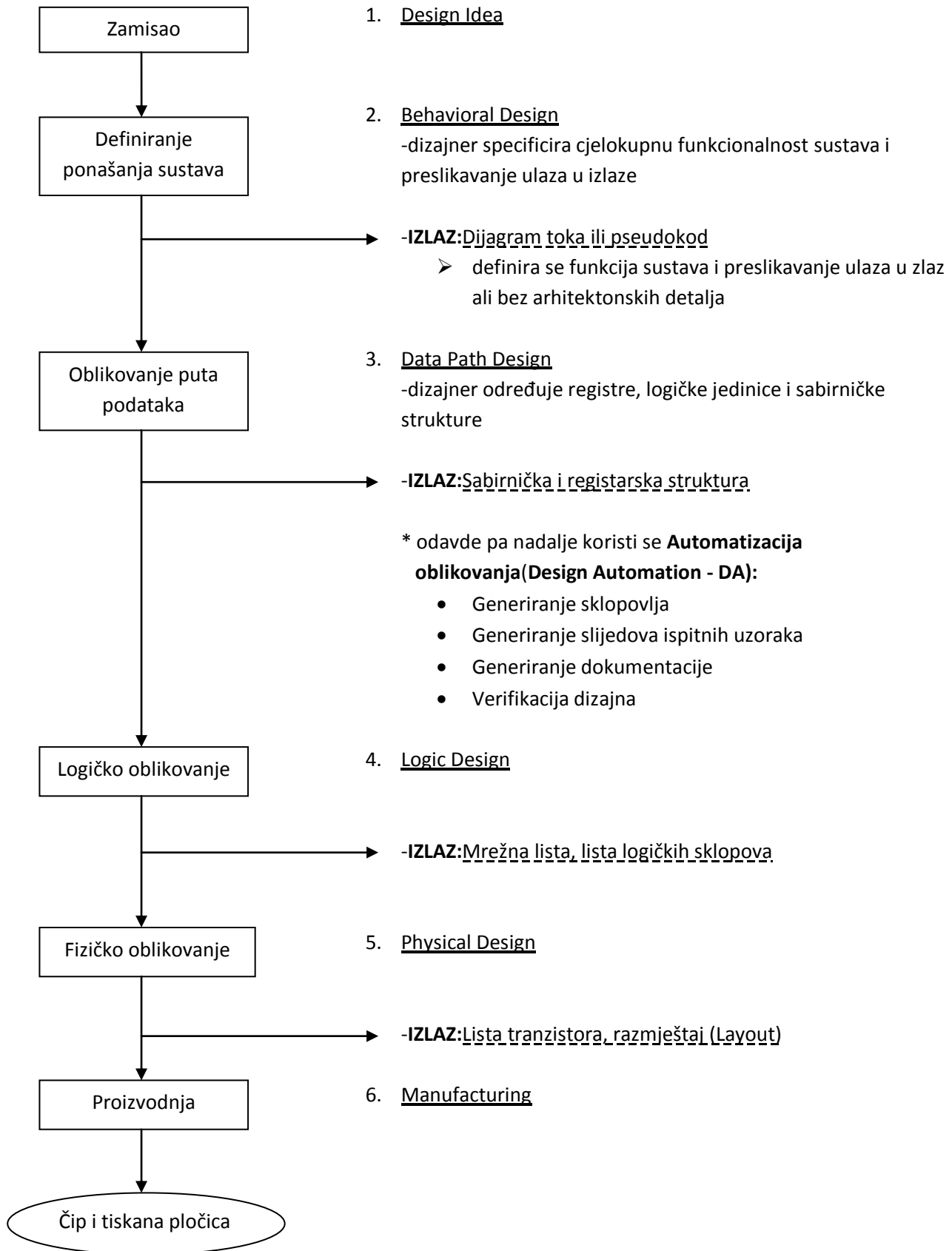
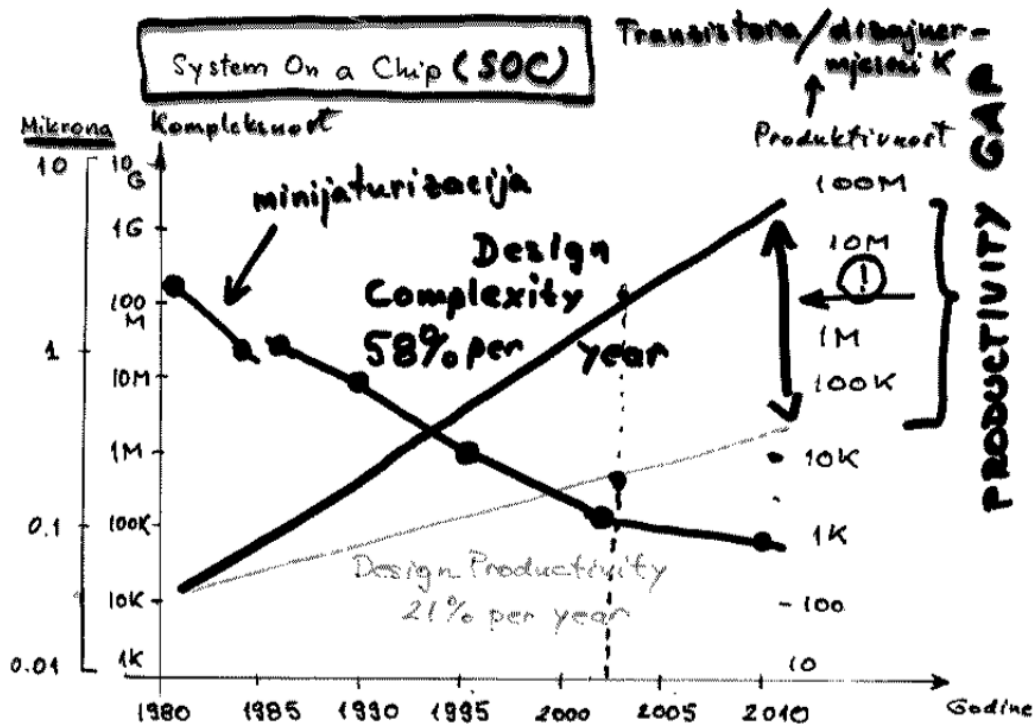


