```
1   LIBRARY ieee  ;
2   USE ieee.std_logic_1164.all  ;
3
4
5   ENTITY gen_add_sub_tb  IS
6   END ;
7
8   ARCHITECTURE gen_add_sub_tb_arch OF gen_add_sub_tb IS
9     TYPE EN_ARRAY  IS ARRAY (1 TO 4) OF STD_LOGIC;
10    TYPE DATAARRAY IS ARRAY (1 TO 4) OF STD_LOGIC_VECTOR(31 DOWNTO 0);
11
12    CONSTANT DELAY    :  time := 10 ns;
13    CONSTANT OP_IN    :  EN_ARRAY  := ('1','1','1','1');
14
15    CONSTANT D1_IN    :  DATAARRAY := ("00000000000000000000000000000011",
      "00000000000000000000000000000000",
16                                     "01111111111111111111111111111111",
                                       "10000000000000000000000000000000");
17
18    CONSTANT D2_IN    :  DATAARRAY := ("11111111111111111111111111111101",
      "00000000000000000000000000000000",
19                                     "01000000000000000000000000000000",
                                       "01000000000000000000000000000000");
20
21
22
23
24    SIGNAL DATA1_IN   :  std_logic_vector (31 downto 0) :=
      "00000000000000000000000000000000"  ;
25    SIGNAL V          :  STD_LOGIC := '0'  ;
26    SIGNAL Z          :  STD_LOGIC := '0'  ;
27    SIGNAL DATA2_IN   :  std_logic_vector (31 downto 0) :=
      "00000000000000000000000000000000"  ;
28    SIGNAL OP         :  STD_LOGIC := '1'  ;
29    SIGNAL C          :  STD_LOGIC := '0'  ;
30    SIGNAL RESULT     :  std_logic_vector (31 downto 0) :=
      "UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU"  ;
31
32    COMPONENT GEN_ADD_SUB
33      PORT (
34        DATA1_IN  : in std_logic_vector (31 downto 0) ;
35        V  : out STD_LOGIC ;
36        Z  : out STD_LOGIC ;
37        DATA2_IN  : in std_logic_vector (31 downto 0) ;
38        OP  : in STD_LOGIC ;
39        C  : out STD_LOGIC ;
40        RESULT  : out std_logic_vector (31 downto 0) );
41    END COMPONENT ;
42
43  BEGIN
44
45  TEST : PROCESS
46      variable i : integer := 1;
47      BEGIN
48        FOR i IN 1 TO 4 LOOP
49          DATA1_IN   <= D1_IN(i);
50          DATA2_IN  <= D2_IN(i);
51          OP      <= OP_IN(i);
52          wait for (DELAY/2);
53
54        end loop;
55        wait;
56      end process TEST;
57
58
59    DUT  : GEN_ADD_SUB
60      PORT MAP (
61        DATA1_IN   => DATA1_IN  ,
62        V   => V  ,
```

```
63          Z    => Z   ,
64          DATA2_IN    => DATA2_IN   ,
65          OP    => OP   ,
66          C    => C   ,
67          RESULT    => RESULT    ) ;
68    END ;
69
70
```