

Latvijas Universitāte
Datorikas fakultāte

Ievadlekcija

Kurss "Ievads digitālajā projektēšanā"
Lekcija 07.09.2012

Autors: Rinalds Ruskuls

Sāksim ar aptauju!

- Vārds, uzvārds, e-pasts
- Mana pieredze (kas no manām esošajām zināšanām varētu būt noderīgs, apgūstot šo priekšmetu)
- Kas ir digitālā projektēšana?
- Kas ir kopīgs, un kas - atšķirīgs digitālajai projektēšanai un programmatūras izstrādei?
- Ko Jūs sagaidat no šī priekšmeta?
- Katru atbildi formulēt 2 -3 teikumos
- Aptaujas laiks – 10 minūtes

Nedaudz par mani

- Izglītība
 - Doktorants – Leo Seļāvo vadībā (LU)
 - Maģistra grāds elektronikā (RTU)
 - Bakalura grāds elektronikā (RTU)
- Mana pieredze
 - Digitālo iekārtu izstrāde/projektēšana (FPGA)
 - Principiālo elektrisko iespaidplašu projektēšana
 - Nedaudz programmatūras izstrādes

Kas šajā kursā būs?

- Lekcijas
- Praktiskie darbi
- Mājas darbi
- Vidus semestra kontroldarbs
- Kursa projekti
- Konsultācijas
- Gala eksāmens

Lekcijas

- Piektdienās:
 - 8.30 – 10.10 (bet varam mainīt vietām ar praktiskajiem darbiem)
 - 312. telpā
- Tēmas – atbilstoši kursa aprakstam
- Lekcijās stāstītais var būt, bet var arī nebūt atrodams kursa literatūrā
- Lekcijās apskatītās tēmas pārzināšana atvieglo pārbaudījumu sekmīgu nokārtošanu

Praktiskie darbi

- Piektdienās:
 - 10.30 – 12.10 (bet varam mainīt vietām ar lekcijām)
 - 312. telpā
 - septembris/oktobra sākums
- Darbs ar Xilinx ISE izstrādes vidi
 - Rīka apgūšana
 - Izstrādes plūsmas iepazīšana
 - Sagatavošanās darbam pie kursa projektiem

Mājas darbi

- Izpildes termiņš - 1 nedēļa (šajā piektdienā tiek uzdots, nākošajā jābūt iesniegtam)
- Līdzīgi kā praktiskie darbi septembris/oktobra sākums
- Mājas darbi, ja vien nav norādīts savādāk, tiek iesniegti PDF failu veidā (1 mājas darbs = 1 PDF fails)
- PDF failu nosaukumi veidojami pēc sekojošas struktūras:
 - DIP_MD[mājas darba numurs]_[autora vārds]_[autora uzvārds]_[autora studenta apliecības numurs].PDF
 - PDF failu nosaukumi nesatur specifiskās LV rakstzīmes
- Mājas darbi tiek iesniegti nosūtot uz sekojošu e-pasta adresi:
 - Rinalds punkts ruskuls pie gmail punkts com
- E-pasta lauka SUBJ saturs identisks PDF faila nosaukumam (tikai simbols _ tiek aizstāsts ar simbolu **SPACE**)
- Mājas darba iesniegšana **pēc** norādītā termiņa var **negatīvi** ietekmēt mājas darba vērtējumu

Vidus semestra kontroldarbs

- Kā jau nosaukums rāda notiks ~ semestra vidū (datums tiks izsludināts 2 nedēļas iepriekš)
- Kontroldarbs sastāv no divām daļām:
 - Teorētiskā – jautājumi par tēmām, kas apskatītas lekcijās
 - Praktiskā – pašu projektētas digitālās iekārtas tests
- Nedēļu pēc kontroldarba paredzēta tā rezultātu analīze (pareizās atbildes, tipiskākās kļūdas, utt.)
- Iespēja rakstīt kontroldarbu ārpus paredzēta laika tiek uzskatīta par ekskluzīvu iespēju, tāpēc tai ir nepieciešam būtisks iemesls, piemēram, atrašanās komandējumā ārvlastīs

Kursa projekti

- Pavisam 3:
 - 32 bitu ALU projektēšana (individuāli)
 - Vienkāršas pabeigtas digitālas iekārtas (iespējams – Running Lights) projektēšana (grupās pa 2)
 - Sarežģītākas pabeigtas digitālas iekārtas projektēšana (grupās pa...)
- Izpildes laiks – oktobris/novembris/decembris
- Kamēr nav pabeigts iepriekšējais projekts, nedrīkst sākt darbu pie nākošā
- Darbam pie projektiem izmantojami gan LU, gan citi pieejami datori (Xilinx ISE WebPack)
- Projektu testēšana uz aparatūras tikai 312. telpā

Konsultācijas

- Pasniedzējs
 - Rinalds punkts Ruskuls pie gmail punkts com
- Kurša google group
 - Lu-dip-b pie googlegroups punkts com
- Piektdienās pēc praktiskajiem darbiem 312. telpā (~ 12.10 – 13.00), protams, ja nepieciešams

Gala eksāmens

- Semestra beigās – janvārī
- Saturs zināmā mērā līdzinās vidus semestra kontroldarbam
- Papildus nāk klāt referāts/plakāta prezentācija par kursa projektu

Vērtēšana

Kursa galējais vērtējums sastāv no sekojošām komponentēm:

- 20% - mājas darbi
- 25% - kontroldarbs semestra vidū
- 25% - rakstisks gala eksāmens
- 5% - kursa projekts #1
- 10% - kursa projekts #2
- 15% - kursa projekts #3

Mani ieteikumi

- Droši jautāt, ja kaut kas ir neskaidrs!
- Atvēlēt ~ (6h - 10h) nedēļā brīvā laika šim kursam
- Apspriest savā starpā izklāstītās tēmas
- Strādājot grupās izmantot versiju kontroles rīku, piemēram, Bitbucket?
- Izmantot atvēlēto laiku darbam ar izstrādes rīkiem

Pēc kursa pabeigšanas Jūs spēsiet...

- Izprast digitālo iekārtu projektēšanas gaitu
- Saprast kādi ir digitālo iekārtu pamat-bloki
- Izveidot vienkāršas digitālas iekārtas
 - Digitālo pulksteni
 - PS/2 kontrolieri
 - Alfabētiskā LCD kontrolieri
 - VGA kontrolieri
 - RISC tipa CPU...?

