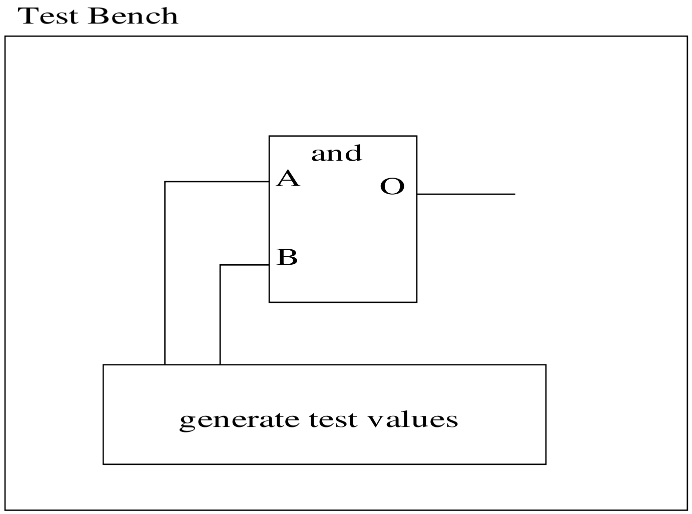
Introduction to VHDL
-riviera and test benches

COMP311 2007

Tony McGregor

Test Bench

- To test a VHDL design we need to apply signals to its inputs
- May want to test a number of different inputs
- Put the design under test into a “test bench”



Test Benches

- The test bench itself has no inputs
- The test bench is a higher level entity than the system under test
  - In VHDL an entity may contain other enties
  - Will look at hierarchical design more later
- Set signal values for inputs then wait for a time
- Set new values ... repeat

AND Gate

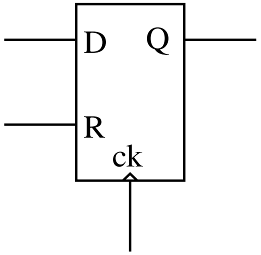
```
library ieee;                -- Use the IEEE library
use ieee.std_logic_1164.all; -- std_logic_1164 package

entity and_gate is
  port (
    and_x : in std_logic;    -- These are the two input ports
    and_y : in std_logic;
    and_output : out std_logic  -- This is the output
    );
end and_gate;

architecture structural of and_gate is
begin
  -- Simple concurrent VHDL assignment
  and_output <= and_x and and_y;
end structural;
```

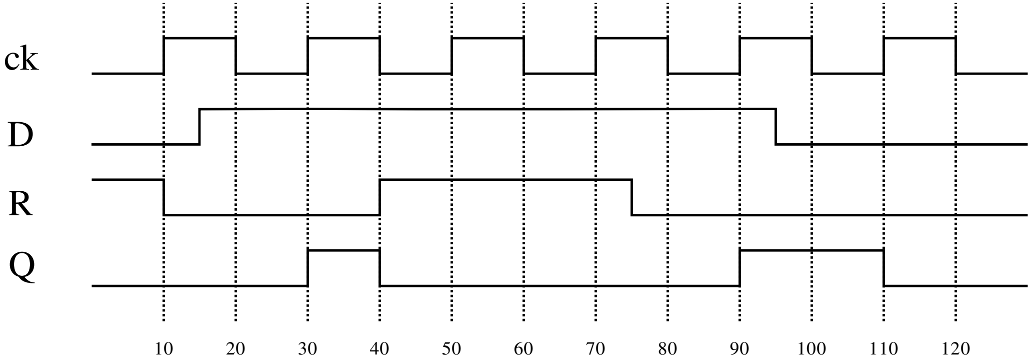
And gate Test Bench

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

entity and_gate_tb is
end and_gate_tb;

architecture tb of and_gate_tb is
  component and_gate
    port (
      and_x      : in  std_logic;
      and_y      : in  std_logic;
      and_output : out std_logic);
  end component;

  signal and_x_i      : std_logic;
  signal and_y_i      : std_logic;
  signal and_output_i : std_logic;

begin
 DUT: and_gate
    port map (
        and_x      => x_i,
        and_y      => y_i,
        and_output => output_i);

  test : process
begin
  x_i <= '0'; y_i <= '0'; wait for 10ns;
  x_i <= '0'; y_i <= '1'; wait for 10ns;
  x_i <= '1'; y_i <= '0'; wait for 10ns;
  x_1 <= '1'; y_i <= '1'; wait for 10ns;
  wait;
  end process;
end tb;
```

D- Flipflop with reset

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

entity D_flipflop is
  port (
    reset, clk : in std_logic;
    D : in std_logic;
    Q : out std_logic
    );
end D_flipflop;

architecture rtl of D_flipflop is
begin
  process (reset, clk)
  begin
    if reset = ‘1’ then
      Q <= ‘0’;
    elsif rising_edge(clk) then
      Q <= D;
    end if;
  end process;
end rtl;
```





Riviera Crib Sheet

- Run `/usr/local/riviera-2007.06/rungui`
- Select `file->change directory`
- `library->create->library->ok`
- `compilation->compile_files`

  select each of the vhdl files then compile (check for errors at the bottom of the window)
- `simulation->initialise` (select test bench)
- `view->structure browser`
- click on waveform (small icon near top of window)
- drag the signals from the structure browser into the waveform window
- `simulation->run`

  (You should see the waveforms of the simulation in the waveform window, if they're very small, use `waveform->zoom_to_fit`)