PROCES OBLIKOVANJA DIGITALNOG SUSTAVA



Automatizacija oblikovanja

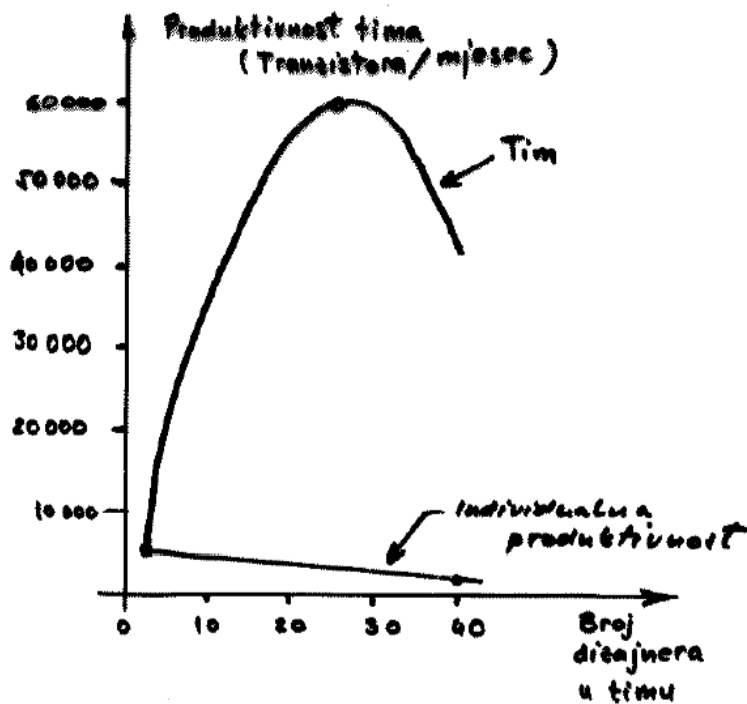
- s motrišta dizajnera faza oblikovanja je završena kada se zamisao transformira u arhitekturu ili u opis puta podataka
- ostale aktivnosti su rutinski posao koji stroj može bolje obaviti od inženjera
- 5% - 10% je kreativni dio digitalnog dizajna, ostalo je mukotrpan posao („turning te crank“)



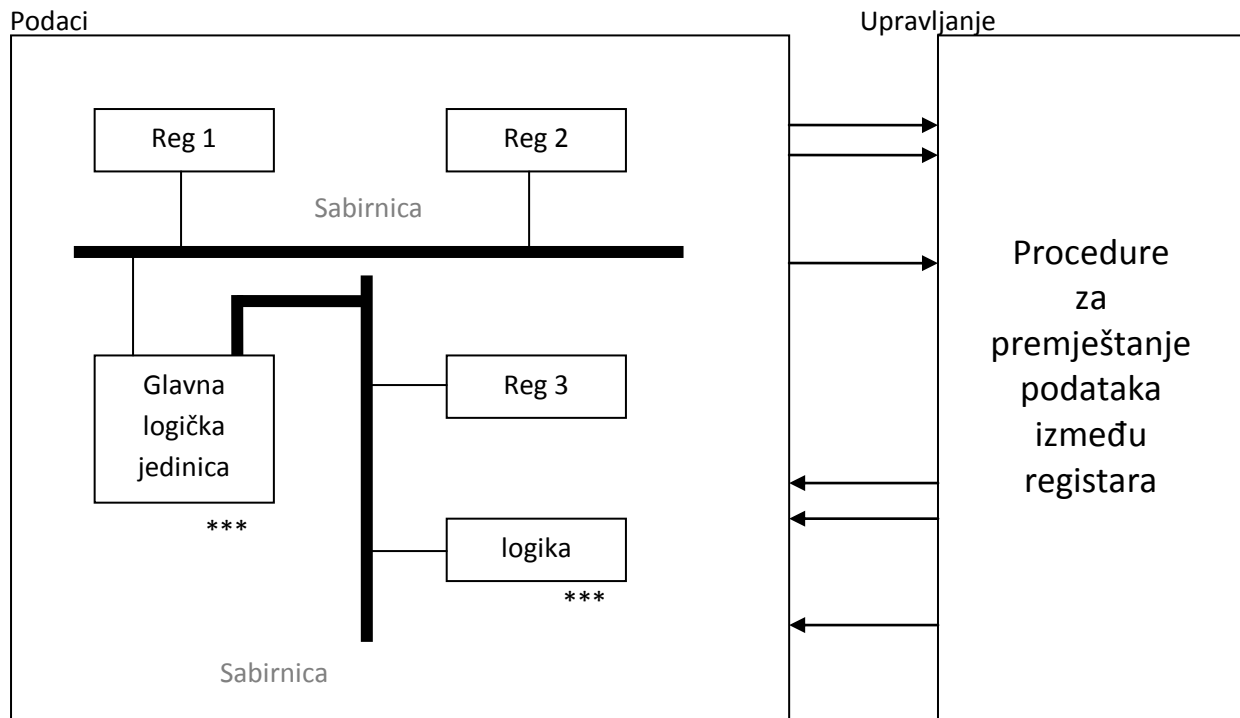
Produktivnost dizajnera : 1981.g. - 100 dizajner-mjeseci