Literatūra

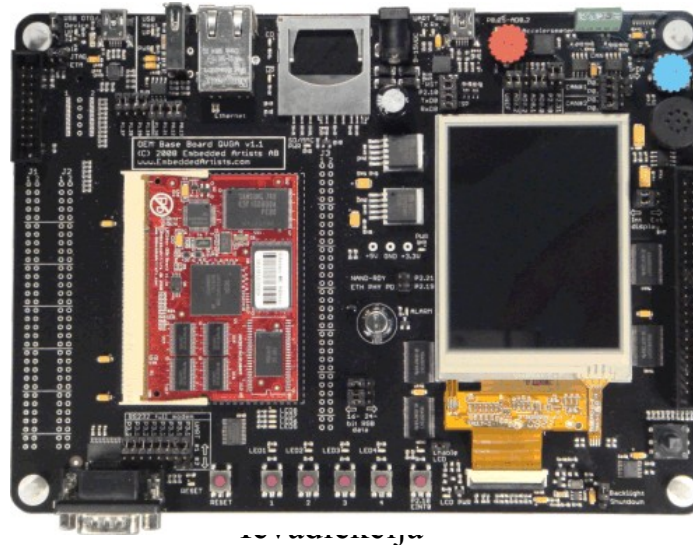
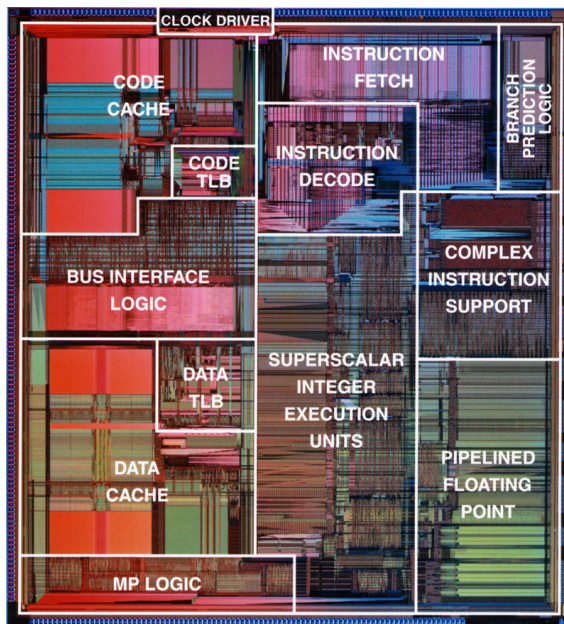
- John L. Hennessy, David A. Patterson, “Computer Organization and Design : The Hardware/Software Interface,” Morgan Kaufmann Publishers, 2005 – 656 pages.
- John L. Hennessy, David A. Patterson, “Computer Architecture, a Quantitative Approach,” Morgan Kaufmann Publishers, 2007 – 704 pages.
- Peter, J.A., “The Designer’s Guide to VHDL,” Morgan Kaufmann Publishers, 1996.
- Pong P. Chu, "RTL Hardware Design Using VHDL: Coding for Efficiency, Portability, and Scalability", Wiley-IEEE Press, 2006 – 694 pages
- Volnei A. Pedroni, "Circuit Design and Simulation with VHDL", The MIT Press, 2010 – 640 pages
- Daļa minēto grāmatu (ne svaigākie izdevumi) pieejama lietošanai uz vietas, 312. telpā

Ievads digitālajā projektēšanā

- **Kas ir digitālā projektēšana?**
- Kāpēc tieši digitālā projektēšana?
- Iekārtu tehnoloģijas
- Sistēmas reprezentācija
- Abstrakcija
- Izstrādes posmi
- Izstrādes gaita



Kas ir digitālā projektēšana?



Case study

Ievads digitālajā projektēšanā

- Kas ir digitālā projektēšana?
- **Kāpēc tiek attīstīta digitālā elektronika?**
- Iekārtu tehnoloģijas
- Sistēmas reprezentācija
- Abstrakcija
- Izstrādes posmi
- Izstrādes gaita

Kāpēc tiek attīstīta digitālā elektronika? (diskusija)

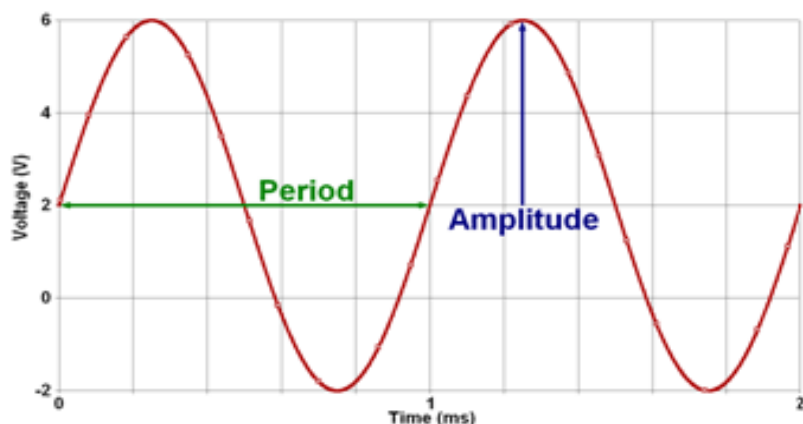
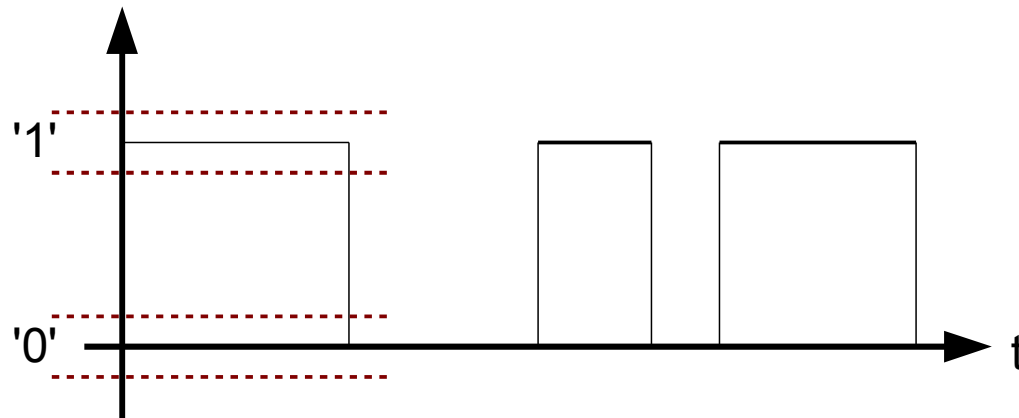


