

IL VHDL

Cosa è il VHDL?

NON è un linguaggio di programmazione!

E' uno standard IEEE per la descrizione dell'hardware

VHDL: VHSIC Hardware Description Language

VHSIC: Very High Speed Integrated Circuit

Perché si usa un linguaggio di descrizione dell'hardware?

Permette di formalizzare il progetto di sistemi digitali complessi

Si può descrivere l'hardware:

Strutturale: come deve essere

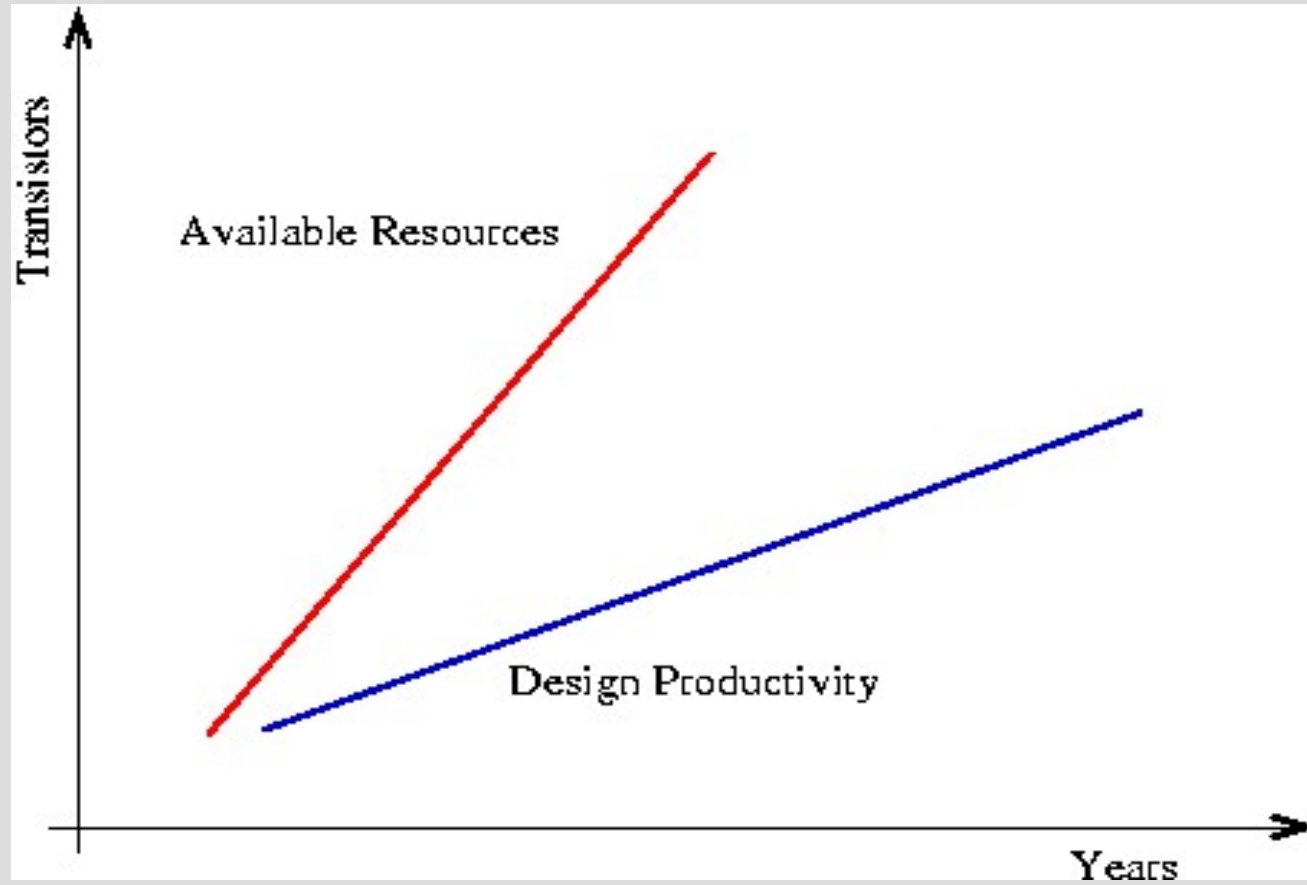
Comportamentale: cosa deve fare

**Con il VHDL si può descrivere l'hardware in tutti e due i modi
all'interno dello stesso progetto**

Obiettivi

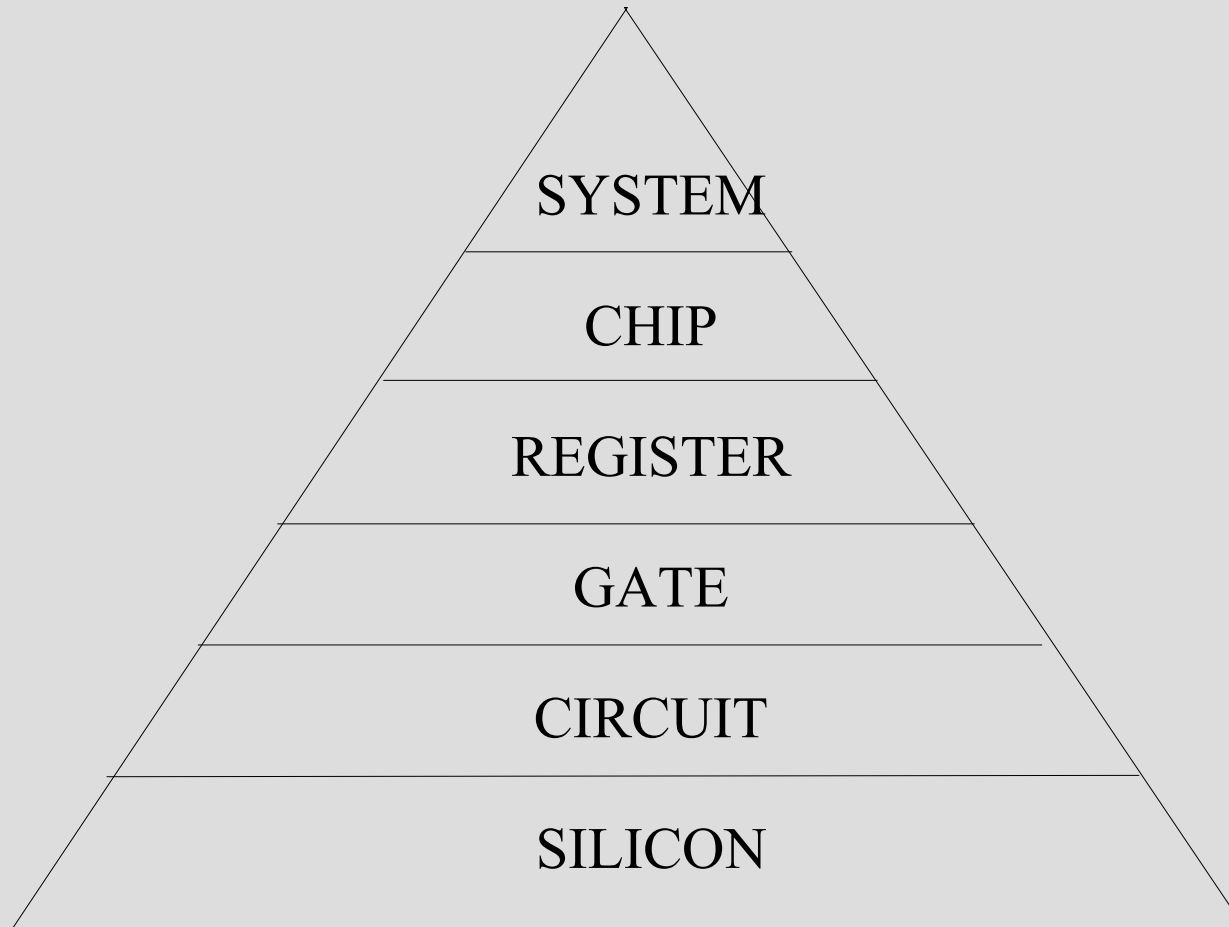
- **Imparare la sintassi e la semantica del VHDL**
- **Descrivere sistemi digitali a vari livelli di astrazione usando il VHDL**
- **Imparare il subset VHDL sintetizzabile**
- **Principi di simulazione e sintesi in VHDL**
- **Introduzione al progetto di ASIC e FPGA**

