

KONSEP DASAR FPGA (FIELD PROGRAMMABLE GATE ARRAY)

VERONICA ERNITA K.

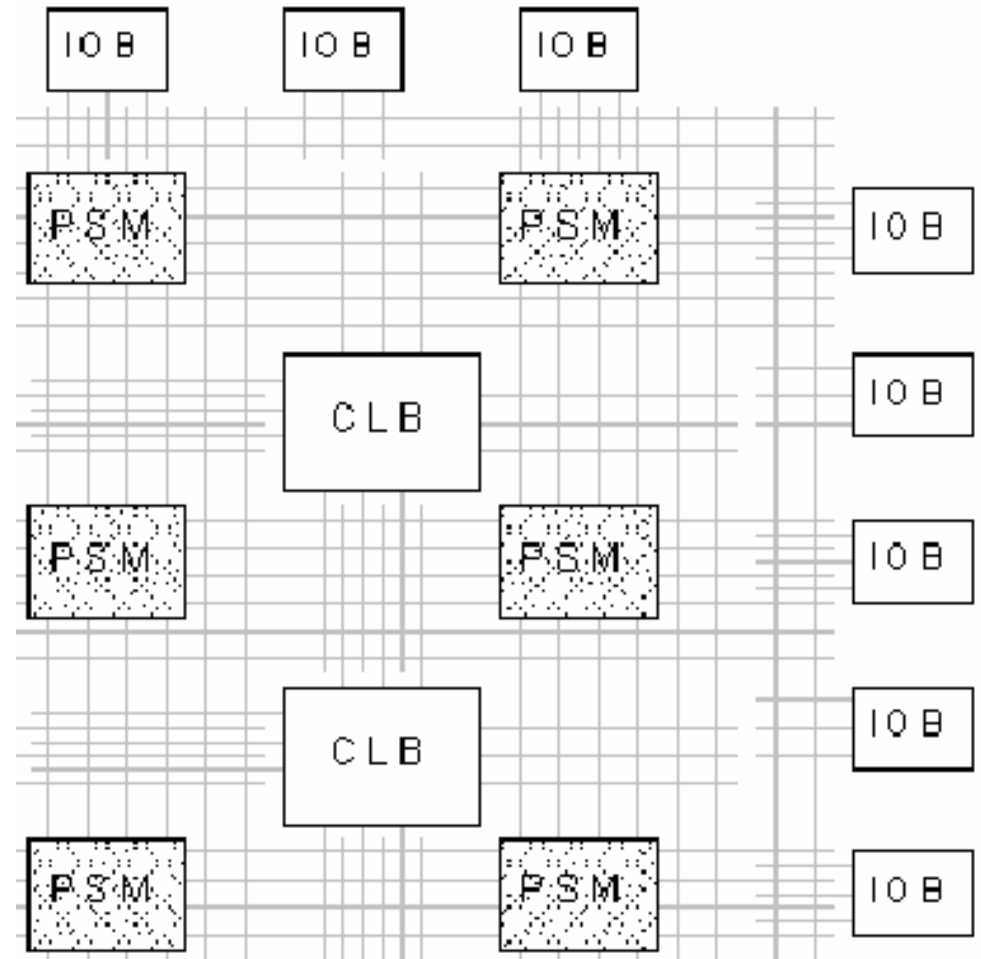
Field Programmable Gate Array (FPGA)

- Untuk mengimplementasikan rangkaian yang lebih besar lagi, digunakan tipe IC yang dapat mempunyai kapasitas logika yang lebih besar:
 - Field Programmable Gate Array (FPGA)
- Tidak berisi blok AND dan OR.
 - Tapi, berisi array blok logika dan jalur interkoneksi antar blok.
 - Jalur interkoneksi disusun sebagai kanal routing secara horisontal dan vertikal yang berisi programmable switch.
- Dapat mengimplementasikan fungsi logika dengan jumlah gerbang ekuivalen jutaan.

Field Programmable Gate Array (FPGA)

Terdiri dari:

- Configurable Logic Blocks (CLB).
- Input/Output Blocks (IOB).
- Programmable Switch Matrix (PSM)



Configurable Logic Blocks (CLB)

- CLB merupakan blok untuk membangun komponen-komponen combinational/sequential.
- Untuk dapat membentuk CLB dibutuhkan truth table dalam membangun jalur logika.