Figure 2

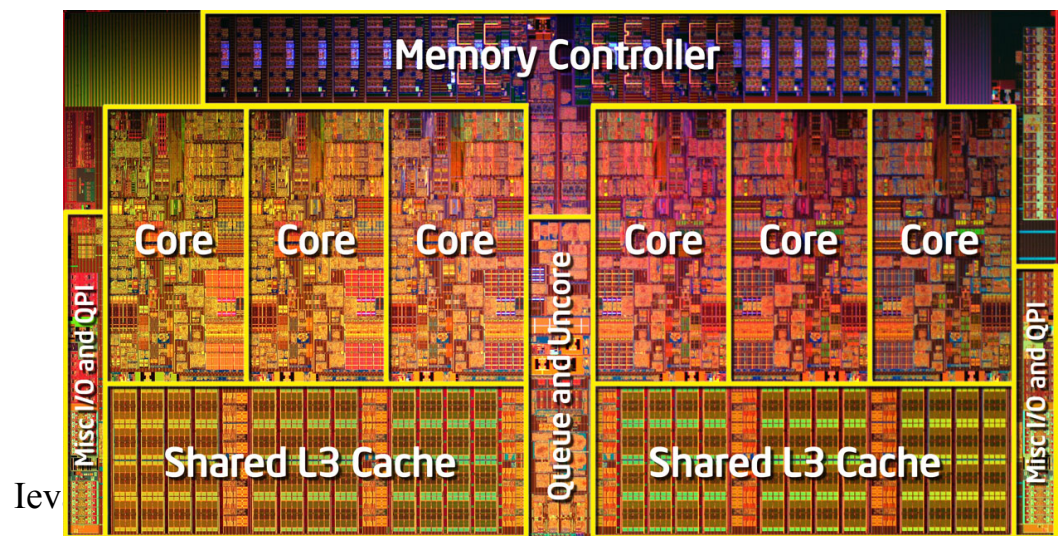
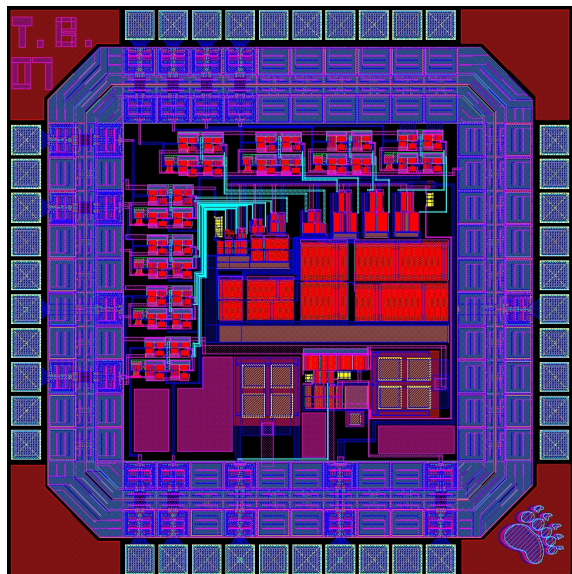


- Priekšrocības

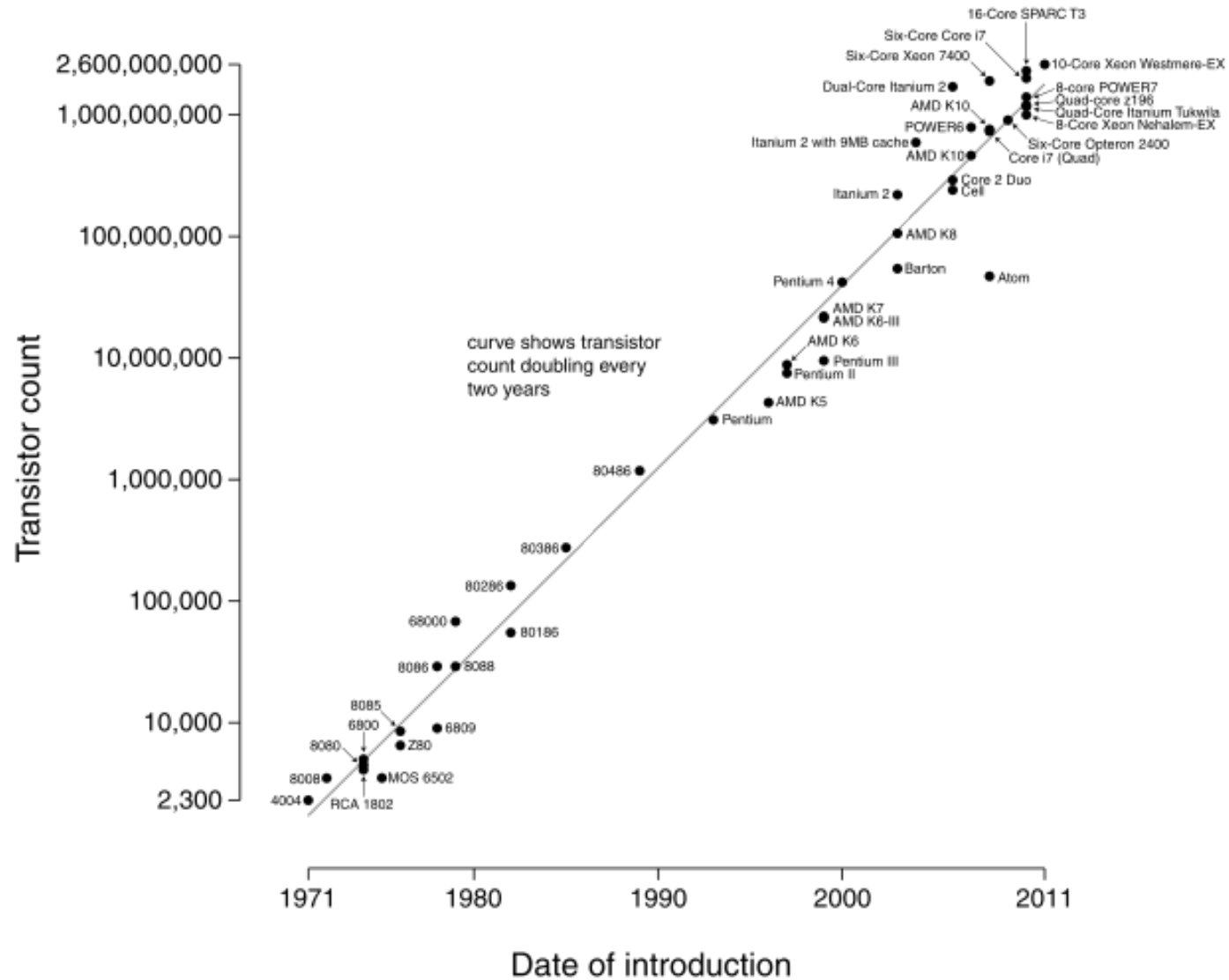
- Trokšņu noturība
- Vieglāk saglabāt, pārsūtīt, manipulēt ar informāciju, nekā analogā veidā
 - Kopēšana bez kļūdām
 - Laika invarianta apstrāde, piemēram, kompresija
- Ekonomija – vieglāk projektēt un ražot digitālas sistēmas

