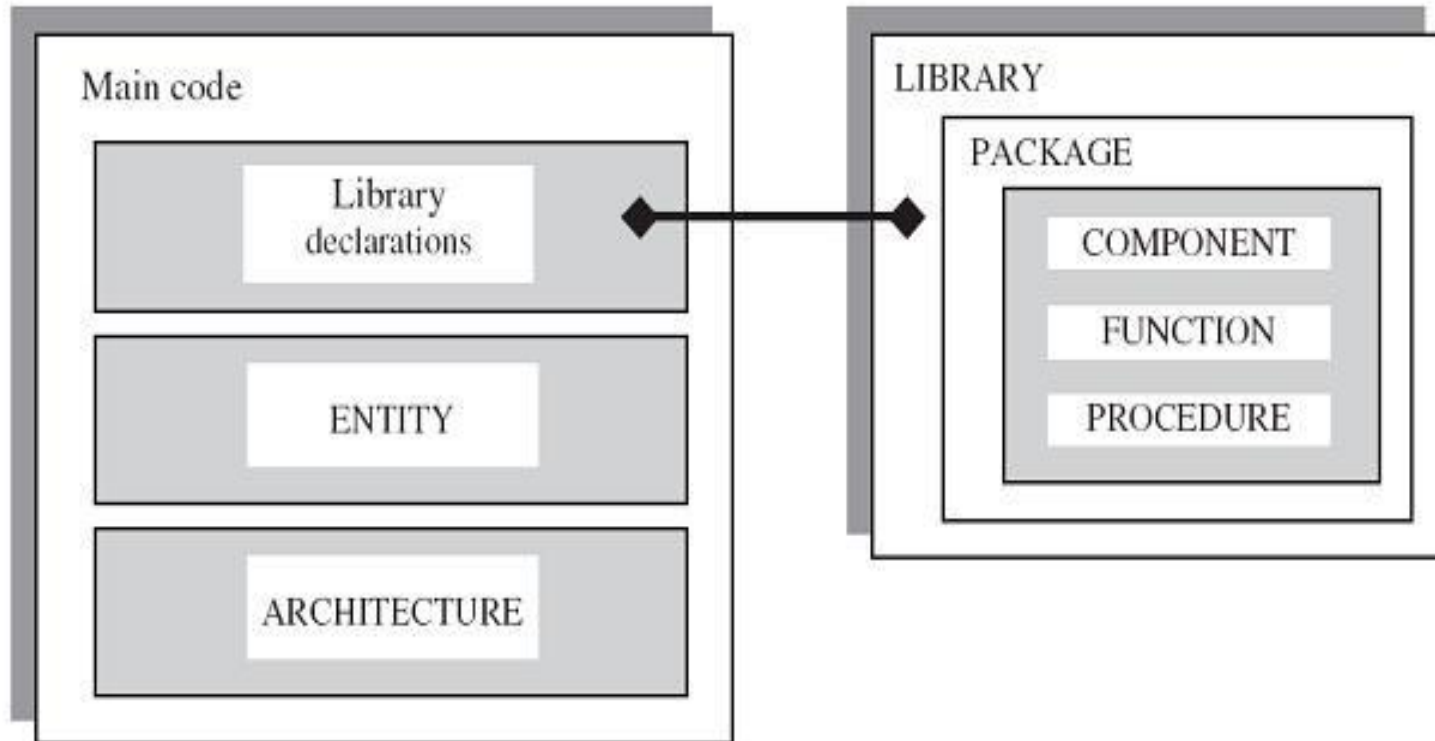


Pemrograman VHDL

- Ada beberapa dasar-dasar serta teknik penulisan kode dalam VHDL, yaitu:
 - *Entity*
 - *Architecture*
 - *Basic type variables and operator*
 - *Decisions*
 - *Loops*, dsb.