2002.g. - 30000 dizajner-mjeseci za leading-edge chip

30000×5000 tranzistora/dizajner-mjeseci = 150000000 tranzistora

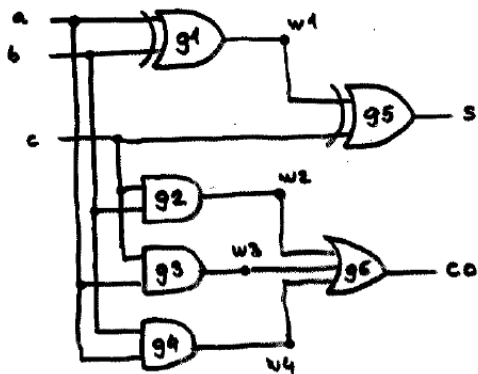


-mogući rezultat faze oblikovanja koja se odnosi na putove podataka:



Primjer:

- logički dijagram
- mrežna lista specificirana Verilogom



```
timescale 1 ns/1 ns
// A 6-gate full adder
module fulladder(s,co,a,b,c);
//Port declarations
output s,co;
input a,b,c;
//Intermediate wires
wire w1,w2,w3,w4;
//Netlist descriptions
xor #(16,12)g1(w1,a,b);
xor #(16,12)g5(s,w1,c);
and #(12,10)g2(w2,c,b);
and #(12,10)g3(w3,c,a);
and #(12,10)g4(w4,b,a);
or #(12,10)g6(co,w2,w3,w4);
endmodule
```

- transformacija iz jedne razine u drugu (od 4. – 6.), verifikacija rezultata oblikovanja i/ili generiranje ispitnih podataka → DA (Design Automation)
- alati DA pomažu dizajneru generiranjem sklopovlja, generiranju slijedova ispitnih uzoraka, dokumentaciji, verifikaciji i rukovanju postupkom oblikovanja
 - npr. za verifikaciju rezultata faze *Oblikovanja puta podataka* sabirnička i registarska struktura može biti ulaz u simulacijski program
- sintetizator DA može automatski generirati mrežnu listu na temelju sabirničke i registarske strukture
HDL – based DA tool

MODEL I MODELIRANJE

Model - shematski opis sustava, teorije ili fenomena koji uzima u obzir njegova nova ili zaključena svojstva i može poslužiti za daljnju njegovu analizu

Modeliranje → prikazivanje sustava na određenoj razini apstrakcije

Raspoloživi alati za modeliranje:

- papir i olovka
- eksperimentalne pločice
- HDL – Hardware Design Languages – jezici za opisivanje i oblikovanje sklopova („sklopovski jezici“)
- posebni jezici za modeliranje (npr. UML – Unified Modeling Language)

APSTRAKCIJA (eng. Abstraction)

- pojednostavljeni ili selektivni opis sustava koji naglašava (ističe) neka svojstva dok potiskuje druga

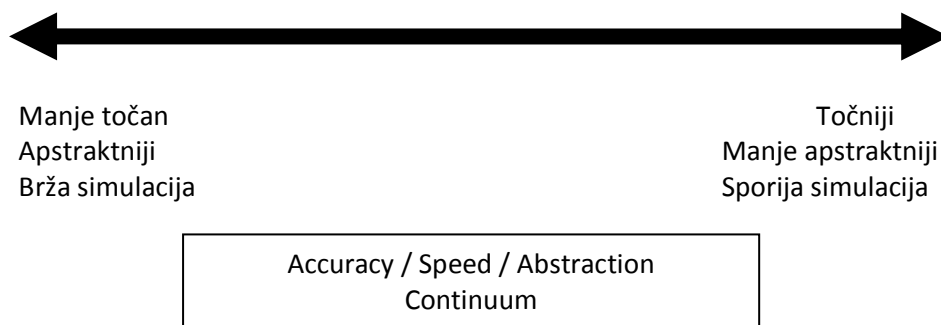
POSTUPKOVNA APSTRAKCIJA (eng. Procedural Abstraction)

- opisuje ponašanje nekog algoritma ili postupka dok ignorira detalje kako se takvo ponašanje ostvaruje

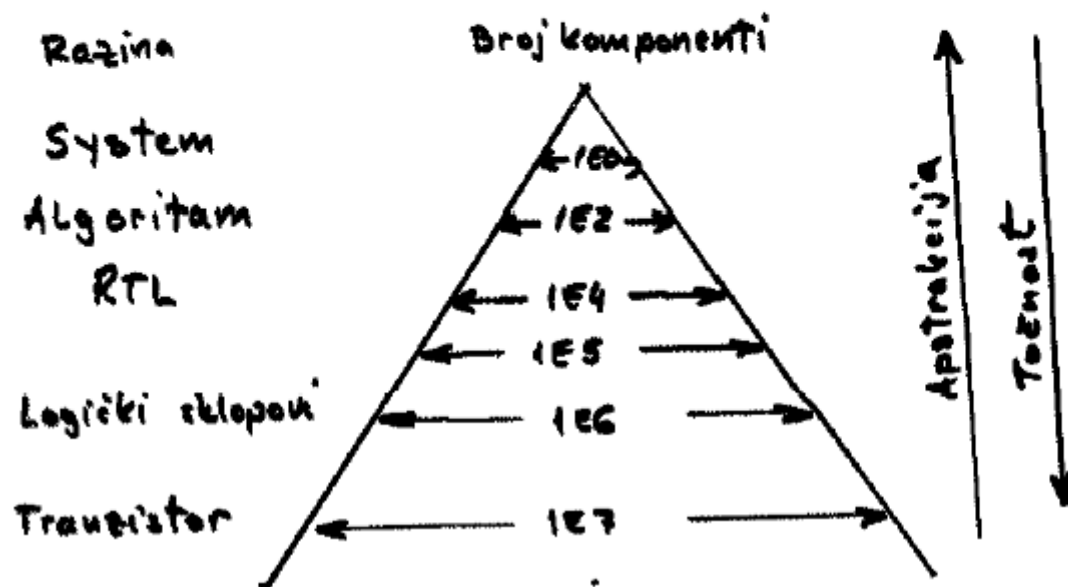
APSTRAKCIJA PODATAKA (eng. Data Abstraction)