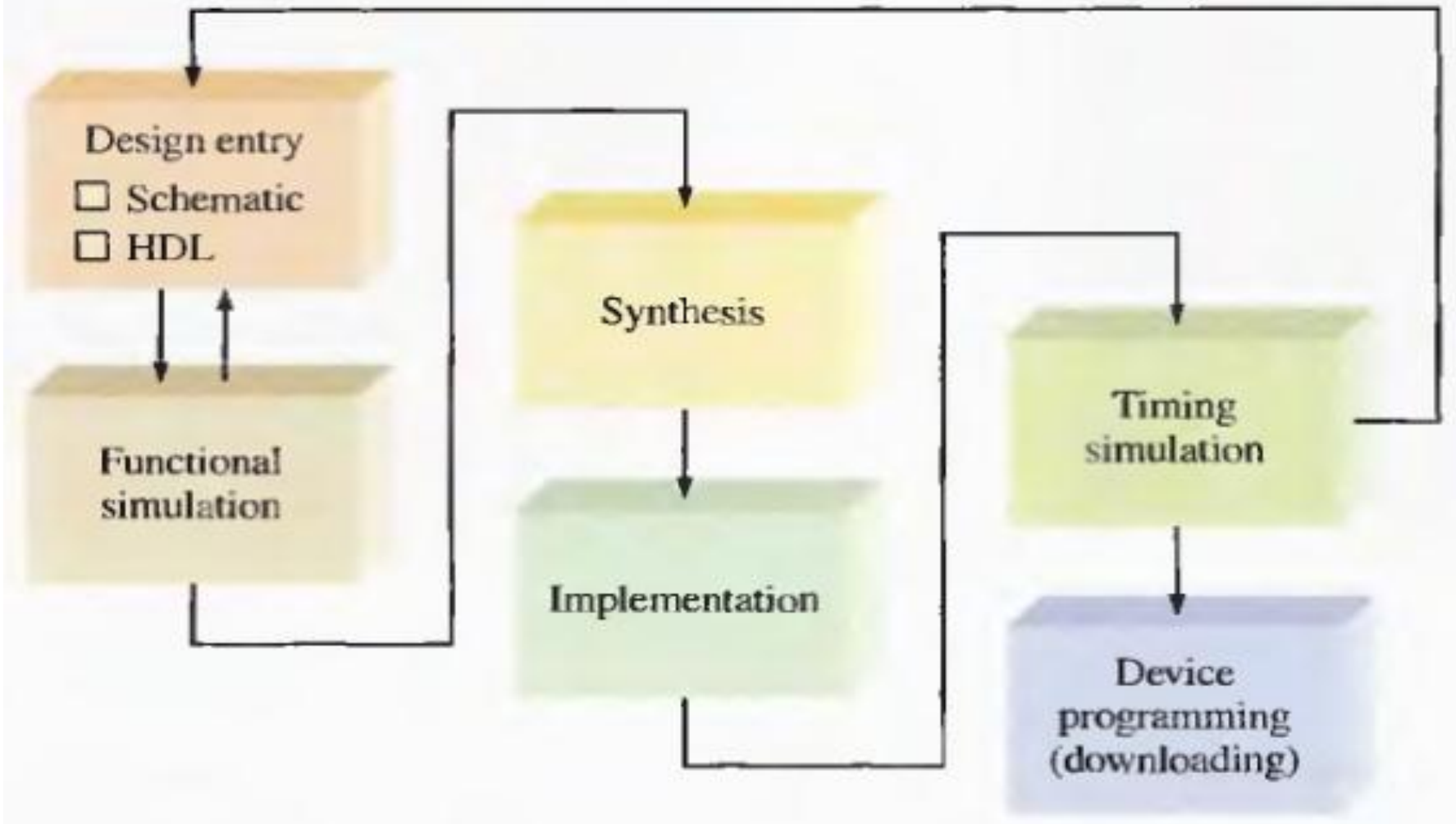
Input/Output Blocks (IOB)

- IOB merupakan blok yang digunakan untuk mengirimkan sinyal keluar dari chip dan sekaligus membaca sinyal yang masuk ke dalam chip.