Il Design Productivity Gap

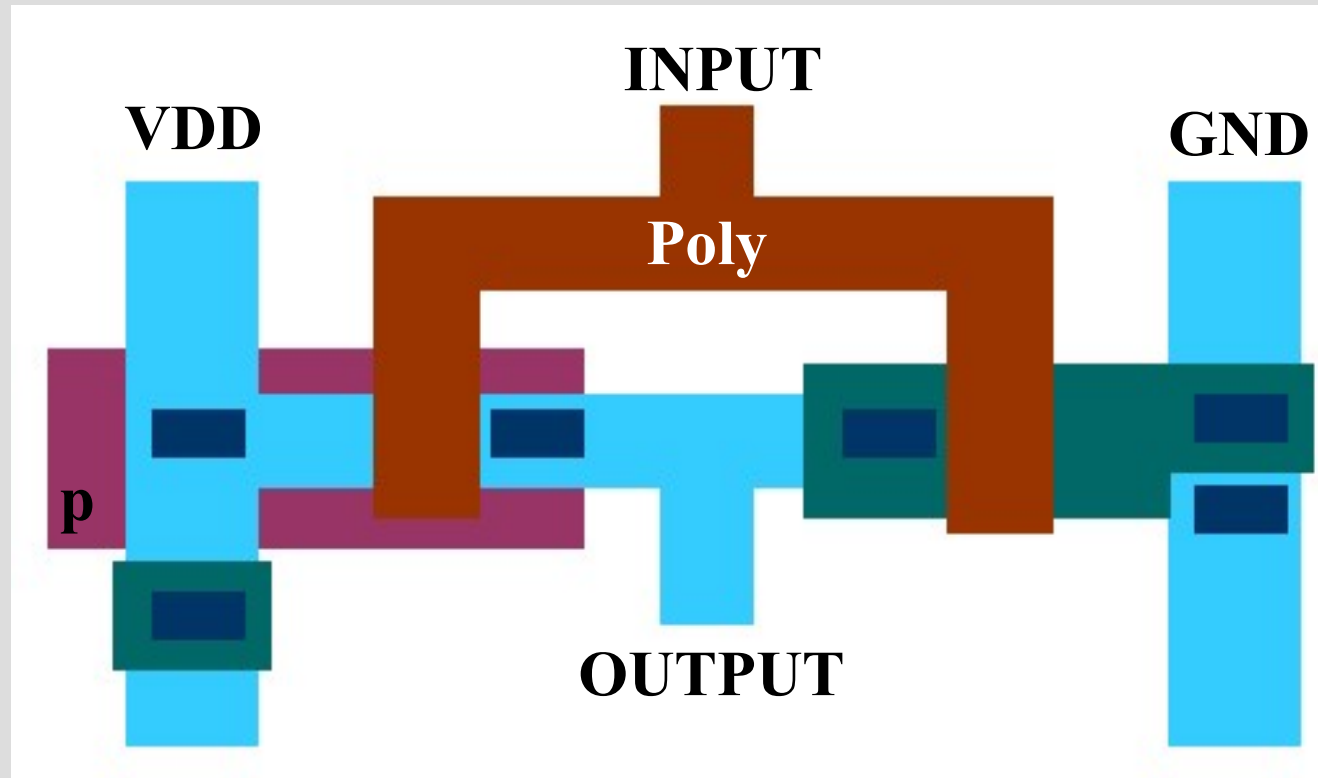


Lo sviluppo della tecnologia offre una quantità di risorse di calcolo che supera la capacità del progettista di utilizzarle.