- definira „ponašanje“ podatkovnih objekata u smislu operacija

Prikaz:

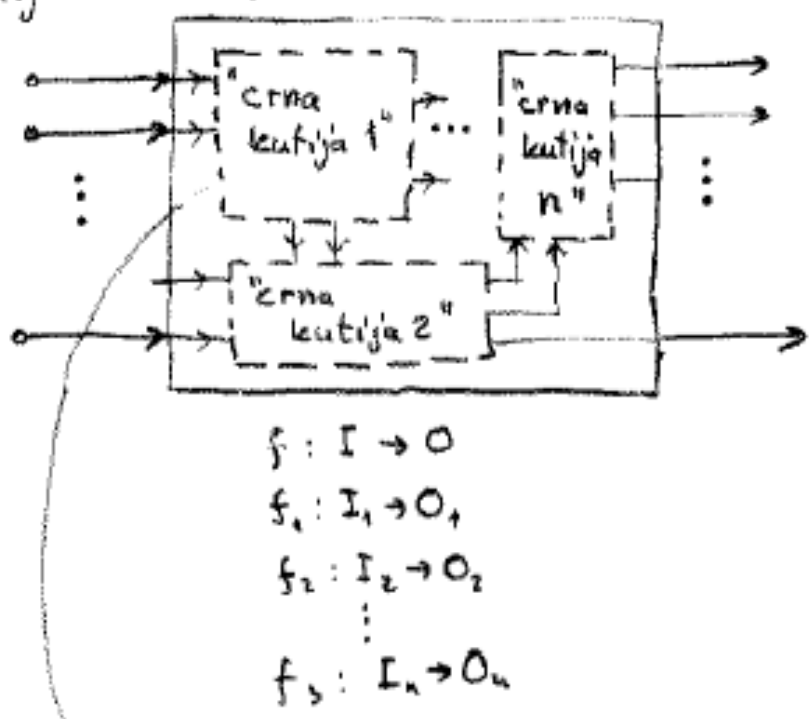
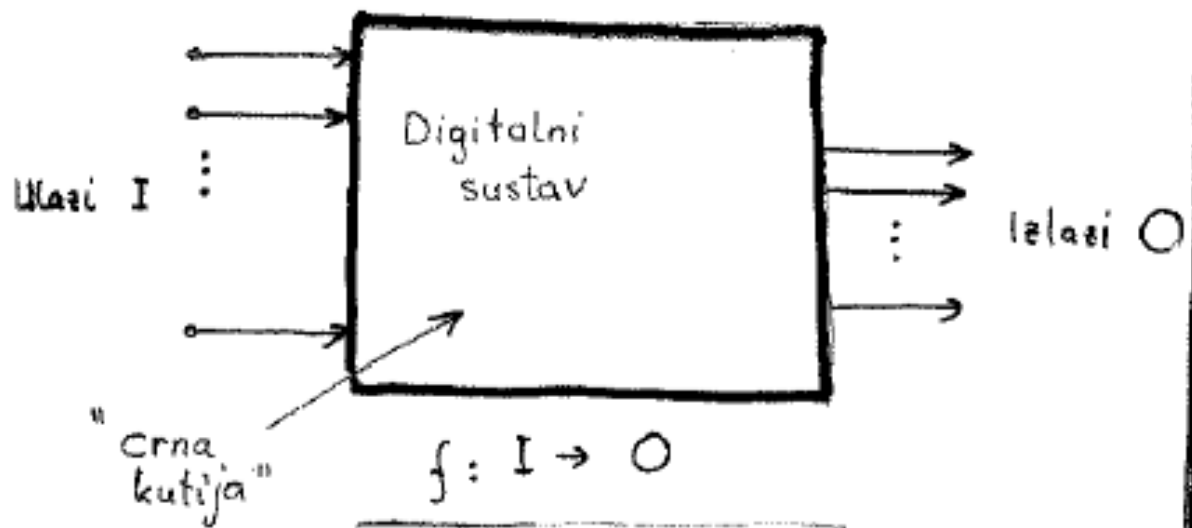


SOC - System-on-Chip



Funkcijski opis

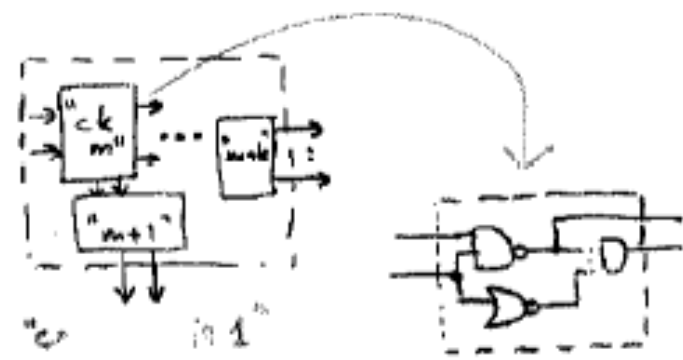
niska



razine apstrakcije



niska



DA (Design Automation) alati temeljeni su na sklopovskim jezicima (HDL - Hardware Description Language(s))i dopuštaju:

- analizu sklopova
- transformaciju HDL ulaza u HDL koji sadrži više sklopovske informacije