Programmable Switch Matrix (PSM)

- PSM merupakan penghubung yang bisa diatur sedemikian rupa untuk menghubungkan antar CLB dan CLB dengan IOB secara horizontal maupun secara vertikal.
- Dapat mengeluarkan fan out (kemampuan untuk mengirimkan data ke banyak sumber untuk dijadikan input sekaligus) untuk multiple output.
- Jadi, output dari CLB yang satu dapat dihubungkan ke input CLB yang lain melalui multiple PSM.

Proses Pemrograman pada FPGA



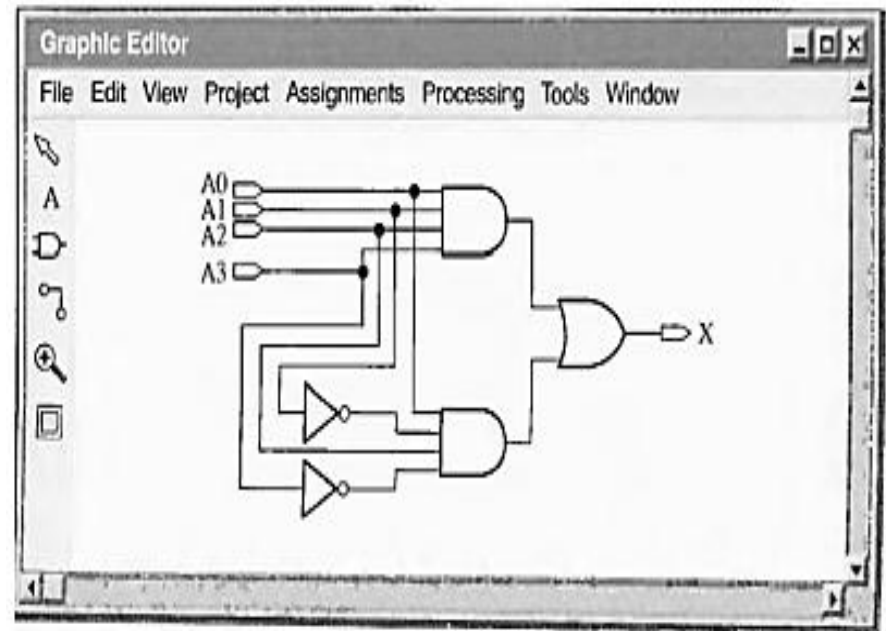
Design Entry

```
Text Editor
File Edit View Project Assignments Processing Tools Window

entity AND_OR is
  port (A0, A1, A2, A3: in bit; X: out bit);
end entity AND_OR;

architecture LogicFunction of AND_OR is
begin
  X <= (A0 and A1 and A2 and A3) or
        (A0 and not A1 and A2 and not A3);
end architecture LogicFunction;
```

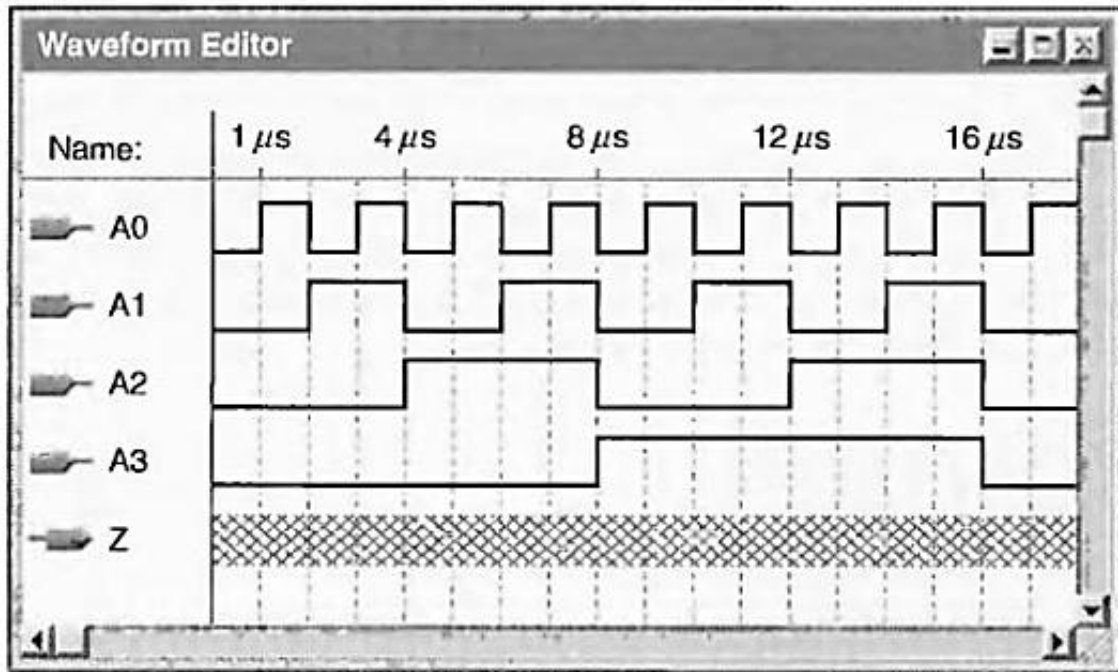
Text Entry menggunakan
**VHDL (Verylog Hardware
Definition Language)**



