VHDL Statement Types

Concurrent Statements:

assignment (concurrent - behavioral)**:**

```
example:
      w <= a and b or c;
```

---

when-else Conditional Signal Assignment:
(concurrent - behavioral):

```
example1:
      x <= '1' when b = c else '0';

example2:
      x <= "01" when b = "00" else
           "10" when b = "01" else
           "00" when others;
```

---

with-select-when Selected Signal Assignment
(concurrent - behavioral):

```
example:
      ENTITY mux IS
        PORT (
               i0 : in STD_LOGIC;
               i1 : in STD_LOGIC;
               sel: in STD_LOGIC_VECTOR(1 downto 0);
               z  : out STD_LOGIC
               );
      END mux;

      ARCHITECTURE behavior OF mux IS
      BEGIN

      with sel select
      z <=   i0 when '0',
             i1 when '1',
             '0' when others;


      END behavior;
```

VHDL Statement Types

port map statement (concurrent - structural VHDL)
example 1:

```
    U2:  add1
    port map(
       cin => ssc(0),
       a => a(1),
       b => b(1),
       g => ssg(1),
       p => ssp(1),
       s => s(1)
       );
```

example 2:

```
    U3: lca_gen
    port map(
       cin => cin,
       g0 => sg(0),
       g1 => sg(1),
       p0 => sp(0),
       p1 => sp(1),
       cout0 => ssc(0),
       cout1 => ssc(1),
       bg => blk_g,
       bp => blk_p
       );
```

---

component declaration (concurrent - structural VHDL)
example:

```
    component add1
    port (
       cin: in std_logic;
       a: in std_logic;
       b: in std_logic;
       g: out std_logic;
       p: out std_logic;
       s: out std_logic
       );
    end component;
```

VHDL Statement Types

Sequential Statements:

Process Statement:

```
  architecture archcompare of compare is
  begin
  label:  process (a, b) <--- sensitivity list
      begin              <-- beginning of process block
        .. process body
        .. statements within are sequential
        .. statements here will not be executed
           unless there are changes in a or b.
      end process label;
  end archcompare;
```

---

if-then-else Statement

(sequential - behavioral):

example 1:

```
  comp: process (a, b)
  begin
        if a = b then
            aeqb <= '1';
        else
            aeqb <= '0';
        end if;
  end process comp;
```

example 2:

```
  mux4_1:   process (a, b, c, d)
      begin
          if s = "00" then
             x <= a;
          elsif s = "01" then <--watch out about elsif (spelling)
             x <= b;
          elsif s = "10" then
             x <= c;
          else
             x <= d;
          end if;
      end process mux4_1;
```

example 3:

```
  library ieee;
  use ieee.std_logic_1164.all;

  entity registered is port(
      d, clk: in std_logic;
      q:  out std_logic);
  end registered;

  architecture archregistered of registered is
  begin
  reg:process (clk)
      begin
          if (clk'event and clk= '1') then
              q <= d;
          end if;
      end process reg;
  end archregistered;
```

---

case-when Statement
(sequential - behavioral):
example:

```
  process (address)
       begin
            case address is
                when "001"  => decode <= X"11";
                when "111"  => decode <= X"42";
                when "010"  => decode <= X"44";
                when "101"  => decode <= X"88";
                when others => decode <= X"00";
            end case;
       end process;
```

VHDL Statement Types

for-loop Statement (sequential - behavioral):
example:

```
 loopf:
 for i in 3 downto 0 loop
    if reset(i) = '1' then
       data_out(i) := '0';
    end if;
 end loop loopf;
```

---

while-loop Statement (sequential):
example:

```
  count := 16;
  loopw:
    while (count > 0) loop
       count := count - 1;
       result <= result + data_in;
    end loop loopw;
```