MODELIRANJE DIGITALNOG SUSTAVA S MOTRIŠTA DIZAJNERA SKLOPOVLJA

- RAZINE APSTRAKCIJE

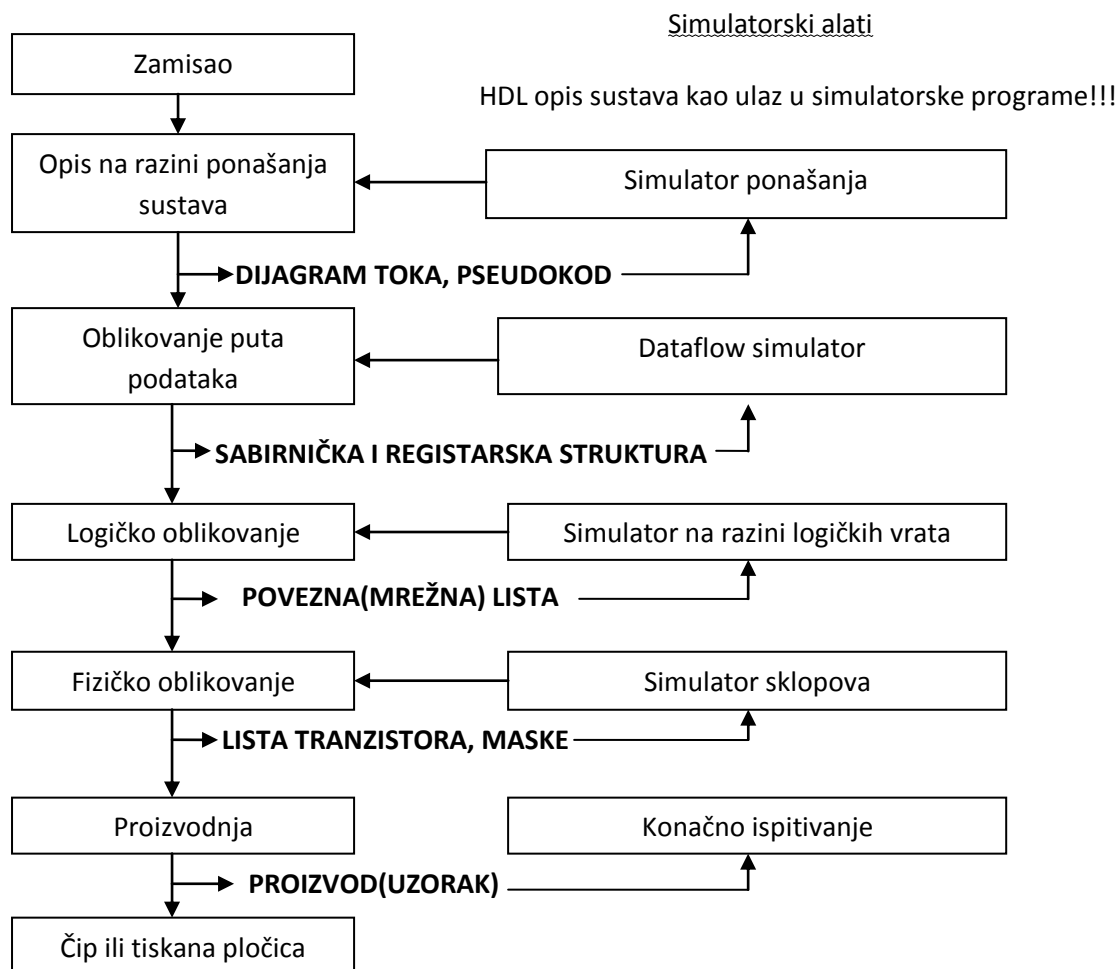
- ako se model rabi za dokumentiranje funkcionalnosti sklopovlja na vrlo visokoj apstraktnoj razini (ponašajni model) -> jednostavni apstraktni model
- ako se model koristi za verifikaciju vremenskih odziva potreban je podrobniji opis -> složeniji apstraktni model

ALATI ZA MODELIRANJE SKLOPOVLJA:

- paper & pencil
- shematski dijagrami
- eksperimentalne pločice
- sklopovski opisni jezici HDL(s)
- posebni jezici za modeliranje (npr. UML - Unified Modeling Language)

NAČELA MODELIRANJA (G. Booch et al. 1999)

1. Izbor modela ima dubok utjecaj na to kako će problem biti napadnut i kakvo će rješenje dobiti
2. Svaki model može se predložiti različitim razinama točnosti
3. Najbolji modeli su povezani sa stvarnosti
4. Jedan model nije dovoljan. Svaki netrivialni sustav se najbolje modelira malim skupom gotovo nezavisnih modela.



SKLOPOVSKI JEZICI (HDL)

HDL

- opis sklopovlja u cilju simulacije, ispitivanja, oblikovanja i dokumentiranja
- za cjelovitu i hijerarhijsku reprezentaciju funkcijskih detalja digitalnog sustava
- mogu prikazivati sklopovlje na različitim razinama apstrakcije
- sadrže simulatore i sintetizatore sklopovlja

Simulacijski programi mogu se koristiti za verifikaciju dizajna dok se sintetizatori rabe za automatsko generiranje sklopovlja.