Gordona Mūra likums

- Tranzistoru skaits, ko iespējams uzprojektēt uz viena čipa laukuma vienību dubultojas ik 18 mēnešus
- Iekārtas kļūst vēl mazākas, ātrākas, lētākas un komplicētākas
- Mūsdienās ir iespējams uzprojektēt čipus, kas sastāv no simtiem vai tūkstošiem miljonu tranzistoru



Microprocessor Transistor Counts 1971-2011 & Moore's Law



Digitālās elektronikas iekārtas

- "Digitalizācijas" rezultātā radās dažāda veida iekārtas – datori, telekomunikācijas sistēmas, kontroles iekārtas u.t.t.
- Tās aizvieto daudzas analogās ķēdes:
 - Audio atskaņošana – no kasetes uz CD līdz MP3 atskaņotājiem



Digitālās elektronikas iekārtas

- "Digitalizācijas" rezultātā radās dažāda veida iekārtas – datori, telekomunikācijas sistēmas, kontroles iekārtas u.t.t.
- Tās aizvieto daudzas analogās ķēdes:
 - Attēlu iegūšana – sudraba halogenīdu filmas aizvietoja ar digitālām fotokamerām



Ievadlekcija



Vai digitālā elektronika pilnībā var aizstāt analogo elektroniku?

Pamatmetodes kā izveidot digitālās sistēmas

1. Izmantojot procesorus ar programmatūru (ASM, c++ u.t.t.)

- Plaša pielietojuma procesori, piemēram, Intel Pentium procesorus vai arī mikrokontrolierus (TI, Atmel, ARM, PIC)
- Speciāla pielietojuma procesorus, piemēram, DSP – specializēti ātrai reizināšanai un saskaitīšanai, grafiskie procesori

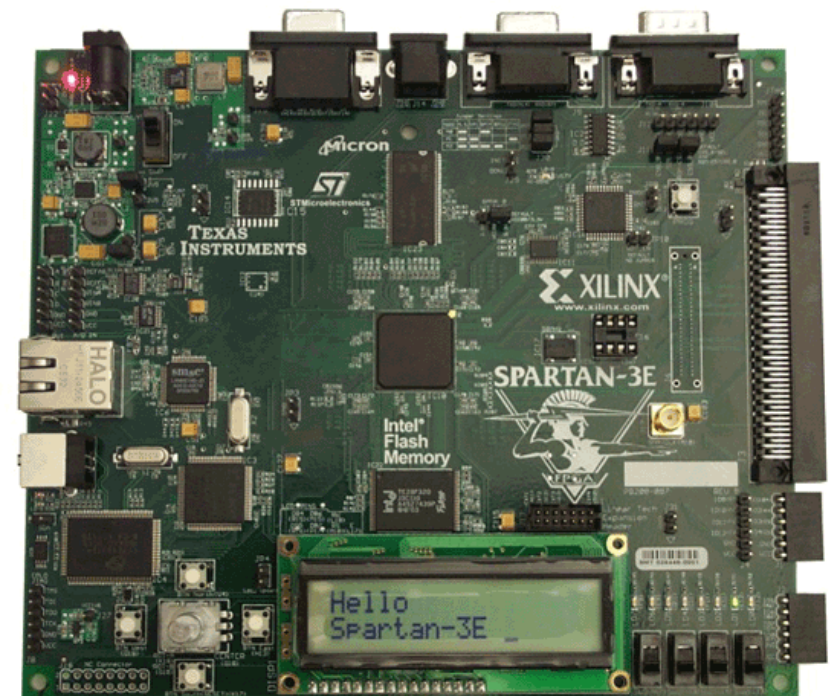
2. Veidojot specifiskas digitālās shēmas

3. Apvienojot izveidoto aparatūru ar programmēto procesoru vai mikrokontrolieri

- Nav viena universāla risinājuma, katram vajag savu pieeju, savus parametrus, ...
- Jāizvēlas starp iekārtas funkcionalitāti, sarežģītību, pielietojumu, jaudas patēriņu



Ievadlekcija

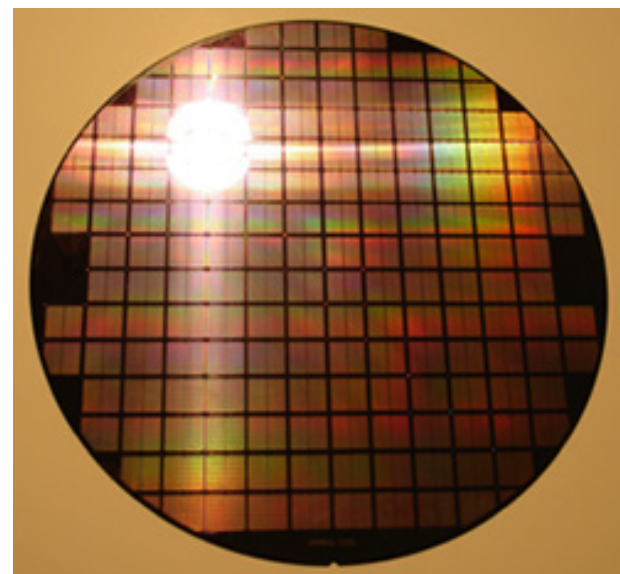
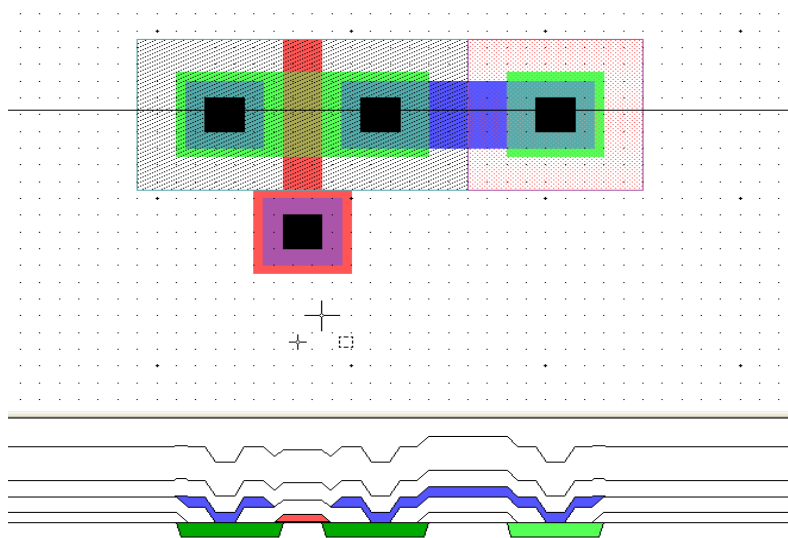


