جامعة نينوى
كلية هندسة الإلكترونيات

Circuit Design with VHDL 9

Textbook: Volnei A. Pedroni

Submitted By: Hussein Aideen

VHDL> Arrays in VHDL

- Arrays are collections of objects of the same type.
- one-dimensional (1D),

0	1	0	0	0

- two-dimensional (2D),

0	1	0	0	0
1	0	0	1	0
1	1	0	0	1

- one-dimensional-by-one-dimensional (1Dx1D).

0	1	0	0	0
1	0	0	1	0
1	1	0	0	1

VHDL> Arrays in VHDL

- First the new TYPE must be defined,
- Then the new SIGNAL, VARIABLE, or CONSTANT can be declared using that data type.

TYPE type_name IS ARRAY (specification) OF data_type;

SIGNAL signal_name: type_name [:= initial_value];

```
TYPE row IS ARRAY (7 DOWNTO 0) OF STD_LOGIC;     -- 1D array
TYPE matrix IS ARRAY (0 TO 3) OF row;            -- 1Dx1D array
SIGNAL x: matrix;                                -- 1Dx1D signal

TYPE matrix IS ARRAY (0 TO 3) OF STD_LOGIC_VECTOR(7 DOWNTO 0);
```

VHDL> Arrays in VHDL

- 2D Array

```
TYPE matrix2D IS ARRAY (0 TO 3, 7 DOWNTO 0) OF STD_LOGIC;
-- 2D array
```

- Initial value:

```
... :="0001";                                          -- for 1D array
... :=('0','0','0','1')                                -- for 1D array
... :=(('0','1','1','1'), ('1','1','1','0'));         -- for 1Dx1D or
                                                       -- 2D array
```

VHDL> Arrays in VHDL

- Arrays assignments

```
x(0) <= y(1)(2);        -- notice two pairs of parenthesis
                        -- (y is 1Dx1D)
x(1) <= v(2)(3);        -- two pairs of parenthesis (v is 1Dx1D)
x(2) <= w(2,1);         -- a single pair of parenthesis (w is 2D)
```

- ROM (Read Only Memory)
- Example: Write a VHDL code to design a ROM which has a size of 64bit i.e. word size=8, number of addresses =8, assume a random data are stored in the memory.

```
1  ------------------------------------------------
2  LIBRARY ieee;
3  USE ieee.std_logic_1164.all;
4  ------------------------------------------------
5  ENTITY rom IS
6     GENERIC ( bits: INTEGER := 8;     -- # of bits per word
7               words: INTEGER := 8);  -- # of words in the memory
8     PORT ( addr: IN INTEGER RANGE 0 TO words-1;
9            data: OUT STD_LOGIC_VECTOR (bits-1 DOWNTO 0));
10 END rom;
```

- ROM (Read Only Memory)

```
11 ------------------------------------------------
12 ARCHITECTURE rom OF rom IS
13    TYPE vector_array IS ARRAY (0 TO words-1) OF
14       STD_LOGIC_VECTOR (bits-1 DOWNTO 0);
15    CONSTANT memory: vector_array := (  "00000000",
16                                        "00000010",
17                                        "00000100",
18                                        "00001000",
19                                        "00010000",
20                                        "00100000",
21                                        "01000000",
22                                        "10000000");
23 BEGIN
24    data <= memory(addr);
25 END rom;
26 ------------------------------------------------
```