control_tb.vhd

```
-------------------------------------------------------------------------------
--
-- Title       : control_tb
-- Design      : testbench
-- Author      : Dongyun Lee
-- Company     : Stony Brook University
--
-------------------------------------------------------------------------------
--
-- File        : \\Mac\Home\Documents\Aldec_Codes\test\test\src\test.vhd
-- Generated   : 2024-03-04  15:00:00
-- From        : interface description file
-- By          : Itf2Vhdl ver. 1.22
--
-------------------------------------------------------------------------------
--
-- Description : testbench for control
-------------------------------------------------------------------------------

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use ieee.numeric_std.all;
use work.all;

entity control_tb is
end control_tb;

architecture tb_architecture of control_tb is

    -- Stimulus signals - signals mapped to the inputs of tested entity
    signal gn, an: std_logic;
    -- Observed signals - signals mapped to the outputs of tested entity
    signal c0, c1 : std_logic;


    type test_vector is record
        gn : std_logic;
        an : std_logic;
        c0 : std_logic;
        c1 : std_logic;
    end record;

    type test_vector_array is array (natural range <>) of test_vector;


    constant test_vectors : test_vector_array := (
    --    gn,   an,  c0,  c1
        ( '0', '0', '1', '0'),
        ( '0', '1', '0', '1'),
        ( '1', '0', '0', '0'),
        ( '1', '1', '0', '0')
        );

    -- time between application of input stimulus vectors
    --constant period : time := 20ns;

begin
```

```vhdl
    -- Unit Under Test port map
    UUT : entity control
    port map (
            gn => gn, an => an, c0 => c0, c1 => c1
        );

    verify: process
    begin
    for i in test_vectors'range loop
        gn <= test_vectors(i).gn;
        an <= test_vectors(i).an;
        c0 <= test_vectors(i).c0;
        c1 <= test_vectors(i).c1;
        wait for 20 ns;
        assert (gn = test_vectors(i).gn) and (an = test_vectors(i).an) and (c0 = test_vectors(i).c0) and (c1 = test_vectors(i).c1)
        report "test vector " & integer'image(i)& " failed" & " for input gn = " & std_logic'image(gn) & " and an = " & std_logic'image(an)
        severity error;
    end loop;

        --std.env.finish;   --stop simulation
    end process;

end tb_architecture;

```