Ievads digitālajā projektēšanā

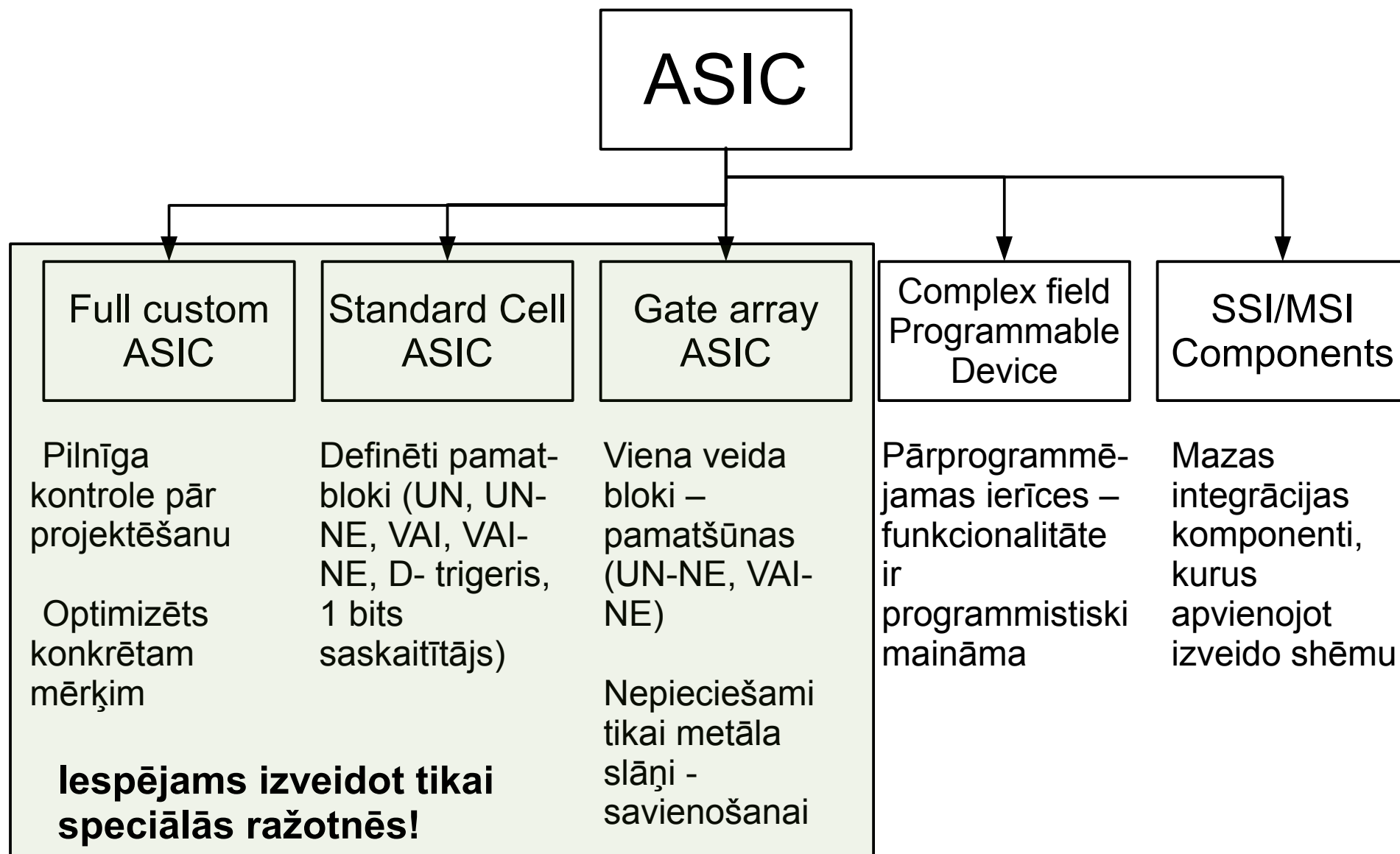
- Kas ir digitālā projektēšana?
- Kāpēc tiek attīstīta digitālā elektronika?
- **Iekārtu tehnoloģijas**
- Sistēmas reprezentācija
- Abstrakcija
- Izstrādes posmi
- Izstrādes gaita

Kas ir IC, kā to veido?