Schematic Entry

Functional Simulation

- Tujuan: memastikan rancangan rangkaian logika bekerja sesuai keinginan dengan *input waveforms* yang digunakan.



Synthesis

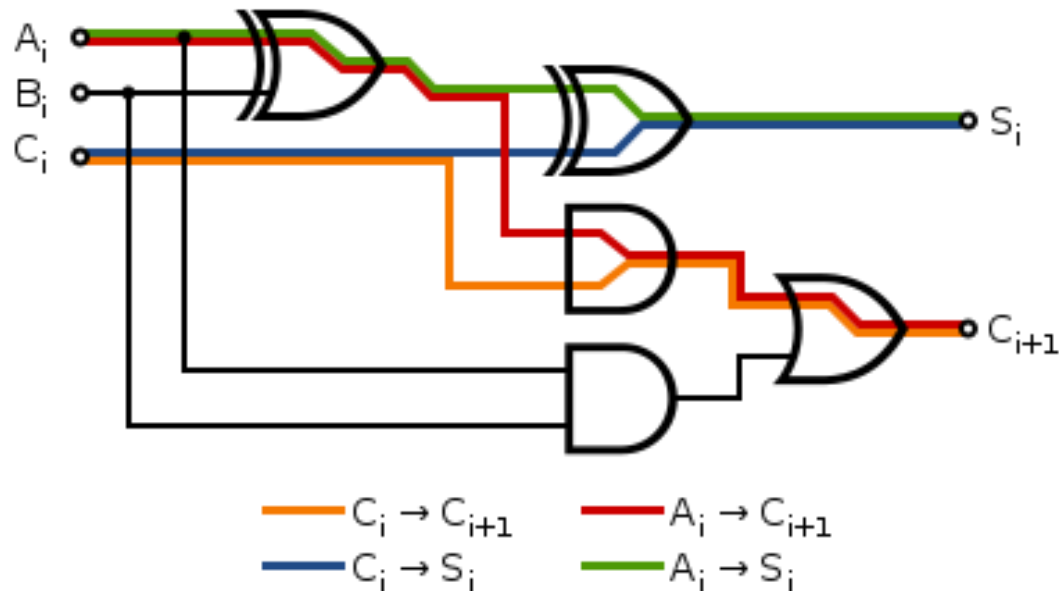
- Optimasi rangkaian logika dalam penggunaan gerbang.
- Menghasilkan netlist yang merupakan daftar connection yang mendeskripsikan komponen dan keterhubungan komponen tersebut.

Implementation

- Meliputi: *mapping, placing, and routing* rancangan sehingga dapat diimplementasikan ke IC FPGA sesuai **arsitektur** dan **konfigurasi** pin IC FPGA tersebut.

Timing Simulation

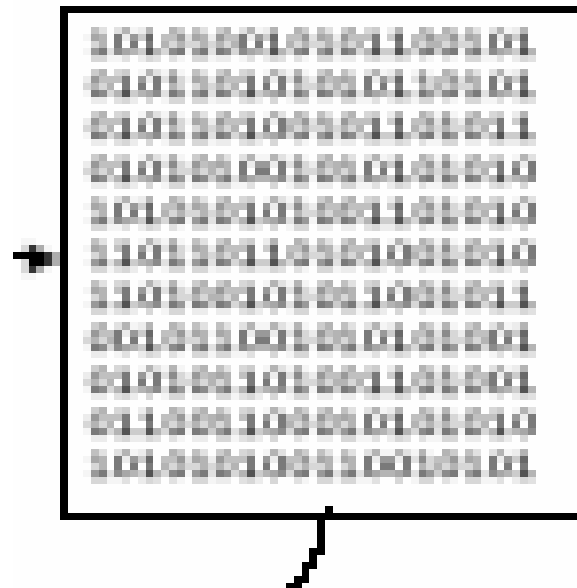
- Verifikasi rangkaian apakah sudah bekerja pada frekuensi yang diinginkan dan tidak ada *propagation delay*.



