

Chapter 3

1) - Signal: Has current and projected values, which can be changed using assignment operations as many times as needed.

- Constant: Has an assigned value that cannot change.

- Variable: Only has a current value. The value can be changed by assignment operations.

- File: A sequence of values of a specified type.

- Constants, variables, and files exist in traditional programming.

- Constants, variables, and signals are synthesizable.

- Files are not synthesizable and are primarily used in test benches.

2) - Scalar Type: Has a single indivisible value. May be numeric or enumerated.

- Composite: A collection of elements with values. May be an array or record.

 + Array: A collection of elements with the same type.

 + Record: A collection of elements that may vary in type.

- Access: Similar to a pointer. Provides access to objects of a given type.

- File: Provides access to a sequence of values of a given type.

- Protected: Provides exclusive access to variables accessible to multiple processes.

- Only Composite and Scalar are synthesizable.

- The other types may be used to test a design in a TestBench.

7) Entity bcd2xs3 is

```
port( D,C,B,A : in std_logic;
      XS3,XS2,XS1,XS0 : out std_logic);
```

```
end bcd2xs3;
```

9) An initial value can be assigned with ":="

It can simulate, but not synthesize. It is not IEEE Compliant

13) A subtype is a further restricted typing of a base type.

You can assign a subtype to a base type, but not a base type as a subtype.

- 15) U - un initialize
 X - force unknown
 O - force low
 I - force high
 Z - high impedance
 W - weak unknown
 L - weak low
 H - weak high
 — - don't care

- 18) Z L
 Z I
 H

20) type MV3 is ('0','1','Z');
 Subtype MV3r is resolve MV3 MV3;

- 21) True - precedence of the enum listed above
 True - char uses ASCII values, 1 is > 0
 True - enum above
 False - ASCII 'A' < 'a'
 True - bit 1 > 0

25) Subtype word is std_logic_vector(15 downto 0);
 Subtype dword is std_logic_vector(31 downto 0);
 Subtype qword is std_logic_vector(63 downto 0);

Signal address: dword;

Signal data: address;

26) entity memory is

```
Port( address: in std_logic_vector(7 downto 0);  
      data_in: in std_logic_vector(3 downto 0);  
      we_bar: in std_logic;  
      oe_bar: in std_logic;  
      data_out: out std_logic);
```

end memory

27) a = 1100

b = 0000

p = 10101000

q = 10001111

28) f: unsigned, 8 elements, 10010001

g: signed, 8 elements, 11100001

h: std_logic_vector, 5 elements, 10011

j: integer 13

k: integer -3

30) @1000

(A) 15

(C) 109

(B) 51

(E) 175

31) @ 3 bits

(B) 5 bits

(C) 1 bit

(D) 3 bits

(E) 6 bits