Livelli di astrazione

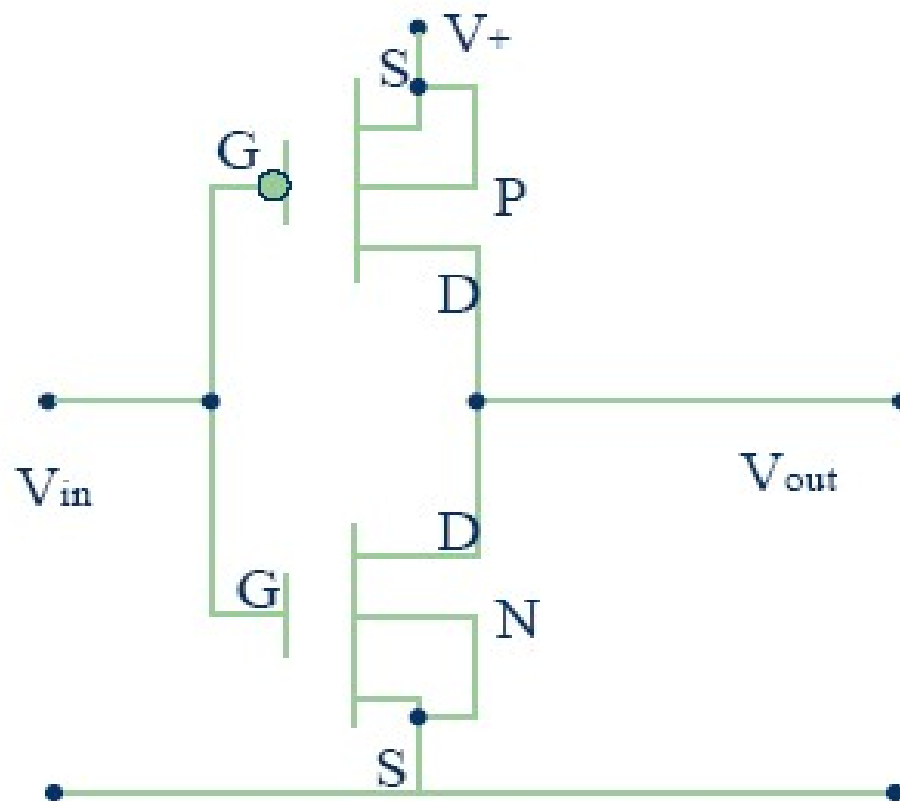


Livelli di astrazione: silicon level



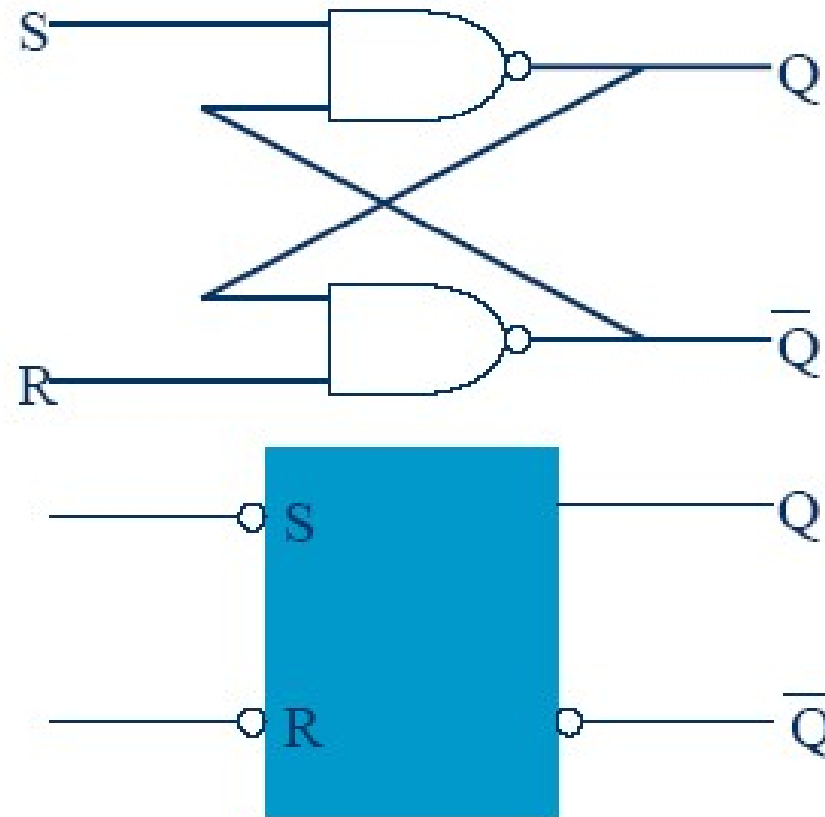
Livelli di astrazione: circuitale

Inverter

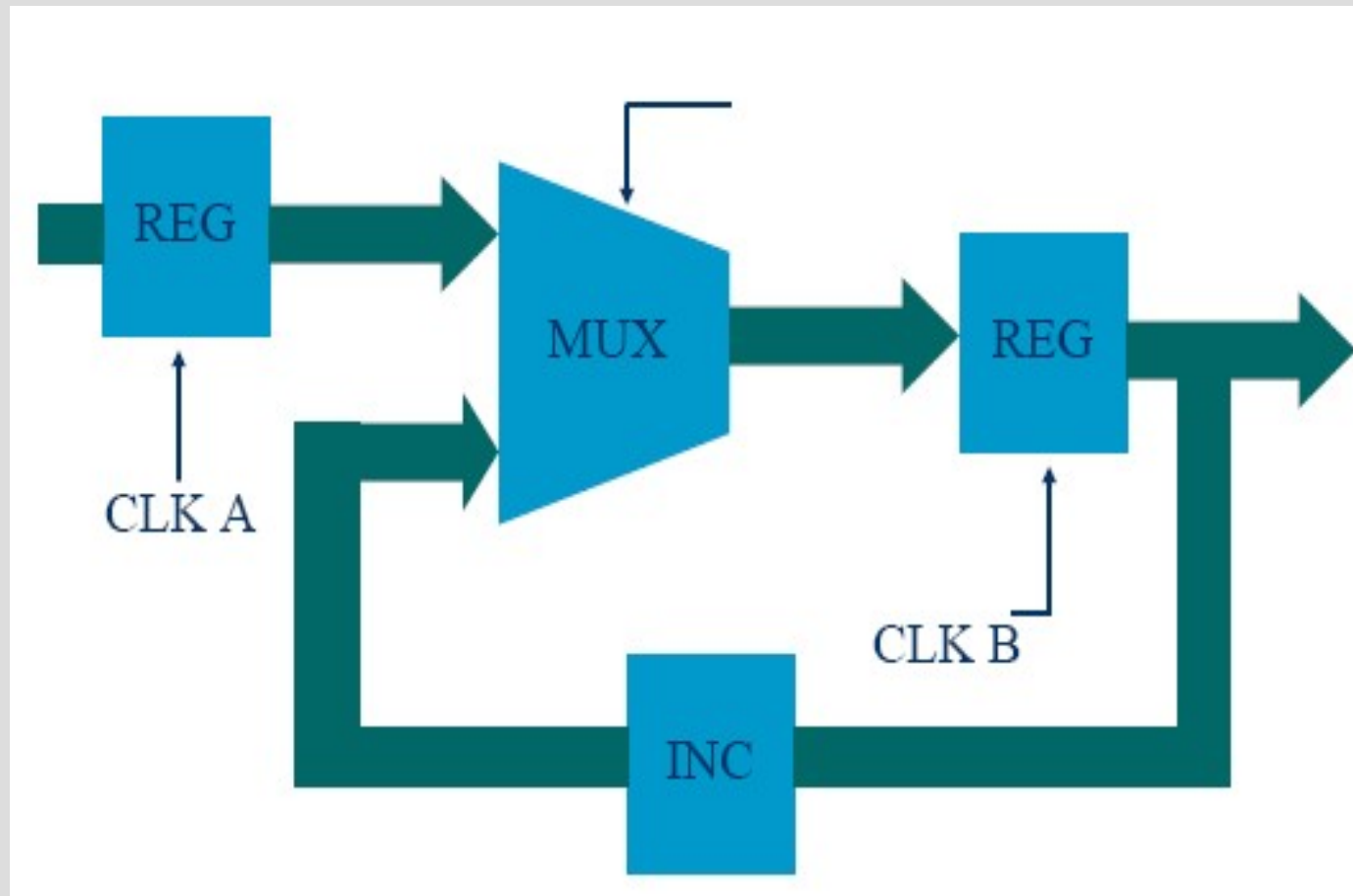


Livelli di astrazione: gate

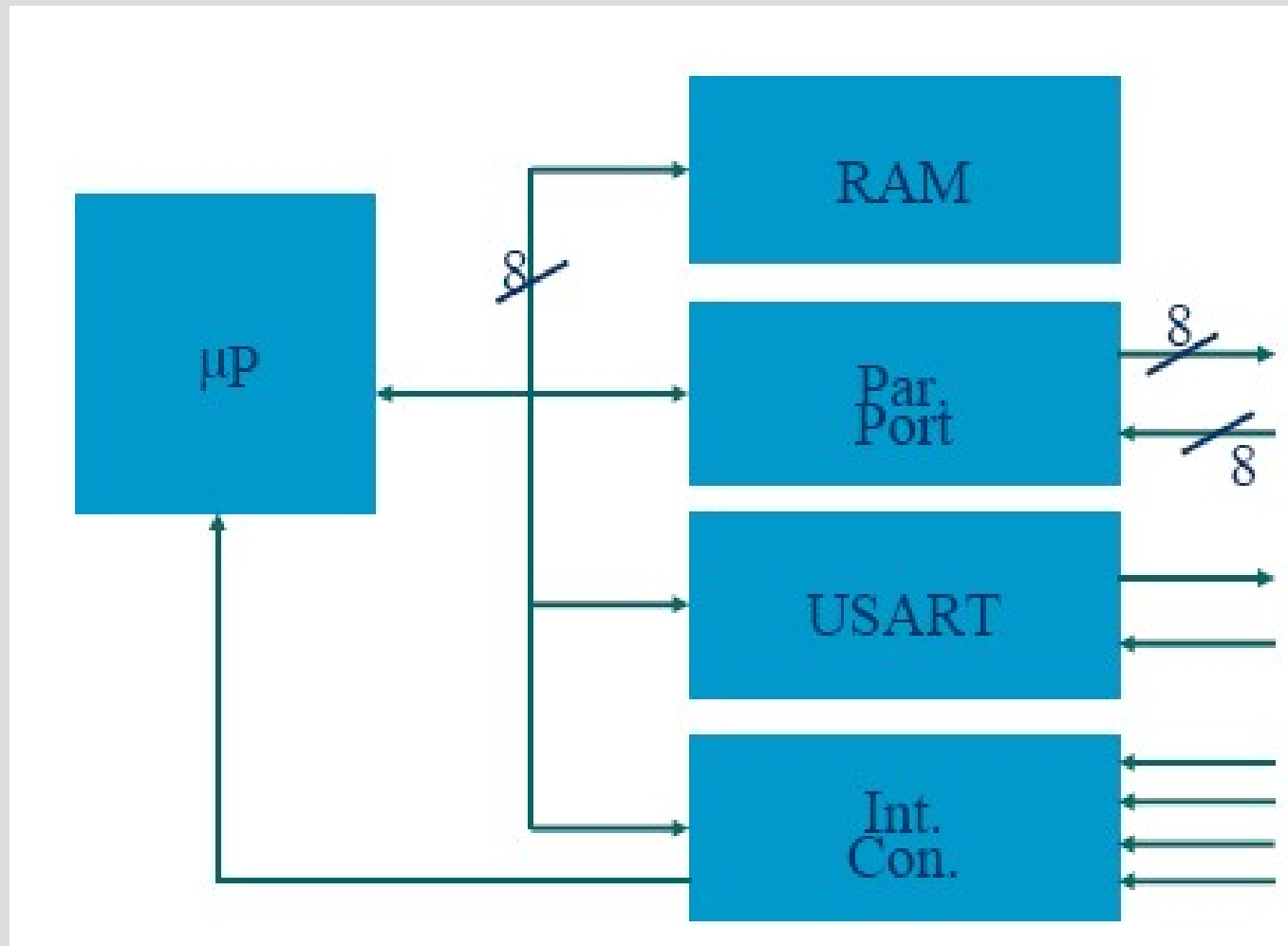
SR Flip Flop



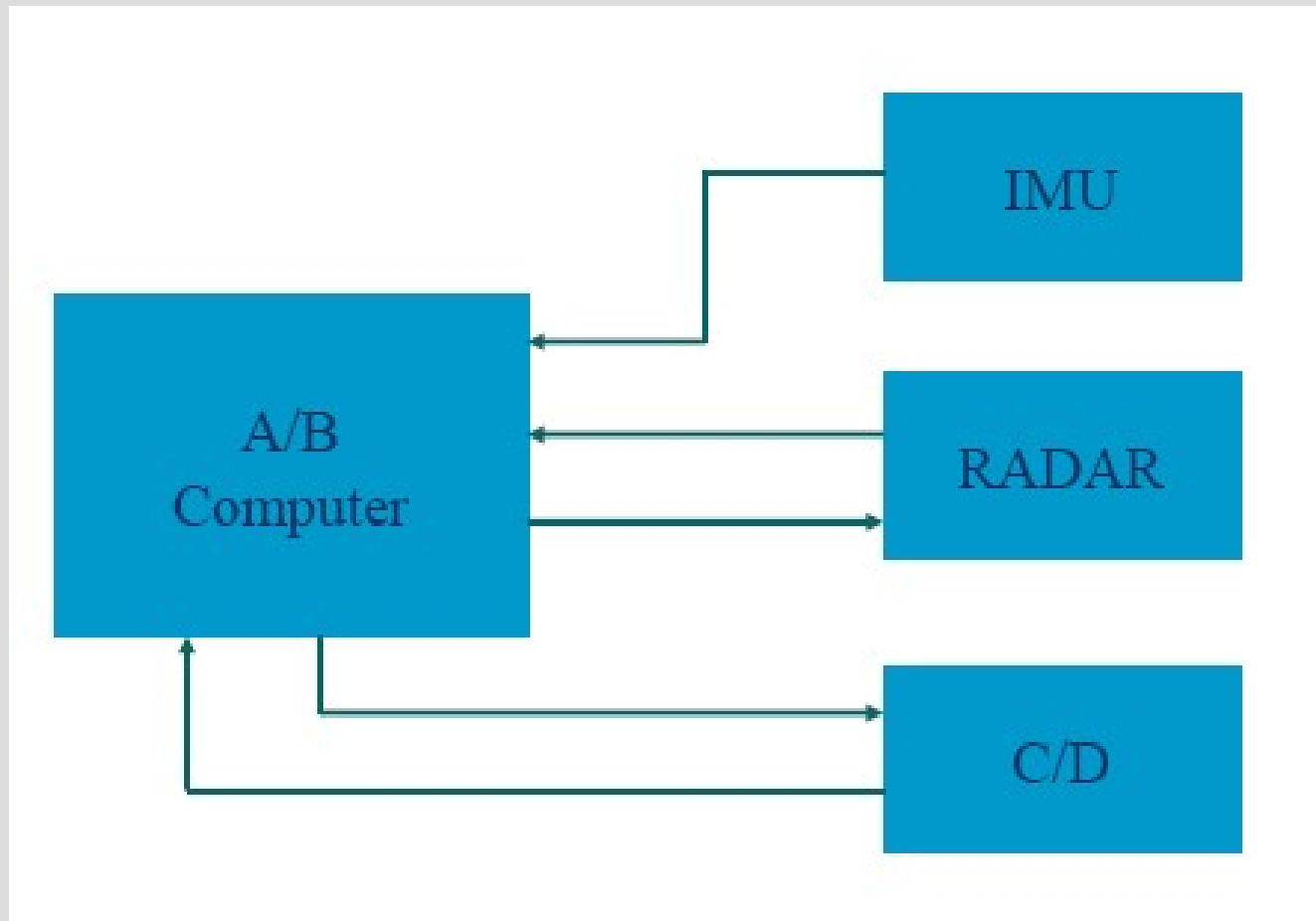
Livelli di astrazione: registri



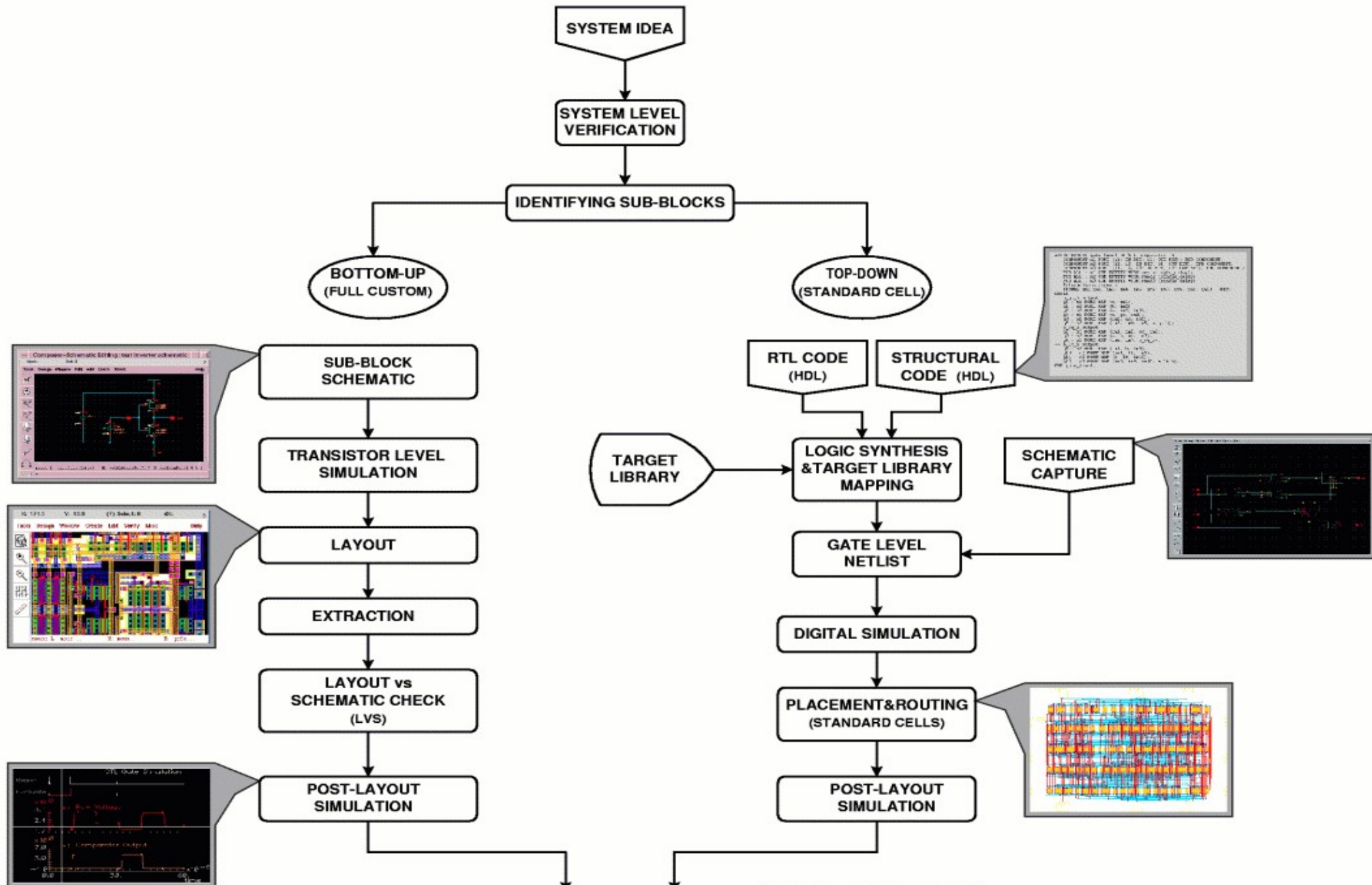
Livelli di astrazione: chip



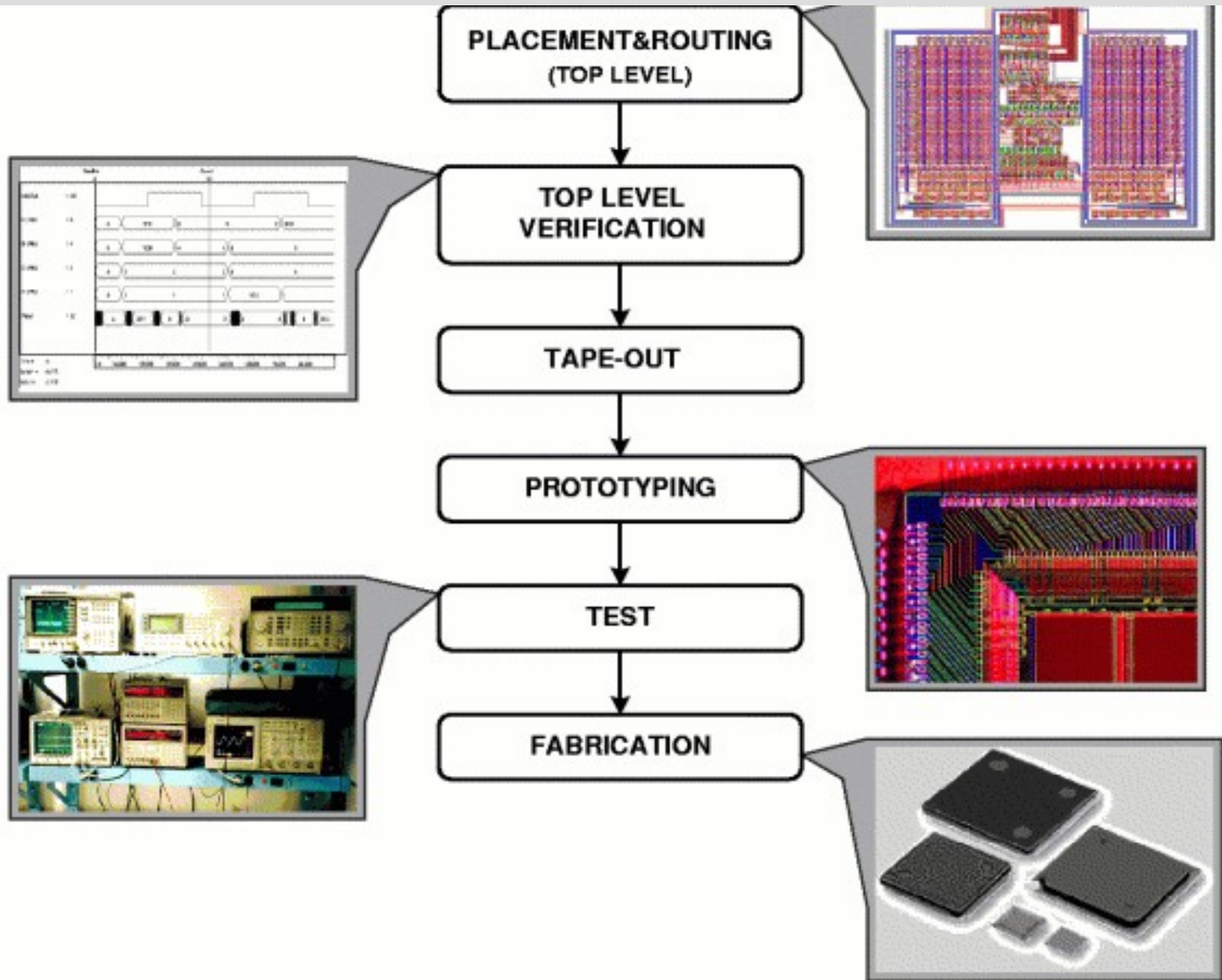
Livelli di astrazione: sistema



Flusso di progetto



Flusso di progetto



Differenze rispetto a un linguaggio di programmazione

- I sistemi hardware sono concorrenti
