

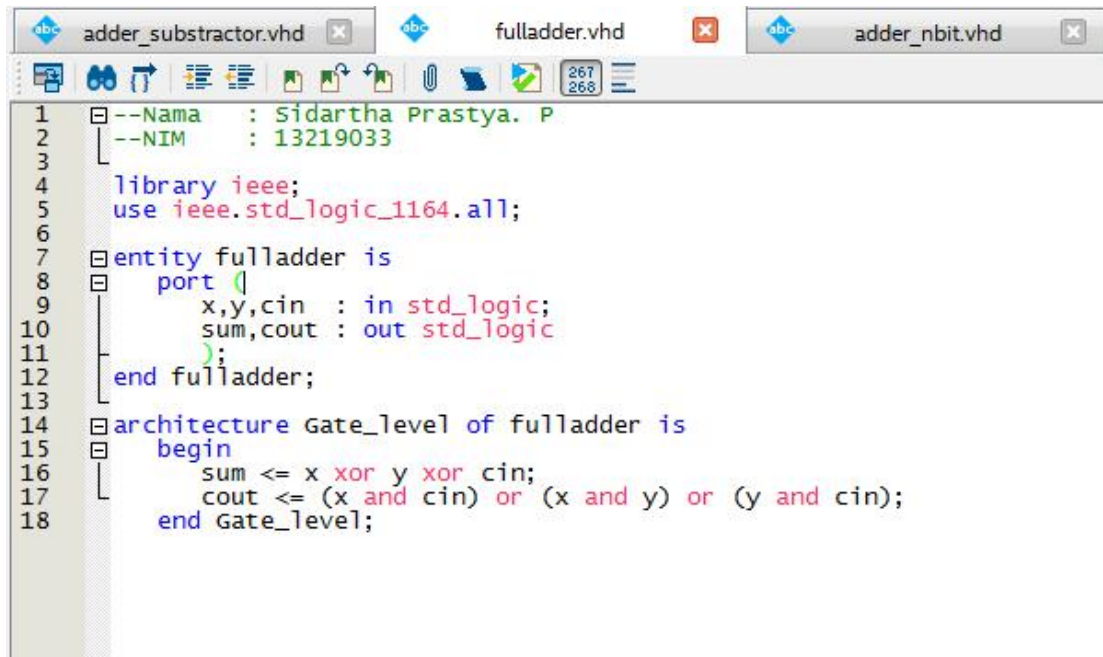
Nama : Sidartha Prastya. P

NIM :13219033

Praktikum 2

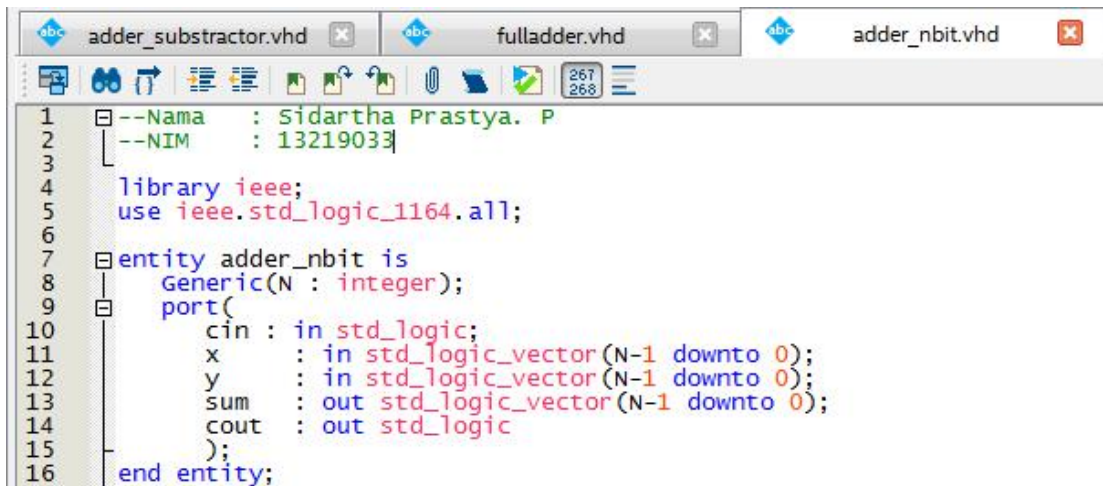
1. Pada project pertama ini, diperintahkan untuk membuat suatu adder-subtractor (substractor yang terbuat dari fulladder) yang terdiri dari 3 file, yaitu fulladder, adder-n-bit, dan adder-subtractor. (Bilangan 16 bit)