HDL(s) se upotrebljavaju za opis sklopovlja za simulaciju, modeliranje, ispitivanje, oblikovanje i dokumentiranje

- Jezik za opis ponašanja sklopovlja ISPS (Instruction Set Processor Specification) razvijen u Carnegie Mellon University na temelju ISP notacije (C. G. Bell, 1971.),

ISPS - simulacija sklopovlja

- automatizacija oblikovanja

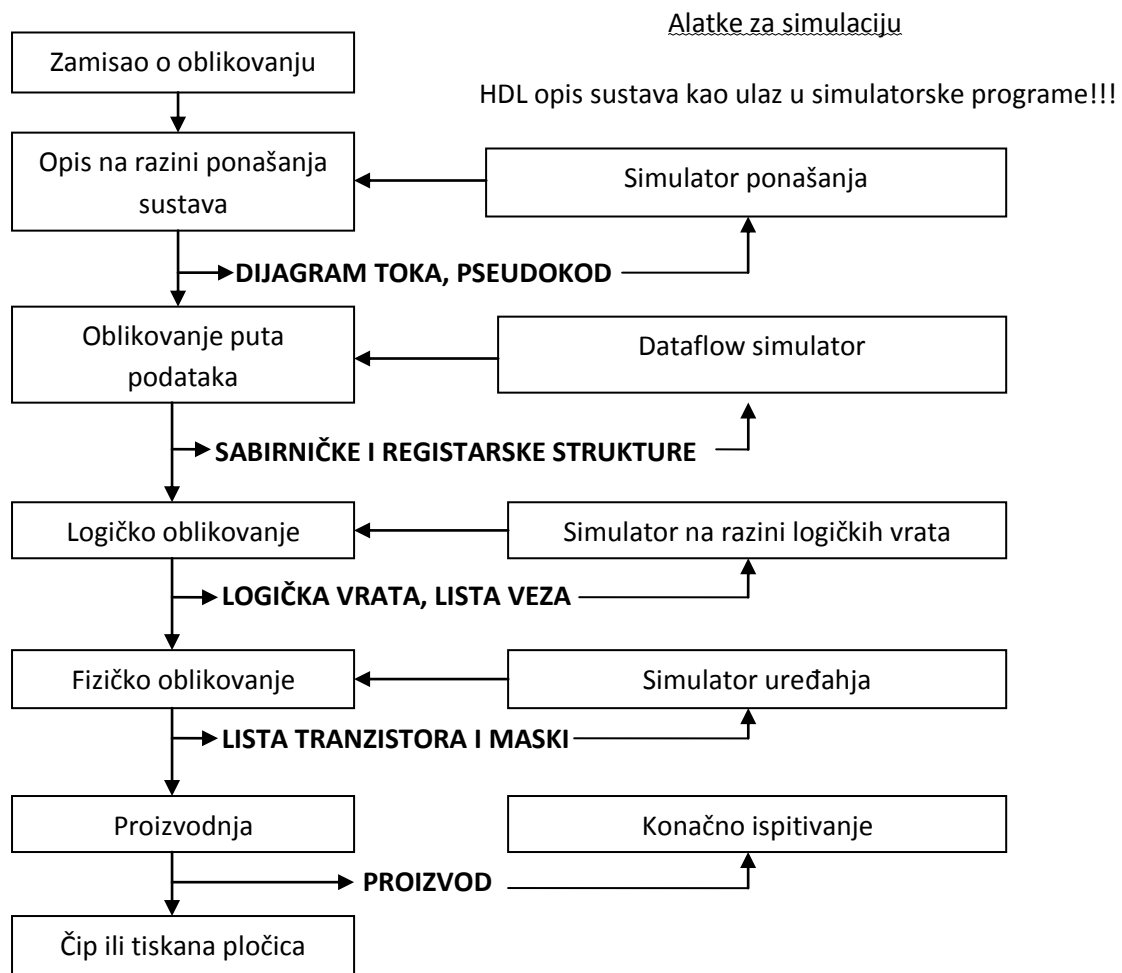
- Jezik za opis toka podataka APLH (A Hardware Programming Language) razvijen u University of Arizona

- Jezik za opis mrežne liste (netlist) - Podskup jezika Verilog

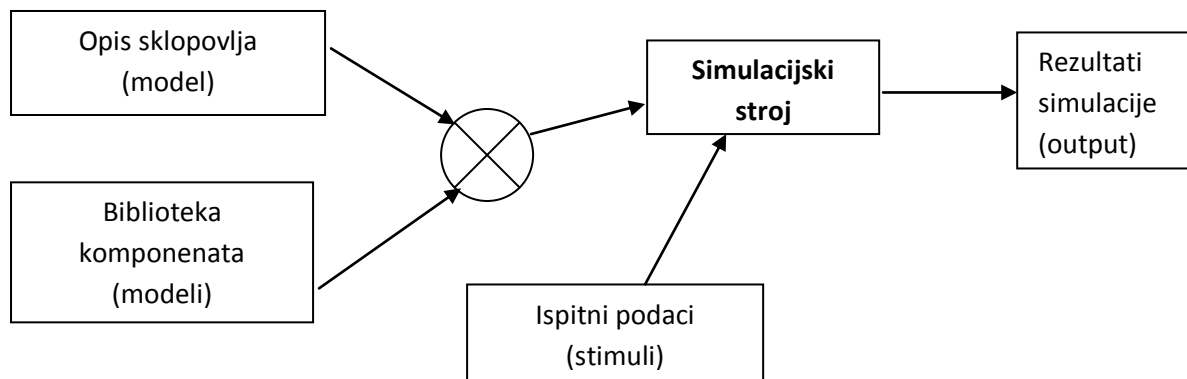
VERILOG

- jezik za opisivanje sklopovlja, počeci 1984. Gateway Design Automatisation
- Verilog -> Motorola, National ASIC
- PLI (Programming Language Interface) – omogućava korisnicima Veriloga proširenje i prilagođavanje okolinama za simulaciju
- Tehnologija sinteze -> tvrtka Synopsys (1987.) temeljena na Verilogu
- Verilog HDL
 - o sintaksa slična programskom jeziku C (lako se uči!)
 - o dopušta prikaz sklopovlja na različitim razinama apstrakcije, te kombinaciju različitih razina apstrakcije u jednom (mješovitom) modelu
- Alatke za sintezu podržavaju Verilog HDL
- Isporučitelji komponenti isporučuju i Verilog HDL biblioteke za postlogičku simulaciju
- **PLI** – bitna komponenta koja dopušta interakciju namjenskog C koda s inst. struk. podataka