- I sistemi hardware possono essere “distribuiti” (il sistema è realizzato usando molti componenti)
- Frequenze diverse di funzionamento
- I sistemi hardware sono sistemi “timed”, quindi la gestione del timing è cruciale

conclusione: le tecniche di progettazione “software” sono inadatte

Vantaggi

- Progetto di tipo “technology -independent”
 - Impiegato da diversi fornitori / venditori (sia di HW che SW)
 - Facilita gli “aggiornamenti” del sistema progettato
- Miglioramento nella qualita’ del progetto
 - Consente di analizzare varie alternative
 - Consente piu’ livelli di astrazione
 - Verifica ad un elevato livello di astrazione
 - Paragone delle prestazioni tra vari livelli di astrazione
 - Integrazione tra blocchi sviluppati a vari livelli
 - Riutilizzo e condivisione di blocchi gia’ sviluppati

Vantaggi

- Permette di partizionare il progetto:
 - spazialmente: descrizione strutturale
 - temporalmente: descrizione funzionale
- Diminuisce la complessità del progetto supportando livelli di astrazione diversi
- Permette di partizionare il progetto con una distinzione tra funzionalità e interfaccia I/O

Vantaggi

Simulazione:

- Permette di verificare il funzionamento tramite simulazione: i vari livelli di descrizione possono essere simulati con gli stessi input
- Gli output possono essere confrontati per verificare la funzionalità del sistema

Sintesi:

- tool di sintesi possono generare l'hardware in maniera automatica riducendo i tempi di sviluppo e gli errori del progetto manuale

Considerazioni

- E' meno "immediato" di uno schema digitale
 - E' sintatticamente ... pesante
 - Essendo uno "Standard Aperto" vi si possono definire strutture e/o tipologie di variabili non direttamente collegabili alla struttura del circuito
 - Solo un sottoinsieme del VHDL e' fisicamente "sintetizzabile"
 - Esistono tools di sviluppo che consentono di ricavare descrizioni in VHDL partendo da Schemi, Macchine a Stati Finiti, Tabelle di verita', ecc....