File 1.1: fulladder.vhd



```
1  --Nama : Sidartha Prastya. P
2  --NIM : 13219033
3
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  entity fulladder is
8  port (
9      x,y,cin : in std_logic;
10     sum,cout : out std_logic
11 );
12 end fulladder;
13
14 architecture Gate_level of fulladder is
15 begin
16     sum <= x xor y xor cin;
17     cout <= (x and cin) or (x and y) or (y and cin);
18 end Gate_level;
```

File 1.2: adder_nbit.vhd



```
1  --Nama : Sidartha Prastya. P
2  --NIM : 13219033
3
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  entity adder_nbit is
8      Generic(N : integer);
9  port(
10     cin : in std_logic;
11     x : in std_logic_vector(N-1 downto 0);
12     y : in std_logic_vector(N-1 downto 0);
13     sum : out std_logic_vector(N-1 downto 0);
14     cout : out std_logic
15 );
16 end entity;
```

```

17
18 architecture structural of adder_nbit is
19     component fulladder is
20     port(
21         x,y,cin : in std_logic;
22         sum,cout : out std_logic
23     );
24     end component;
25
26     signal c : std_logic_vector(N-1 downto 0);
27
28     begin
29         Adder_0 : fulladder
30         port map(
31             cin => Cin,
32             x => x(0),
33             y => y(0),
34             sum => sum(0),
35             cout => C(0)
36         );
37
38         Adders : for i in 1 to N-1 generate
39             Adder : fulladder
40             port map(
41                 cin => c(i-1),
42                 x => x(i),
43                 y => y(i),
44                 sum => sum(i),
45                 cout => C(i));
46         end generate;
47
48         cout <= C(N-1);
49
50     end architecture structural;

```

File 1.3: adder_subtractor.vhd

```

1  --Nama   : Sidartha Prastya. P
2  --NIM    : 13219033
3
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  -- Inisiasi adder subtractor sebagai 16bit
8  entity adder_subtractor is
9  port(
10     OP      : in std_logic;
11     x_sub   : in std_logic_vector(15 downto 0);
12     y_sub   : in std_logic_vector(15 downto 0);
13     result_sub : out std_logic_vector(15 downto 0);
14     cout_sub : out std_logic
15 );
16 end entity;
17
18 architecture structural of adder_subtractor is
19     component adder_nbit is
20         Generic (N: integer);
21         port(
22             cin   : in std_logic;
23             x     : in std_logic_vector(N-1 downto 0);
24             y     : in std_logic_vector(N-1 downto 0);
25             sum    : out std_logic_vector(N-1 downto 0);
26             cout   : out std_logic
27         );
28     end component;
29
30     signal truey: std_logic_vector(15 downto 0);
31

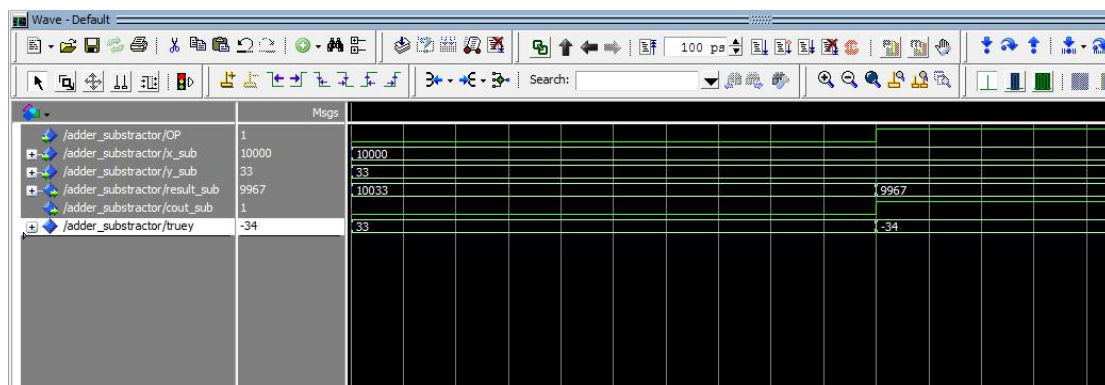
```

```

31
32 begin
33
34   subtractor : for i in 0 to 15 generate
35     truey(i) <= y_sub(i) xor OP;
36   end generate;
37
38   adder : adder_nbit
39     Generic map (N => 16)
40     port map(OP, x_sub, truey,result_sub,cout_sub);
41
42   end Structural;
43

```

Hasil simulasi dapat dilihat seperti gambar di bawah ini:



Penjelasan:

Ketika $OP = 1$, maka vhdl akan melakukan operasi pengurangan. Untuk operasinya sendiri menggunakan input x dan input truey (input y yang diubah menjadi 2's complement bila $OP = 1$ / pengurangan). Input y diubah menjadi 2's complement dengan mengubah 0 menjadi 1 dan sebaliknya, kemudian carry in diinisiasi sama dengan OP.