Postupak verifikacije tijekom oblikovanja digitalnog sustava uporabom alati za simulaciju



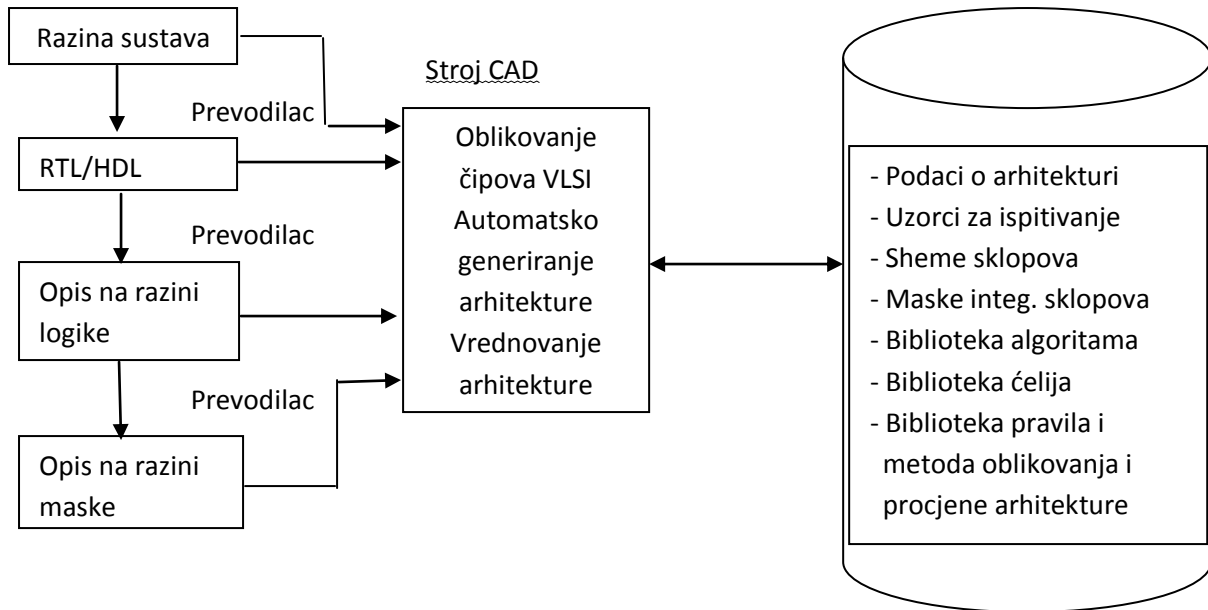
Simulacija sklopovlja



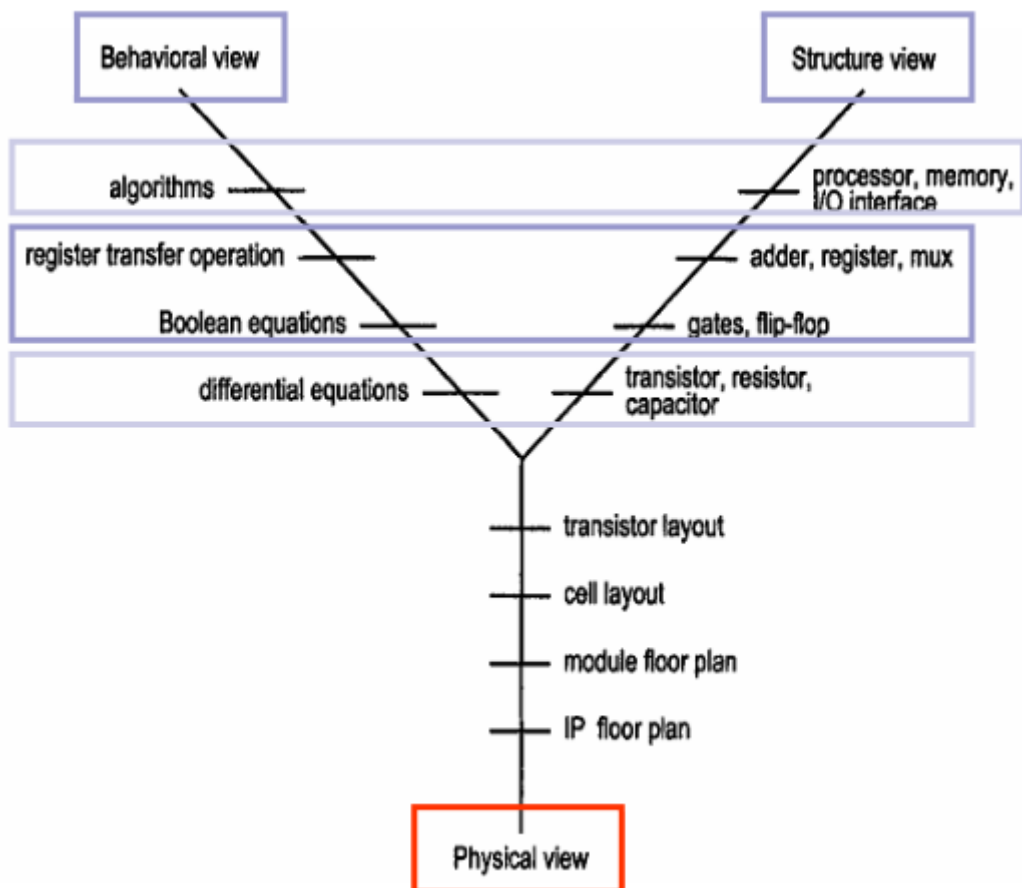
Electronic Design Automation (EDA)