Top - Down Design

- Tecnica di progetto che passa attraverso vari livelli di astrazione:
 - Algoritmico
 - RTL (register transfer level)
 - Porte Logiche a ritardo unitario
 - Porte logiche con ritardo effettivo
- Il passaggio da un livello a quello sottostante dipende da scelte progettuali, ed esempio
 - Tipologia di sommatore usato (Ripple Carry, Carry Look Ahead, Carry Select)
 - Tipologia di porte logiche a disposizione

Top - Down Design

- I vari livelli di astrazione servono a “mascherare” il funzionamento al livello piu’ basso
 - A livello di funzione (a livello algoritmico non interessa conoscere la tipologia del “sommatore”, ma solo la funzione “somma”)
 - A livello di variabili (il VHDL consente l’impiego di variabili (boolean, integer, floating, ecc.) utili solamente per una modellizzazione astratta del sistema o di parti di esso.

Simulazione

- Un sistema descritto in VHDL viene solitamente SIMULATO per analizzarne in comportamento (simulazione comportamentale)
- Bisogna fornire degli stimoli (INPUT)
- Ed avere un sistema capace di osservare l'evoluzione del modello durante la simulazione, registrarne le variazioni per un'eventuale ispezione di funzionamento
- Il simulatore deve aver la possibilità' di rappresentare una variabile come "unknown".
- Package STD_LOGIC_1164

Sintesi Logica

- Passaggio tra descrizione comportamentale e descrizione a porte logiche
- La sintesi avviene tramite appositi programmi che si appoggiano a librerie dove sono descritte le porte logiche da impiegare (fornite dal venditore)
- Solo un ristretto sottoinsieme del VHDL si presta ad essere Sintetizzato automaticamente ovvero:
Non tutto ciò che è scritto in VHDL è sintetizzabile
- La restante parte è da impiegarsi per la descrizione e per la simulazione

Tecnologie per sistemi digitali

- **Full custom:**

tutte le celle logiche sono progettate a livello di lay-out

- **Standard cell:**

si utilizzano celle logiche predisegnate (AND gates, OR gates, MUXs, FFs)

- **Gate array:**

Array di transistor che possono essere interconnessi per implementare il chip

- **FPGA:**

blocchi logici programmabili collegati da matrici di interconnessione programmabili

If you can write this, you can build this...

VHDL

```
architecture STRUCTURAL of CHIP is
  component NAND_GATE
    port (A, B: in bit; Z: out bit);
  end component;

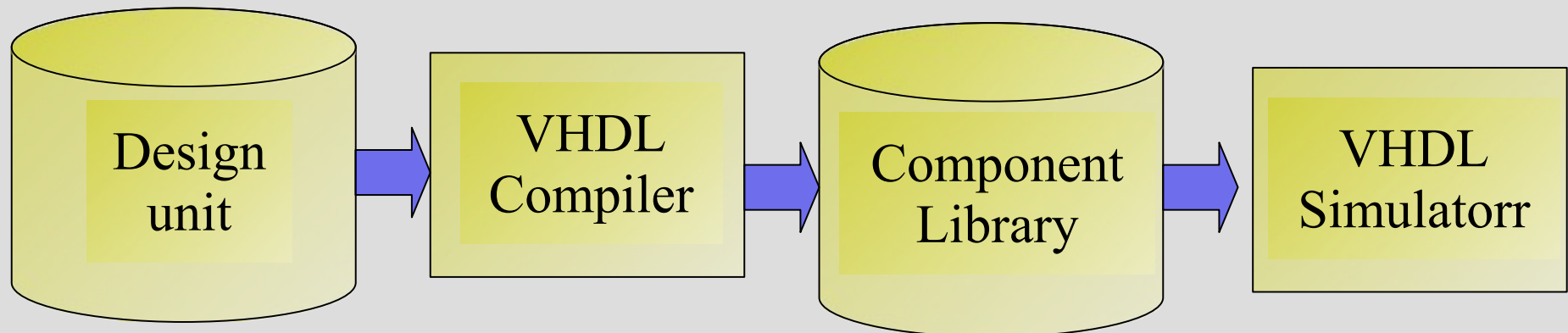
  component INV_GATE
    port (A: in bit; Z: out bit);
  end component;
  signal X : bit;

begin
  u0: NAND_GATE
    port map (IN1, IN2, X);
  u1: INV_GATE
    port map (X, OUT1);
end STRUCTURAL;
```



Programmare in VHDL

- Per certi versi e' simile a programmare in "Fortran", "C", "Pascal", ecc...
- Compilazione:
controllo delle sintassi, generazione di un eseguibile
- Running:
Link con strutture in libreria ed esecuzione dell'algoritmo



Programmare in VHDL

- La fondamentale differenza e' che un modello descritto in VHDL funziona sempre in "Tempo di simulazione" e tutti i vari eventi avvengono in successivi istanti di tempo
- Vi e' la possibilita' da parte di piu' istruzioni di essere eseguite CONTEMPORANEAMENTE indipendentemente dall'ordine in cui sono state scritte (esecuzione concorrente)
- Altre differenze:
 - Presenza del "delay" e dell'ambiente di simulazione
 - Descrizione a piu' livelli di astrazione
 - Descrizione di componenti ed "Istanziamento" all'interno di una descrizione strutturale

Progettare in VHDL

- Consta di piu' unita' compilate e salvate in opportune librerie
- Queste unita' sono:
 - Entity
 - Architecture
 - Configuration
 - Package
- Entity ed Architecture descrivono i componenti come interfaccia e come struttura interna
- Configuration serve per descrivere una particolare versione del progetto
- Package: contiene funzioni e/o grandezze di uso comune