2. Pada project kedua ini, diperintahkan untuk membuat suatu subtractor-adder (subtractor yang terbuat dari fullsubtractor) yang terdiri dari 3 file, yaitu fullsubtractor, subtractor-n-bit, dan subtractor-adder. (Bilangan 16 bit)

File 2.1: fullsubs.vhd

```

1  --Nama   : Sidartha Prastya. P
2  --NIM    : 13219033
3
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  entity fullsubs is
8  port (
9      x,y,bin : in std_logic;
10     subs,bout : out std_logic
11 );
12 end fullsubs;
13
14 architecture Gate_level of fullsubs is
15 begin
16     subs <= x xor y xor bin;
17     bout <= (not(x) and bin) or (not(x) and y) or (y and bin);
18 end Gate_level;

```

File 2.2: substractor_nbit.vhd

```

1  --Nama   : Sidartha Prastya. P
2  --NIM    : 13219033
3
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  entity substractor_nbit is
8  Generic(N : integer);
9  port(
10     bin : in std_logic;
11     x    : in std_logic_vector(N-1 downto 0);
12     y    : in std_logic_vector(N-1 downto 0);
13     subs : out std_logic_vector(N-1 downto 0);
14     bout : out std_logic
15 );
16 end entity;
17
18 architecture structural of substractor_nbit is
19     component fullsubs is
20     port(
21         x,y,bin : in std_logic;
22         subs,bout : out std_logic
23     );
24     end component;
25
26     signal c : std_logic_vector(N-1 downto 0);
27
28     begin
29         Adder_0 : fullsubs
30         port map(

```

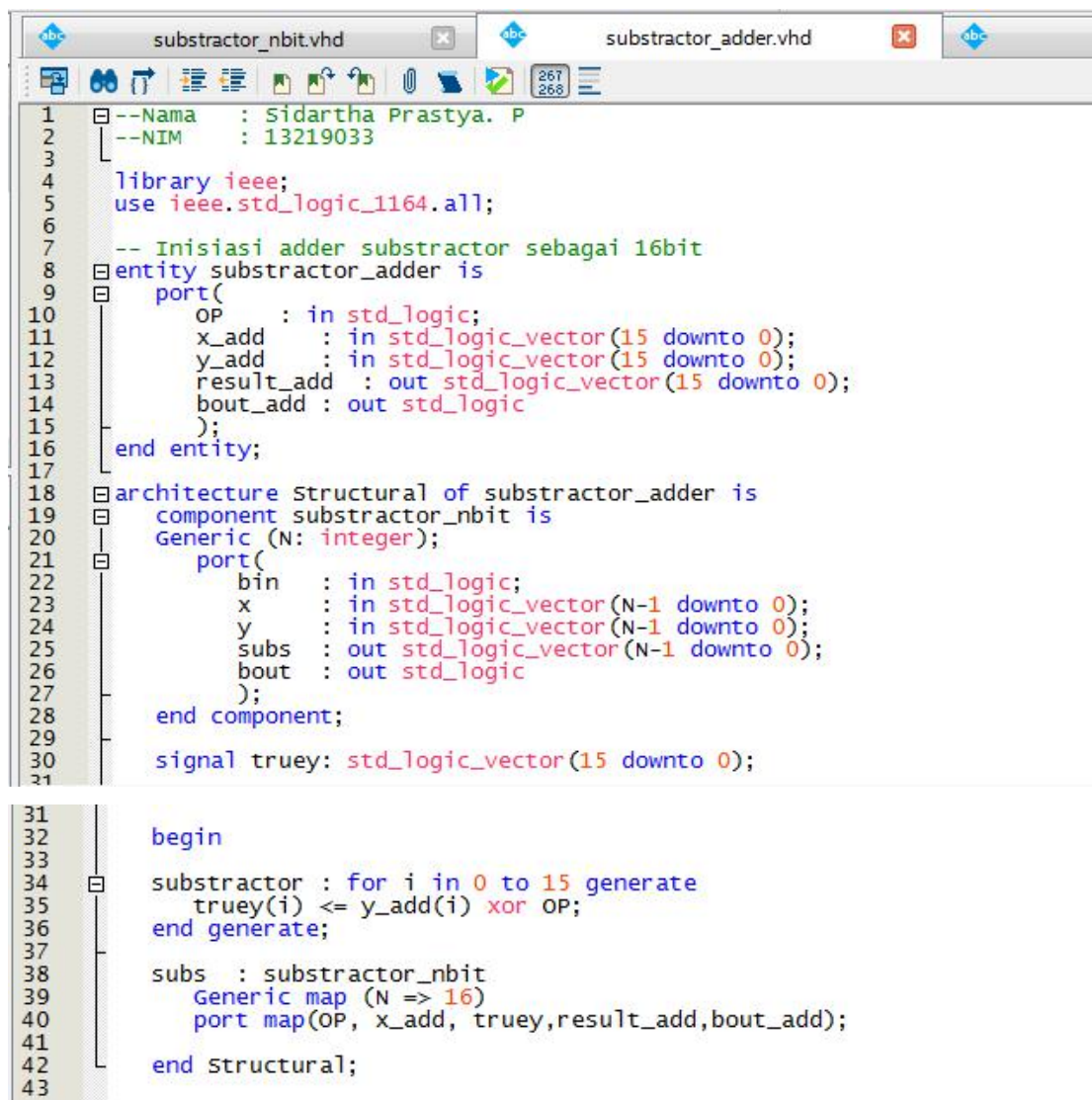


```

31     bin => bin,
32     x => x(0),
33     y => y(0),
34     subs => subs(0),
35     bout => c(0)
36 );
37
38 Adders : for i in 1 to N-1 generate
39     Adder : fullsubs
40     port map(
41         bin => c(i-1),
42         x => x(i),
43         y => y(i),
44         subs => subs(i),
45         bout => c(i));
46 end generate;
47
48 bout <= c(N-1);
49
50 end architecture Structural;