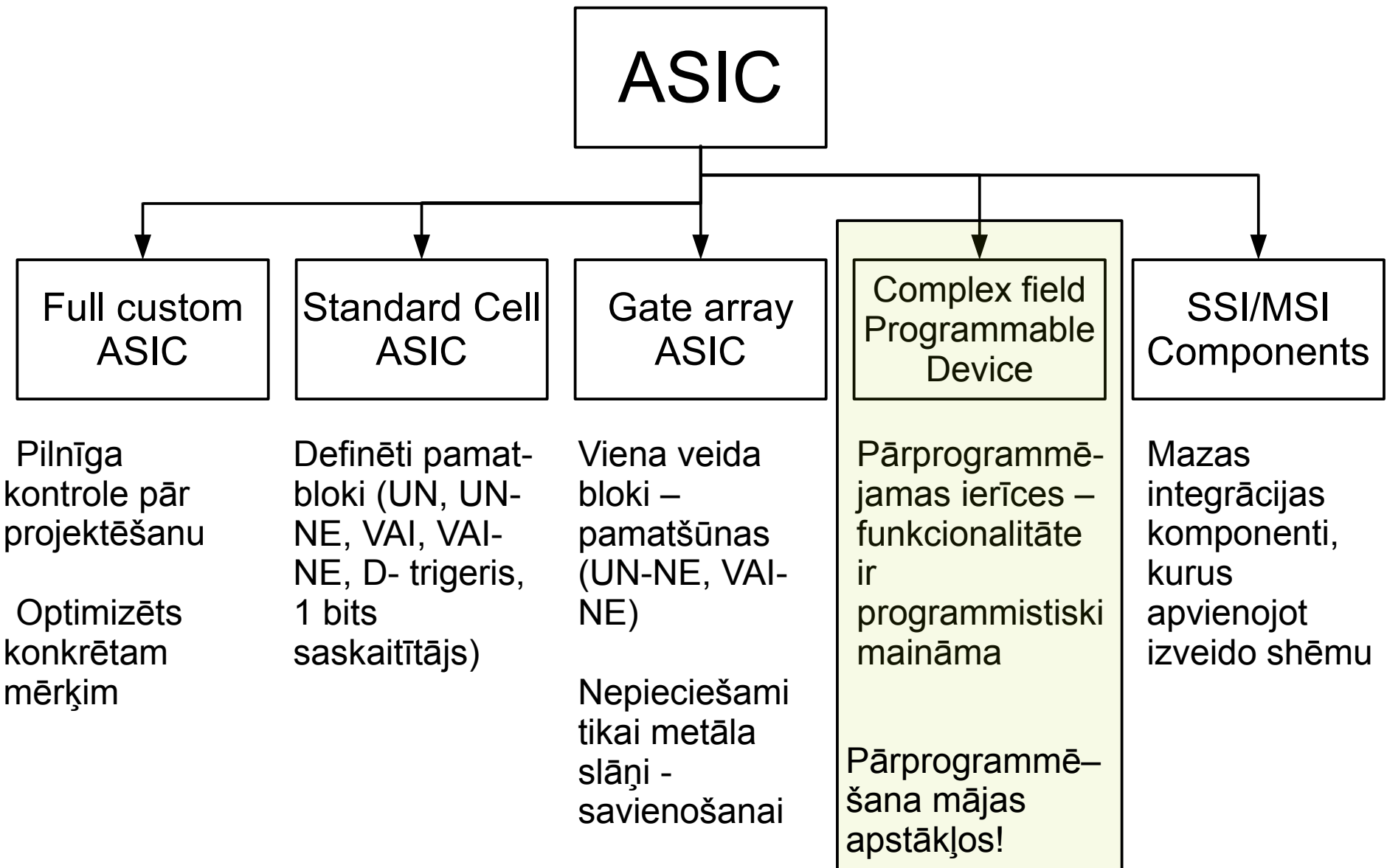
- IC – integrated circuit (integrālā mikroshēma)
 - Sastāv no vairākiem slāņiem (tipiski no 10 – 15)
 - Katram slānim savs izklājums
 - Tehnoloģijas izmēru (um vai nm) nosaka mazākais tranzistora izmērs, ko ar doto tehnoloģiju var uzprojektēt



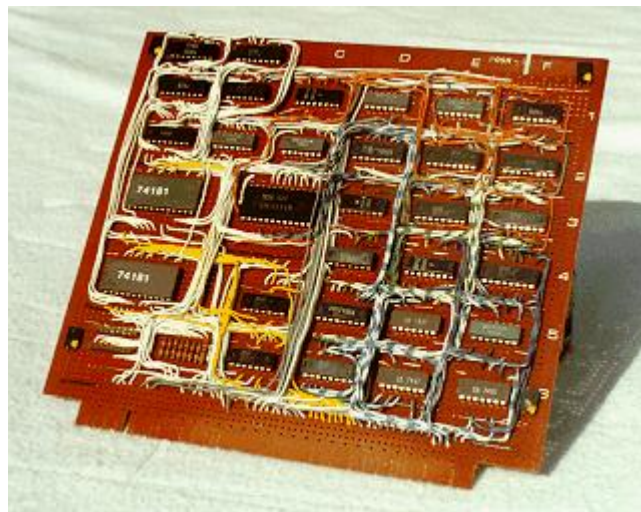
Tehnoloģiju iedalījums



Tehnoloģiju iedalījums



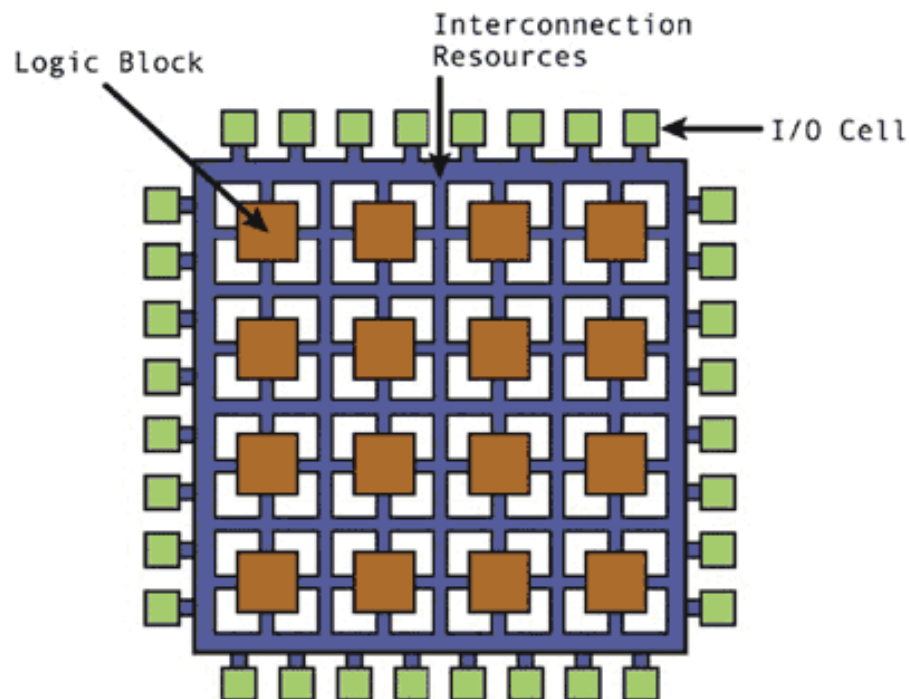
Izlikšanās par citu!



Mēs izmantosim - FPGA

FPGA – **F**ield **P**rogrammable **G**ate **A**rray

- Programmējama loģikas iekārta
- Izmanto digitālo iekārtu prototipēšanai (CPU, MCU...)



