```
    1: library STD;
    2: library IEEE;
    3: use IEEE.std_logic_1164.all;
    4:
    5: entity Cache_Block_Test is
    6:
    7: end Cache_Block_Test;
    8:
    9: architecture test of Cache_Block_Test is
   10:
   11:     component Cache_Block
   12:     port(
   13:         Data_In     :   in  std_logic_vector(7 downto 0);
   14:         Tag_In      :   in  std_logic_vector(2 downto 0);
   15:         Set_Valid   :   in  std_logic;
   16:         Rd_Wr       :   in  std_logic;
   17:         Cache_Write :   in  std_logic;
   18:         Col_En      :   in  std_logic_vector(3 downto 0);
   19:         Row_En      :   in  std_logic_vector(7 downto 0);
   20:         Tag_Wr_En   :   in  std_logic;
   21:         Gnd         :   in  std_logic;
   22:         reset       :   in  std_logic;
   23:         enable      :   in  std_logic;
   24:         Data_Out    :   out std_logic_vector(7 downto 0);
   25:         Tag_Out     :   out std_logic_vector(2 downto 0);
   26:         Valid_Out   :   out std_logic
   27:     );
   28:     end component;
   29:
   30:     signal Data_In, Row_En, Data_Out    :   std_logic_vector(7 downto 0);
   31:     signal Tag_In, Tag_Out              :   std_logic_vector(2 downto 0);
   32:     signal Col_En                       :   std_logic_vector(3 downto 0);
   33:     signal Set_Valid, Rd_Wr, Cache_Write, Tag_Wr_En, reset, enable, Valid_Out
  :   std_logic;
   34:
   35:     for asdf    :   Cache_Block use entity work.Cache_Block(structural);
   36:
   37: begin
   38:
   39:     asdf    :   Cache_Block port map(Data_In, Tag_In, Set_Valid, Rd_Wr, Cache_
Write, Col_En, Row_En, Tag_Wr_En, '0', reset, enable, Data_Out, Tag_Out, Valid_Out);
   40:
   41:     p  :   process
   42:     begin
   43:         Data_In <= "00000000";
   44:         Tag_In  <= "000";
   45:         Set_Valid   <= '0';
   46:         Rd_Wr   <= '0';
   47:         Cache_Write <= '0';
   48:         Col_En  <= "0000";
   49:         Row_En  <= "00000000";
   50:         Tag_Wr_En   <=  '0';
   51:         reset <= '1';
   52:         wait for 5 ns;
   53:         reset <= '0';
   54:         wait for 5 ns;
   55:         Row_En  <= "00100000";
   56:         Col_En  <= "1000";
   57:         Data_In <= "11111111";
   58:         rd_wr   <= '1';
   59:         Tag_In <= "010";
   60:         wait for 5 ns;
   61:         Cache_Write <= '1';
   62:         Set_Valid <= '1';
   63:         Tag_Wr_En <= '1';
   64:         wait for 5 ns;
   65:         Cache_Write <= '0';
   66:         Set_Valid <= '0';
   67:         Tag_Wr_En <= '0';
   68:         rd_wr <= '0';
   69:         wait for 5 ns;
   70:         Row_En <= "10000000";
   71:         Col_En <= "0100";
   72:         Data_In <= "11110000";
   73:         Tag_In <= "001";
   74:         wait for 5 ns;
   75:         Cache_Write <= '1';
   76:         Set_Valid <= '1';
   77:         Tag_Wr_En <= '1';
   78:         wait for 5 ns;
   79:         Cache_Write <= '0';
   80:         Set_Valid <= '0';
   81:         Tag_Wr_En <= '0';
   82:         wait for 5 ns;
   83:         rd_wr   <= '1';
   84:         Row_En  <= "00100000";
   85:         Col_En  <= "1000";
   86:         wait for 5 ns;
   87:         Row_En <= "10000000";
   88:         Col_En <= "0100";
   89:         wait for 5 ns;
   90:         wait;
   91:     end process;
   92: end test;
```