Jezici za opis

Baza podataka CAD



Y-dijagram (D. Gajski)

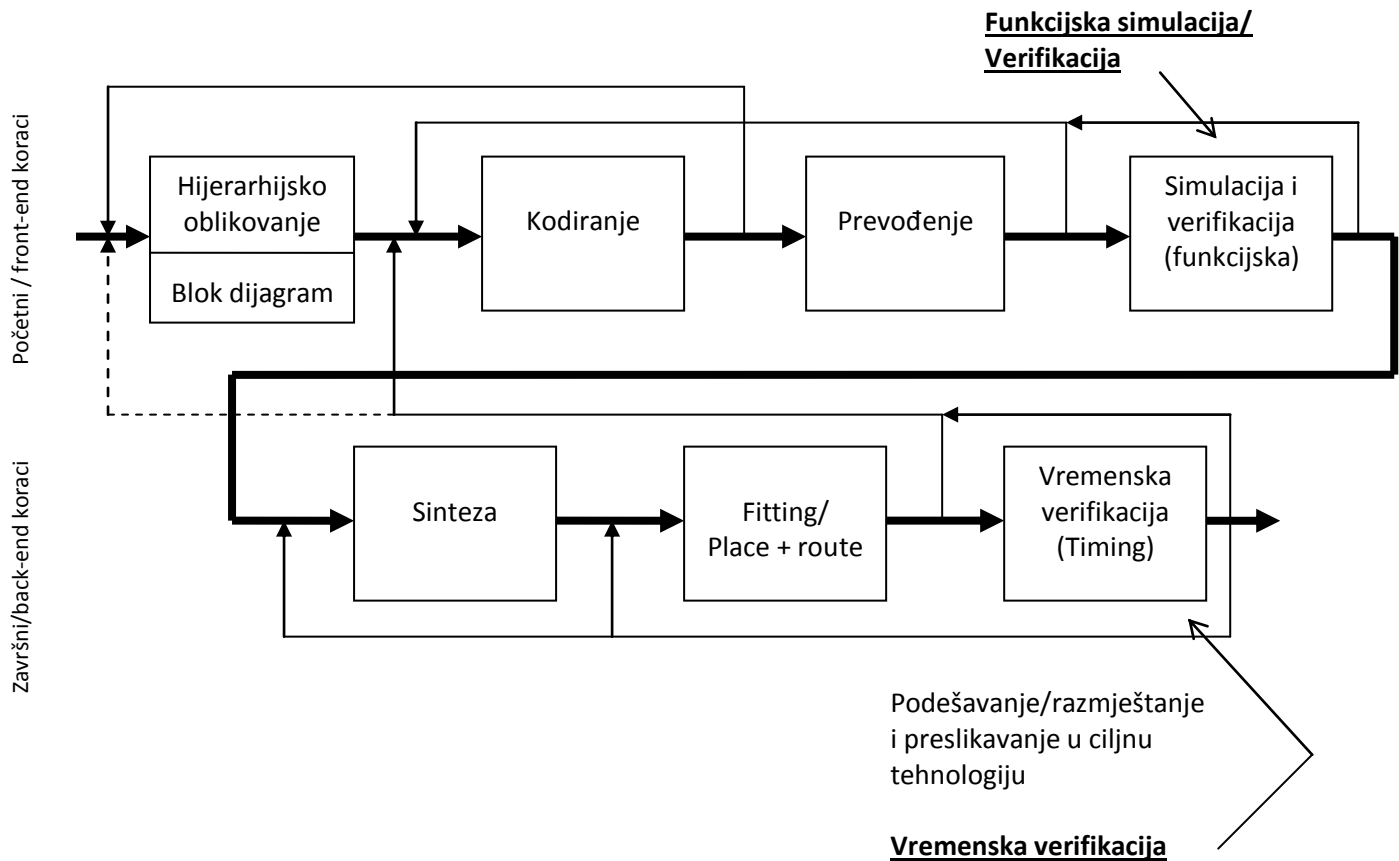


Opis ponašanja – definira izlaze kao funkciju ulaza (ne govori kako se ostvaruje transformacija ulaza u izlaz)

Strukturni opis – ostvarivanje modela ponašanja povezivanjem komponenata sa poznatim ponašanjem

Fizički dizajn – veličina i mjesto na čipu

Tok oblikovanja digitalnog sustava – pristup temeljen na HDL-u



Hijerarhijsko oblikovanje: Složeni logički dizajn -> HDL omogućava definiranje modula i sučelja između njih

Kodiranje: Uporabom tekst editora (specijaliziranih HDL tekst editora)

HDL prevodilac analizira kod (sintaksne pogreške)

Simulacija: Dopuštena uporaba ulaza i promatranje izlaza (bez sklopovske/fizičke izvedbe=

Oblikovanje ispitnih programa (test benches)

Na temelju simuliranih izlaza obavlja se verifikacija – provjera da li se sklop ponaša u skladu s očekivanjem

Funkcijska verifikacija – provjera bez razmatranja vremenskih odnosa

Sinteza: Pretvorba HDL opisa u skup primitiva (komponenti) ciljne tehnologije (npr. PLD, CPLD, AISC...)

Razmještanje komponenti: Preslikavanje sintetiziranih komponenti na raspoložive uređaje, npr. za

PLD ili CPLD preslikavanje na raspoložive AND-OR elemente; ASIC -> smještanje i povezivanje

Vremenska verifikacija: utvrđivanje stvarnog kašnjenja (nastalog zbog duljine veza, tehnoloških značajki sklopovlja...)