Ievads digitālajā projektēšanā

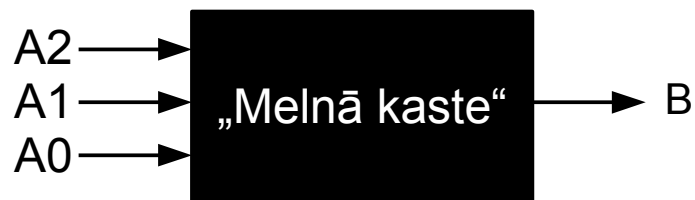
- Kas ir digitālā projektēšana?
- Kāpēc tiek attīstīta digitālā elektronika?
- Iekārtu tehnoloģijas
- **Sistēmas reprezentācija**
- Abstrakcija
- Izstrādes posmi
- Izstrādes gaita

Uzvedības modelis

Blokshēmu modelis

Fizikālais modelis

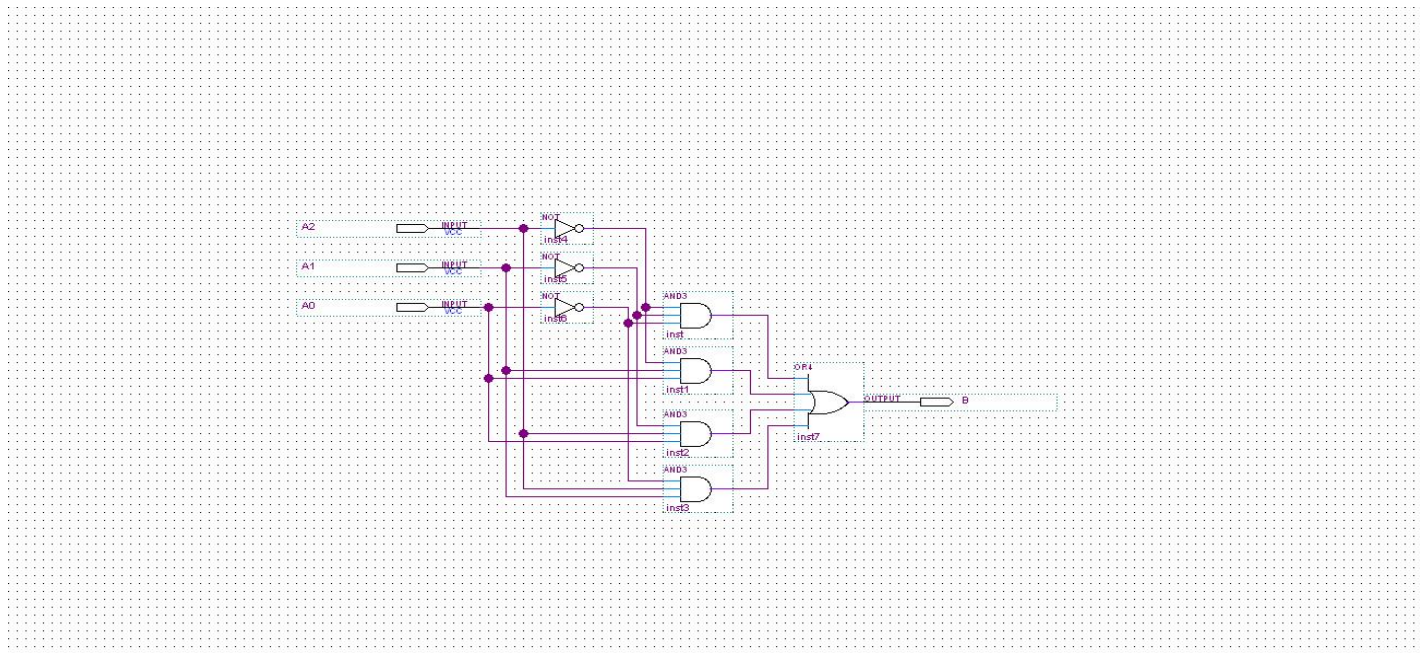
Definē sistēmas funkcionalitāti, ieejas/izejas signālu vērtības. Sistēmu uzskata par „melno kasti” neiedziļinoties kā funkcionalitāte(s) tiek panākta(s).



A2	A1	A0	B
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Blokshēmu modelis

Definē sistēmu blokshēmu līmenī, to savstarpējo interkonekciju utt., piemēram, principālālā elektriskā shēma.

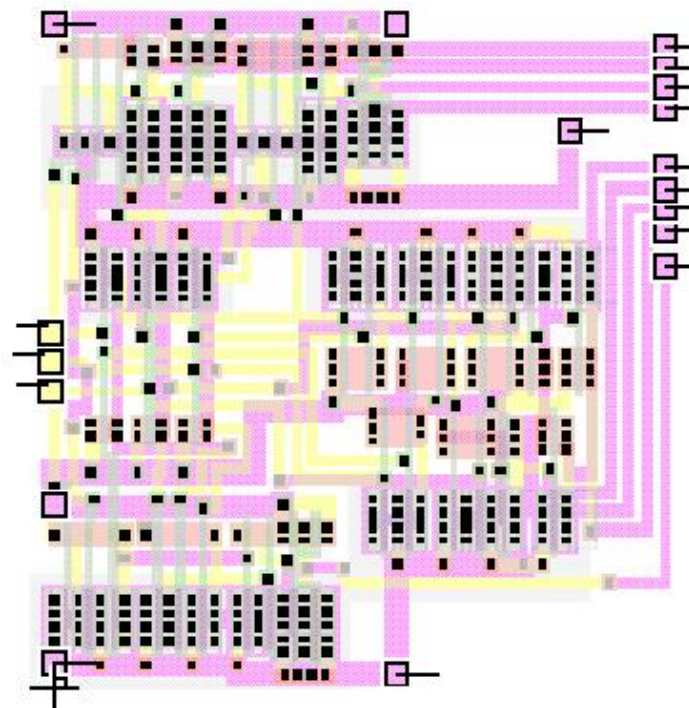
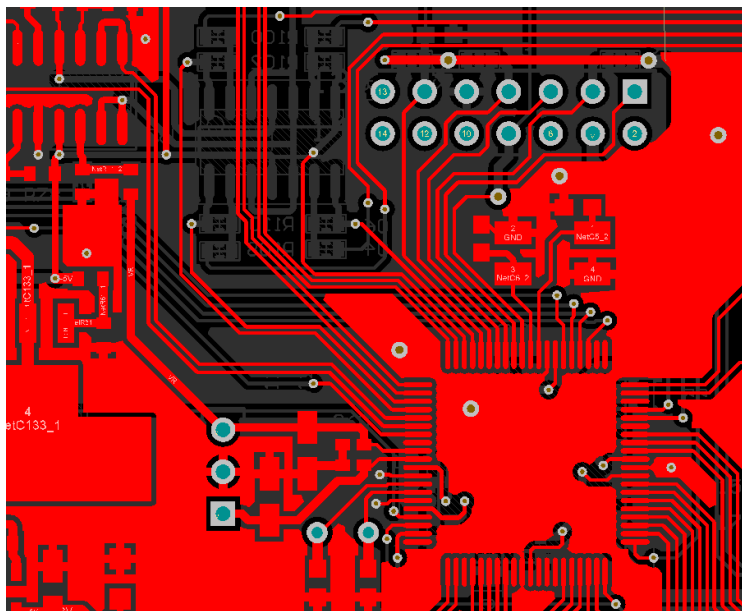


