شرط ما
  - لهذه التعليمة عدة أشكال سيتم التعرف عليها من خلال الأمثلة
  - مثال:
- FOR i IN 0 TO 5 LOOP  
Statements;  
END LOOP;

# تصميم الأنظمة الرقمية باستخدام VHDL

## • تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

### • تعليمة Loop:

- لهذه التعليمة عدة أشكال سيتم التعرف عليها من خلال الأمثلة
- مثال:

- WHILE (i < 10) LOOP  
    WAIT UNTIL clk'EVENT AND clk='1';  
    (other statements)  
END LOOP;
- FOR i IN data'RANGE LOOP  
    CASE data(i) IS  
        WHEN '0' => count:=count+1;  
        WHEN OTHERS => EXIT;  
    END CASE;  
END LOOP;
- FOR i IN 0 TO 15 LOOP  
    NEXT WHEN i=skip; -- jumps to next iteration  
    (other statements)  
END LOOP;

# تصميم الأنظمة الرقمية باستخدام VHDL

• تصميم بنى منطقية تسلسلية ضمن اللغة VHDL:

• تعليمة Loop:

• لهذه التعليمة عدة أشكال سيتم التعرف عليها من خلال الأمثلة

• مثال:

• المطلوب تصميم دائرة تقوم بإحصاء عدد الأصفار اليسارية الموجودة في إشارة الدخل Data ، على سبيل المثال إذا كانت قيمة الدخل 00000001 يظهر البرنامج القيمة 7 على الخرج وهكذا ...

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY LeadingZeros IS
 PORT (data: IN STD_LOGIC_VECTOR (7 DOWNT0 0);
 zeros: OUT INTEGER RANGE 0 TO 8);
END LeadingZeros;
ARCHITECTURE behavior OF LeadingZeros IS
BEGIN
 PROCESS (data)
 VARIABLE count: INTEGER RANGE 0 TO 8;
 BEGIN
 count := 0;
 FOR i IN data'RANGE LOOP
 CASE data(i) IS
 WHEN '0' => count := count + 1;
 WHEN OTHERS => EXIT;
 END CASE;
 END LOOP;
 zeros <= count;
 END PROCESS;
END behavior;
```