```vhdl
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY datapath_tb IS
END datapath_tb;

ARCHITECTURE behavior OF datapath_tb IS

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

    COMPONENT datapath
    PORT(
                        control : in STD_LOGIC_VECTOR(16 downto 0);
                        TD : in STD_LOGIC;
                        TA : in STD_LOGIC;
                        TB : in STD_LOGIC;
         clk : IN  std_logic;
         const_in : IN  std_logic_vector(2 downto 0);
         data_in : IN  std_logic_vector(15 downto 0);
                        PC_in : in STD_LOGIC_VECTOR (15 downto 0);
                        MM : in STD_LOGIC;
         V : OUT  std_logic;
         C : OUT  std_logic;
         N : OUT  std_logic;
         Z : OUT  std_logic;
         address_out : OUT  std_logic_vector(15 downto 0);
         data_out : OUT  std_logic_vector(15 downto 0)
        );
    END COMPONENT;


   --Inputs
   signal control : std_logic_vector(16 downto 0) := (others => '0');
        signal TD : STD_LOGIC := '0';
        signal TA : STD_LOGIC := '0';
        signal TB : STD_LOGIC := '0';
        signal MM : STD_LOGIC := '0';
   signal clk : std_logic := '0';
   signal const_in : std_logic_vector(2 downto 0) := (others => '0');
   signal data_in : std_logic_vector(15 downto 0) := (others => '0');
   signal PC_in : std_logic_vector(15 downto 0) := (others => '0');

        --Outputs
   signal V : std_logic;
   signal C : std_logic;
   signal N : std_logic;
   signal Z : std_logic;
   signal address_out : std_logic_vector(15 downto 0);
   signal data_out : std_logic_vector(15 downto 0);

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

BEGIN

        -- Instantiate the Unit Under Test (UUT)
   uut: datapath PORT MAP (
                         control => control,
                         TD => TD,
                         TA => TA,
                         TB => TB,
                         MM => MM,
          clk => clk,
          const_in => const_in,
          data_in => data_in,
          PC_in => PC_in,
          V => V,
          C => C,
          N => N,
          Z => Z,
          address_out => address_out,
          data_out => data_out
        );

   -- Clock process definitions
   clk_process :process
   begin
```

```vhdl
             clk <= '0';
             wait for clk_period/2;
             clk <= '1';
             wait for clk_period/2;
end process;


-- Stimulus process
stim_proc: process
begin
   -- hold reset state for 100 ns.
             control <= "00000000100000001";

             const_in <= "000";
             wait for 100 ns;

             control <= "00000000100000011";

             data_in <= "0000111100000000";

             wait for 20 ns;
             control <= "00100000100000011";

             data_in <= "1111000011111111";

             wait for 20 ns;
             control <= "00100000100000010";

             wait for 20 ns;
             control <= "01000000100000000";

             wait for 20 ns;
             control <= "01000000100000001";

             wait for 20 ns;
             control <= "01000000100000101";

             wait for 20 ns;
             control <= "01000000100001001";

             wait for 20 ns;
             control <= "01000000100001101";

             wait for 20 ns;
             control <= "01000000100010001";

             wait for 20 ns;
             control <= "01000000100010101";

             wait for 20 ns;
             control <= "01000000100011001";

             wait for 20 ns;
             control <= "01000000100011101";

             wait for 20 ns;
             control <= "01000000100100001";

             wait for 20 ns;
             control <= "01000000100101001";

             wait for 20 ns;
             control <= "01000000100110001";

             wait for 20 ns;
             control <= "01000000100111001";

             wait for 20 ns;
             control <= "01000000101000001";

             wait for 20 ns;
             control <= "01000000101010001";

             wait for 20 ns;
             control <= "01000000101100001";

             wait for 50 ns;
```

```vhdl
                control <= "0000000010000011";
                TD <= '1';
                data_in <= "0101010101010101";

      wait;
   end process;

END;
```