Overview

- Brief overview of VHDL
  - Entities and Architectures
  - Data Types and Issues
  - Packages and Libraries
  - Special Features
- Comparison VHDL vs. Verilog

Entities and Architectures

- Entity:
  - The entity is the primary hardware abstraction in VHDL and represents a part of a hardware design that has well-defined inputs and outputs and a well-defined function.
  - Analogous to a symbol in a block diagram.
Entities and Architectures (Continued)

- Architecture Body:
  - An architecture specifies the relationships between the inputs and outputs of a design entity. An architecture may be a mixture of structural, concurrent and sequential VHDL.
  - A given entity may have multiple different architectures.
  - Architecture to be used specified by a configuration.

library ieee;                             -- Instantiations of a library and a package are tied
use ieee.std_logic_1164.all;     -- to the following entity only.
entity decoder_3_to_8 is      -- Declaration of entity named decoder_3_to_8
  port (A : in std_logic_vector(2 downto 0); -- 3-bit input A of type std_logic
        D : out std_logic_vector(7 downto 0));  -- 8-bit output D of type std_logic
end decoder_3_to_8;       -- End of entity declaration

architecture functional of decoder_3_to_8 is -- Architecture named functional
begin
  D <= B"00000001" when A = B"000"              -- for entity decoder_3_to_8
      when B"001" else B"00001000" when A = B"010"
      else B"00001000" when A = B"011"          -- Concurrent when else statement
      else B"00000010" when A = B"000"         -- for evaluation of output D.
      else B"00000001" when A = B"001"
end decoder_3_to_8;
Entities and Architectures
- An Example (Continued)

```vhdl
else B"00010000" when A = B"100"
else B"00100000" when A = B"101"
else B"01000000" when A = B"110"
else B"10000000" when A = B"111"
else B"XXXXXXXX" when others;
end functional;
```

```
VHDL is case-insensitive: OR, Or and or
are all equivalent.

A signal can be viewed as a wire or cluster
of wires.

Signals are concurrent and sequential
objects.

A variable is not viewed as wire.
A variable is only a sequential object.
```

Signals, Variables and Processes

- VHDL is case-insensitive: OR, Or and or
  are all equivalent.
- A signal can be viewed as a wire or cluster
  of wires.
- Signals are concurrent and sequential
  objects.
- A variable is not viewed as wire.
- A variable is only a sequential object.
Miscellaneous Concepts (Continued)

- Signal assignment:
  - F <= A and B;

- Variable assignment:
  - F = A and B;

- Variables can be used only in a process (sequential VHDL).

- Variables are immediately evaluated with new value available to the next statement.

- Process has sensitivity list.

Signals, Variables and Processes - A Process Example

pick: process (S, D) is  -- Process statement with sensitivity list

begin
  case S is  -- Case statement conditioned by S
    when B"00" => Y <= D(0);  -- Use of subarrays D(i) of
    when B"01" => Y <= D(1);  -- array D.
    when B"10" => Y <= D(2);
    when B"11" => Y <= D(3);
    when others => Y <= 'X';
  end case;
end process pick;  

2/1/2000