

```

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY register_file_tb IS
END register_file_tb;

ARCHITECTURE behavior OF register_file_tb IS

    -- Component Declaration for the Unit Under Test (UUT)

    COMPONENT register_file
    PORT(
        load : IN  std_logic;
        Clk : IN  std_logic;
        dest_sel : IN  std_logic_vector(2 downto 0);
        a_sel : IN  std_logic_vector(2 downto 0);
        b_sel : IN  std_logic_vector(2 downto 0);
        data : IN  std_logic_vector(15 downto 0);
        a_out : OUT  std_logic_vector(15 downto 0);
        b_out : OUT  std_logic_vector(15 downto 0)
    );
    END COMPONENT;

    --Inputs
    signal load : std_logic := '0';
    signal Clk : std_logic := '0';
    signal dest_sel : std_logic_vector(2 downto 0) := (others => '0');
    signal a_sel : std_logic_vector(2 downto 0) := (others => '0');
    signal b_sel : std_logic_vector(2 downto 0) := (others => '0');
    signal data : std_logic_vector(15 downto 0) := (others => '0');

    --Outputs
    signal a_out : std_logic_vector(15 downto 0);
    signal b_out : std_logic_vector(15 downto 0);

    -- Clock period definitions
    constant Clk_period : time := 10 ns;

BEGIN

    -- Instantiate the Unit Under Test (UUT)
    uut: register_file PORT MAP (
        load => load,
        Clk => Clk,
        dest_sel => dest_sel,
        a_sel => a_sel,
        b_sel => b_sel,
        data => data,
        a_out => a_out,
        b_out => b_out
    );

    -- Clock process definitions
    Clk_process :process
    begin
        Clk <= '0';
        wait for Clk_period/2;
        Clk <= '1';
        wait for Clk_period/2;
    end process;

    -- Stimulus process
    stim_proc: process
    begin
        --test load into registers
        data <= "0000000000000001";
        load <= '1';
        dest_sel <="000";

        wait for 20 ns;
        dest_sel <="001";
        data <= "0000000000000011";

        wait for 20 ns;
        dest_sel <="010";
    end process;

```

```

        data <= "000000000000111";

wait for 20 ns;
dest_sel <="011";
data <= "0000000000001111";

wait for 20 ns;
dest_sel <="100";
data <= "0000000000011111";

wait for 20 ns;
dest_sel <="101";
data <= "0000000001111111";

wait for 20 ns;
dest_sel <="110";
data <= "0000000011111111";

wait for 20 ns;
dest_sel <="111";
data <= "0000000111111111";

--test output selection
wait for 20 ns;
a_sel <= "000";
b_sel <= "000";

wait for 20 ns;
a_sel <= "001";
b_sel <= "001";

wait for 20 ns;
a_sel <= "010";
b_sel <= "010";

wait for 20 ns;
a_sel <= "011";
b_sel <= "011";

wait for 20 ns;
a_sel <= "100";
b_sel <= "100";

wait for 20 ns;
a_sel <= "101";
b_sel <= "101";

wait for 20 ns;
a_sel <= "110";
b_sel <= "110";

wait for 20 ns;
a_sel <= "111";
b_sel <= "111";

wait;
end process;

END;

```