Descrizione Strutturale

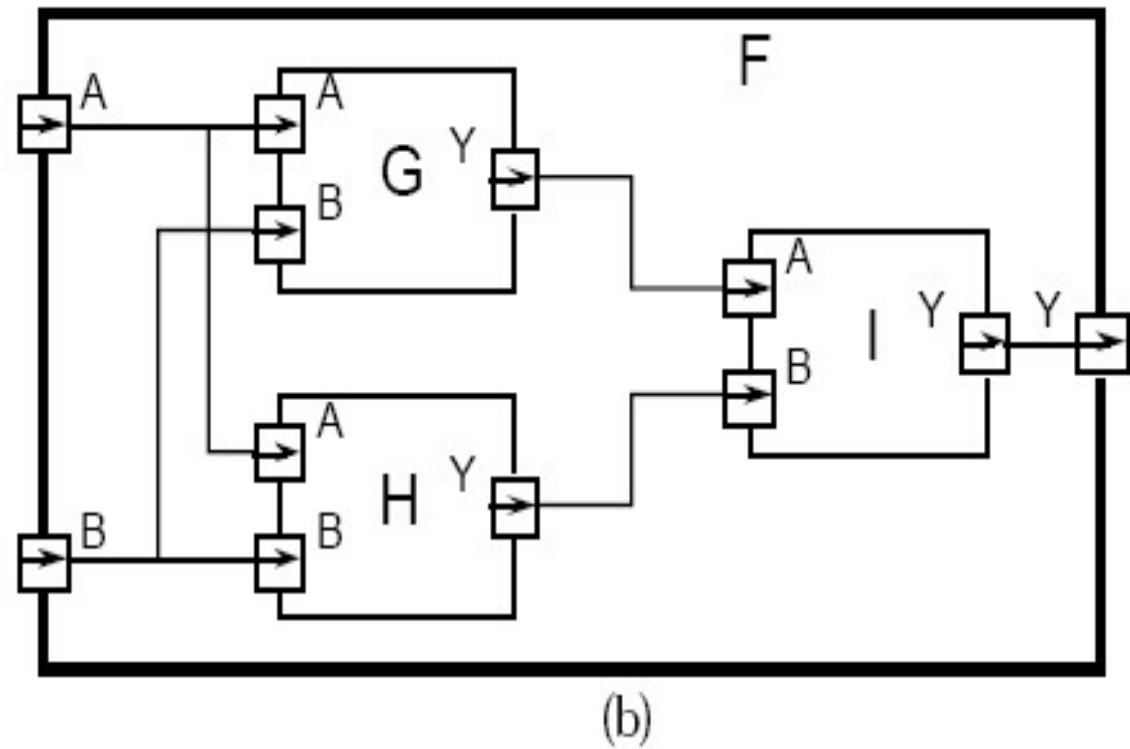
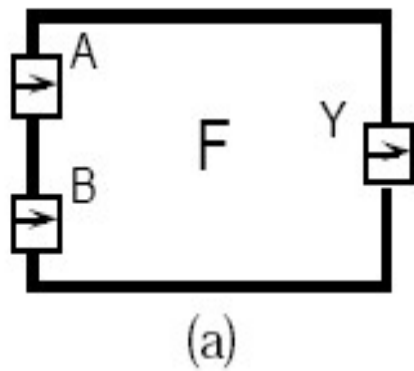
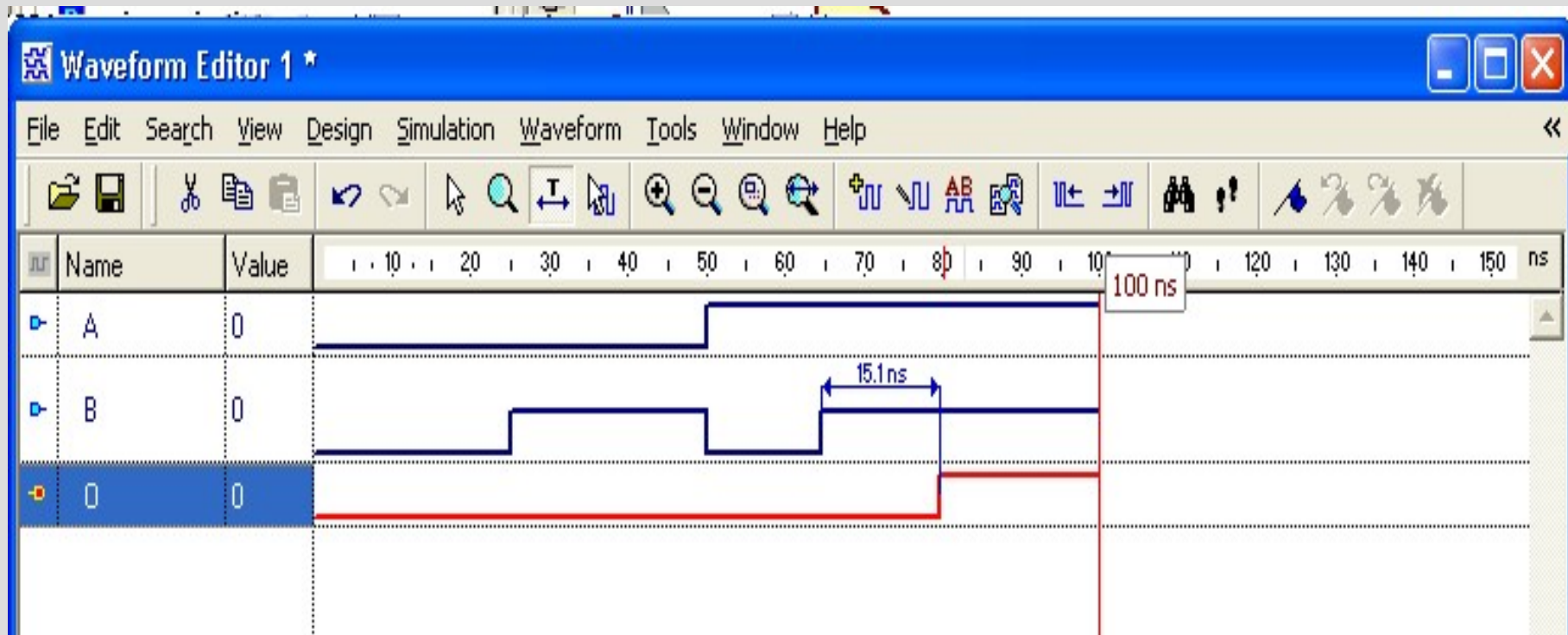
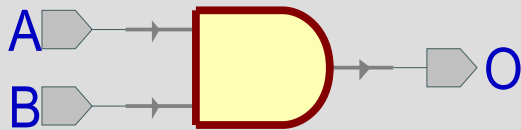


Figure 1-1. Example of a structural description.

Descrizione Comportamentale



Permette di descrivere il comportamento logico (sintesi) e temporale (simulazione)