Device Programming

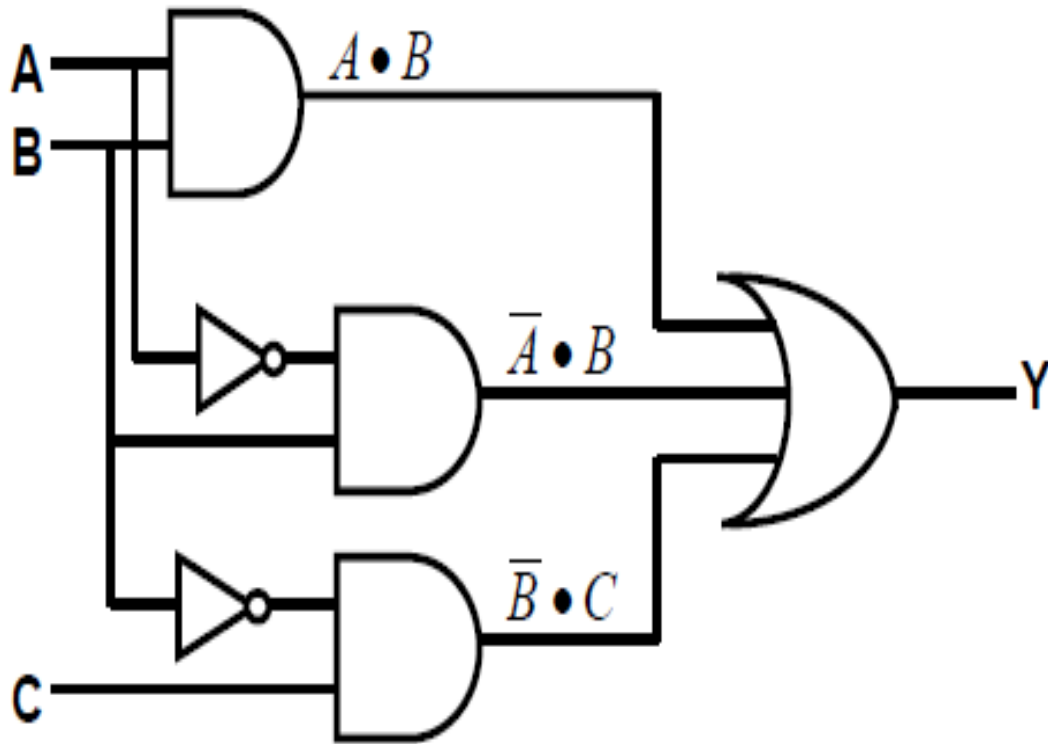
- Membuat *bitstream* yang merepresentasikan rancangan akhir dan akan dikirim ke target device.

Bitstream



Contoh

$$Y = A \cdot B + A' \cdot B + B' \cdot C$$



BEGIN

```
Y <= (A AND B) OR  
(NOT A AND B) OR  
(NOT B AND C);  
END;
```

LATIHAN

$$Y = (A + B + C) \cdot (\bar{A} + \bar{B})$$

$$Y = (A + B) \cdot (\bar{A} + \bar{B})$$

$$Y = (A + B) \cdot \bar{C}$$

$$Y = (A \cdot B) + (\bar{A} \cdot \bar{B})$$

$$Y = (\bar{A} \cdot B) + \bar{C}$$

$$Y = \bar{A} \cdot \bar{C} + A \cdot B \cdot C$$

$$Y = \bar{A} \cdot \bar{B} \cdot \bar{C} + A \cdot \bar{B} \cdot \bar{C}$$

$$Y = A \cdot \bar{B} \cdot C + \bar{A} \cdot B \cdot C$$

$$Y = A \cdot B \cdot C + \hat{A} \cdot \bar{B} \cdot C$$