Arsitektur Model VHDL



Packages

- Component, function, dan procedure termasuk packages nantinya akan dikumpulkan menjadi satu dan disimpan dalam sebuah **library**.
- Struktur aturan penulisan Packages:

```
package name is
...package header contents
end package;
package body name is
... package body contents
end package body;
```

Packages

- Packages terdiri dari dua bagian:
 - Header adalah tempat untuk mendeklarasikan keseluruhan nama-nama berkas (baik function, procedures, atau component) yang diperlukan.
 - Body adalah tempat untuk menjabarkan lebih lanjut mengenai berkas-berkas yang tertulis di header.

Contoh

- Header:

```
package new_functions is
function and10 (a,b,c,d,e,f,g,h,i,j : bit) return bit;
end
```

- Body:

```
package body new_functions is
function and10 (a,b,c,d,e,f,g,h,i,j : bit) return bit is
begin
return a and b and c and d and e and f and g and h and i
and j;
end;
```

Contoh

- Penggabungan dari header dan body:

```
package new_functions is
function and10 (a,b,c,d,e,f,g,h,i,j : bit) return bit;
end;
package body new_functions is
function and10 (a,b,c,d,e,f,g,h,i,j : bit) return bit is
begin
return a and b and c and d and e and f and g and h and i
and j;
end;
end;
```

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
```

Definisi Library

```
entity tutorial is
  Port ( clk : in STD_LOGIC;
        led : out STD_LOGIC_VECTOR (7 downto 0);
        switch : in STD_LOGIC_VECTOR (3 downto 0));
end tutorial;
```

Definisi Port Input/Output

```
architecture Behavioral of tutorial is
```

```
  signal counter : std_logic_vector(29 downto 0);
```

Definisi counter

```
begin
```

```
  process(clk)
  begin
    if(clk'Event and clk='1') then
      if(switch(0)='1') then
        counter<=counter+'1';
      else
        counter<=counter-'1';
      end if;
    end if;
  end process;
```

Counter menghitung maju jika switch on, and menghitung Mundur jika switch off. Setiap penambahan terjadi pada setiap positive clock edge.

Library

- Library berfungsi untuk memudahkan programmer untuk menyelesaikan pekerjaannya karena dalam library tersebut terdapat fungsi-fungsi dan tipe data yang sudah didefinisikan sebelumnya untuk digunakan berulang-ulang.
- Library work:
 - *USE work.my_package.all;*

Library

- Library IEEE:
 - *use IEEE.std_logic_1164.all;*
 - *use IEEE.std_logic_textio.all;*
 - *use IEEE.std_logic_arith.all;*
 - *use IEEE.numeric_bit.all;*
 - *use IEEE.numeric_std.all;*
 - *use IEEE.std_logic_signed.all;*
 - *use IEEE.std_logic_unsigned.all;*
 - *use IEEE.math_real.all;*
 - *use IEEE.math_complex.all;*

Entity

- Entity adalah daftar dengan spesifikasi dari semua pin input dan output (port) dari sirkuit.

```
ENTITY entity_name IS
  PORT (
    port_name : signal_mode signal_type;
    port_name : signal_mode signal_type;
    ...);
END entity_name;
```

- Signal_mode → IN, OUT, INOUT, atau Buffer.
- Signal_type → BIT, STD_LOGIC, INTEGER, DLL.
- Nama *Entity* bebas kecuali kata-kata yang telah ditetapkan untuk VHDL, seperti architecture, entity, other, package, dll.

Architecture

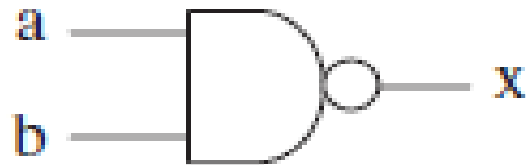
- Architecture adalah deskripsi tentang bagaimana rangkaian harus berperilaku (fungsi).

```
ARCHITECTURE architecture_name OF entity_name IS
    [declarations]
BEGIN
    (code)
END architecture_name;
```

- Arsitektur terdiri dari dua bagian, yaitu:
 - Declarations/deklaratif (optional) → dimana signal dan konstanta dinyatakan.
 - Code/program → dari BEGIN ke bawah.

Contoh

- Mari kita perhatikan gerbang NAND



```
ARCHITECTURE myarch OF nand_gate IS  
BEGIN  
    x <= a NAND b;  
END myarch;
```