# Management and analysis of physics datasets, Part. 1

Third Laboratory

Antonio Bergnoli bergnoli@pd.infn.it

17/11/2021

# Process Statements

# Some clarifications

1. the ability to build a sequential circuit starting from a source code with some special processes (i.e. with **rising_edge(clk)**) is called **INFERENCE**
2. a not empty sensitivity list could not co-exist with **wait** statements
3. unless a **wait** statement ( with no paramenters) is placed in the body a process runs **FOREVER**

# Processes for combinatorial logic

Remember to enumerate all the signals ( in right side of the assignments)

```
process (x,y,z,a,b) is
begin  -- process
  -- statements
  out1 <= x and y;
  out2 <= (z or a) and b;
 end process;
```

# Processes for sequential logic ( synchronous reset)

1. the reset conditional is **inside** the **rising_edge(clk)** conditional
2. only the **clk** signal is enumerated in the sensitivity list

```
process (clk) is
begin  -- process
  if rising_edge(clk) then
    if rst = '1' then
      -- statements
      else
    -- statements
    end if;
  end if;
end process;
```

# Processes for sequential logic ( asynchronous reset)

1. the reset conditional is **outside** the **rising_edge(clk)** conditional
2. reset signal is enumerated in the sensitivity list

```
process (clk, rst) is
begin  -- process
  if rst = '0' then

  -- statements
  elsif rising_edge(clk) then
  -- statements
  end if;
end if;
end process;
```

# Processes that execute once

a **wait** statement is placed as last line of code in the body

```
process is
begin  -- process
-- statements
  wait;
end process;
```

# Processes that run forever

no **wait** statement ( with no arguments) present in the body

```
process is
begin  -- process
  -- statements
  wait for 10 us;
end process;
```