```

File 2.3: substractor_adder.vhd

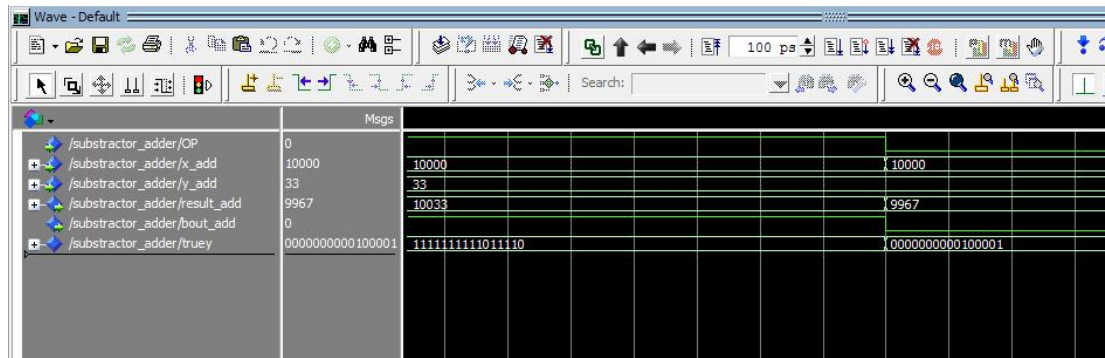


```

1  --Nama : Sidartha Prastya. P
2  --NIM : 13219033
3
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  -- Inisiasi adder substractor sebagai 16bit
8  entity substractor_adder is
9  port(
10     OP : in std_logic;
11     x_add : in std_logic_vector(15 downto 0);
12     y_add : in std_logic_vector(15 downto 0);
13     result_add : out std_logic_vector(15 downto 0);
14     bout_add : out std_logic
15 );
16 end entity;
17
18 architecture Structural of substractor_adder is
19     component substractor_nbit is
20         Generic (N: integer);
21         port(
22             bin : in std_logic;
23             x : in std_logic_vector(N-1 downto 0);
24             y : in std_logic_vector(N-1 downto 0);
25             subs : out std_logic_vector(N-1 downto 0);
26             bout : out std_logic
27 );
28     end component;
29
30     signal truey: std_logic_vector(15 downto 0);
31
32 begin
33
34     substractor : for i in 0 to 15 generate
35         truey(i) <= y_add(i) xor OP;
36     end generate;
37
38     subs : substractor_nbit
39         Generic map (N => 16)
40         port map(OP, x_add, truey, result_add, bout_add);
41
42 end structural;
43

```

Hasil Simulasi dapat dilihat seperti di bawah ini:



Penjelasan:

Ketika $OP = 1$, maka vhdl akan melakukan operasi penambahan. Untuk operasinya sendiri menggunakan input x dan input truey (input y yang diubah menjadi 2's complement bila $OP = 1$ / penambahan). Input y diubah menjadi 2's complement dengan mengubah 0 menjadi 1 dan sebaliknya, kemudian carry in diinisiasi sama dengan OP. VHDL ini sama dengan nomor sebelumnya, hanya saja berbeda di logika fullsubtractor dengan mengubah x menjadi $\text{not}(x)$.