

Process Statement

A **process statement** is a concurrent command that consists of a label, sensitivity list, declaration area, begin-end (body) area, and sequential statements. An example of a process statement description in VHDL is:

```
process (<Sensitivity List>
  -- Variable declaration area
begin
  -- VHDL statement block here
end process;
```

The difference between a **concurrent signal assignment statement** and a **process statement** lies in the sequential statements. The syntax or statements inside the begin-end (body) section are executed sequentially, line by line, just like in general programming. The **process label** itself is simply a self-descriptive naming form to help us recognize which process is being executed in that section, so the naming can be changed or even omitted.

In a **concurrent statement**, every time a change occurs in the input, the output is re-evaluated. In a **behavioral style** model using a process statement, whenever a change occurs in a signal listed in the sensitivity list of the process, all the sequential statements within the process body are re-evaluated.

Since a **process statement** is itself a concurrent statement, if there are two processes in the architecture body, the execution of both processes will be carried out concurrently.

Revision #1

Created 2025-09-09 12:01:57 UTC by BH

Updated 2025-09-09 12:03:47 UTC by BH