Uzvedības modelis

Blokshēmu modelis

Fizikālais modelis

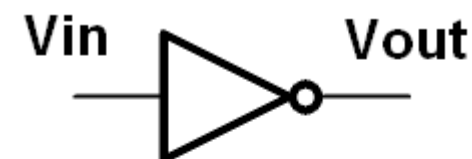
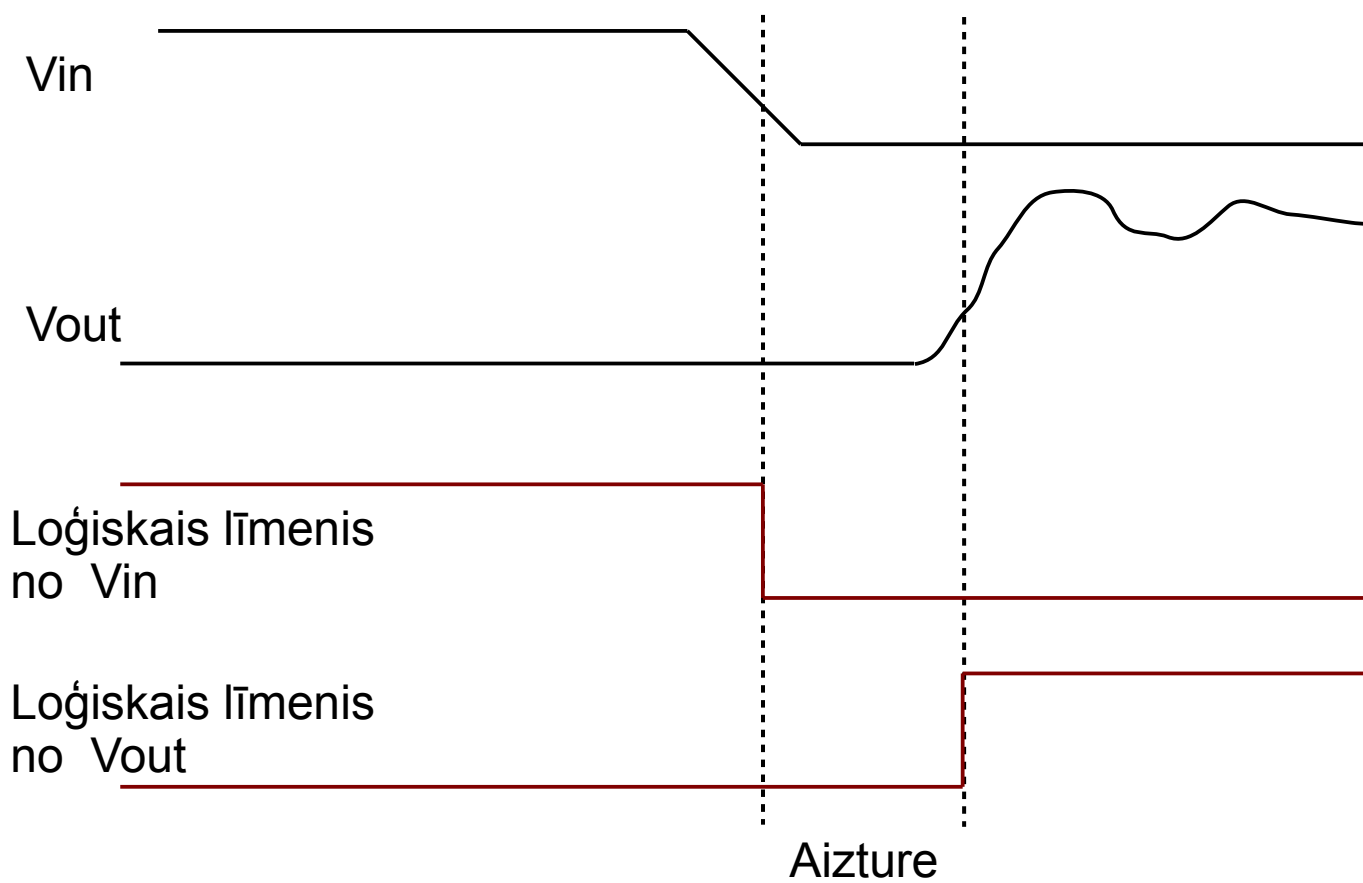
Implementējamās sistēmas izmēri, savienojumi, komponentu ģeometriskie izmēri
(piemēram, iespiedplate (PCB), čipa izklājums (IC layout))



Ievads digitālajā projektēšanā

- Kas ir digitālā projektēšana?
- Kāpēc tiek attīstīta digitālā elektronika?
- Iekārtu tehnoloģijas
- Sistēmas reprezentācija
- **Abstrakcija**
- Izstrādes posmi
- Izstrādes gaita

Vienkāršota sistēmas aplūkošana



Vin	Vout
0	1
1	0

Abstrakciju līmeņi digitālajā elektronikā

	Pamatelementi	Signāla reprezentācija	Laika Reprezentācija	Uzvedība
Tranzistors	Tranzistori, Rezistori	Sprieguma un strāvas vērtības	Nepārtraukta funkcija (analog signāls)	Deferenciāl- vienādojumi
Loģiskie Pamatelementi	Invertors, UN, UN-NE, VAI, VAI-NE...	'0' vai '1'	Loģisko elementu aiztures	Būla algebra
Kompleksie Pamatelementi	Saskaitītāji, Multiplekseri, Trigeri	Stāvoklis	Pulkstenis (clock)	Stāvokļu diagrammas, Pārejas
Procesors	Procesora atmiņa	Abstrakts datu tips	Notikumu secība	Algoritms ASM, C, C++, Java

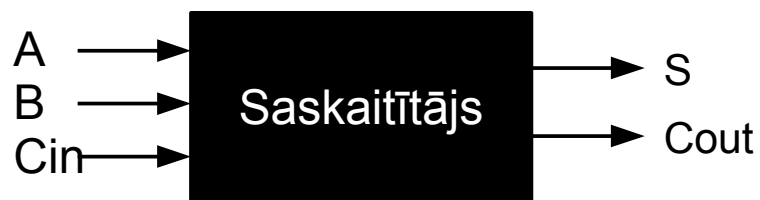
Ievads digitālajā projektēšanā

- Kas ir digitālā projektēšana?
- Kāpēc tiek attīstīta digitālā elektronika?
- Iekārtu tehnoloģijas
- Sistēmas reprezentācija
- Abstrakcija
- **Izstrādes posmi**
- Izstrādes gaita

Izstrādes gaita

- **Uzdevuma (problēmas) aprakstīšana**
- Iekārtas fizikālā projektēšana
- Testēšana

Definē uzvedības un/vai blokshēmu modeļus.



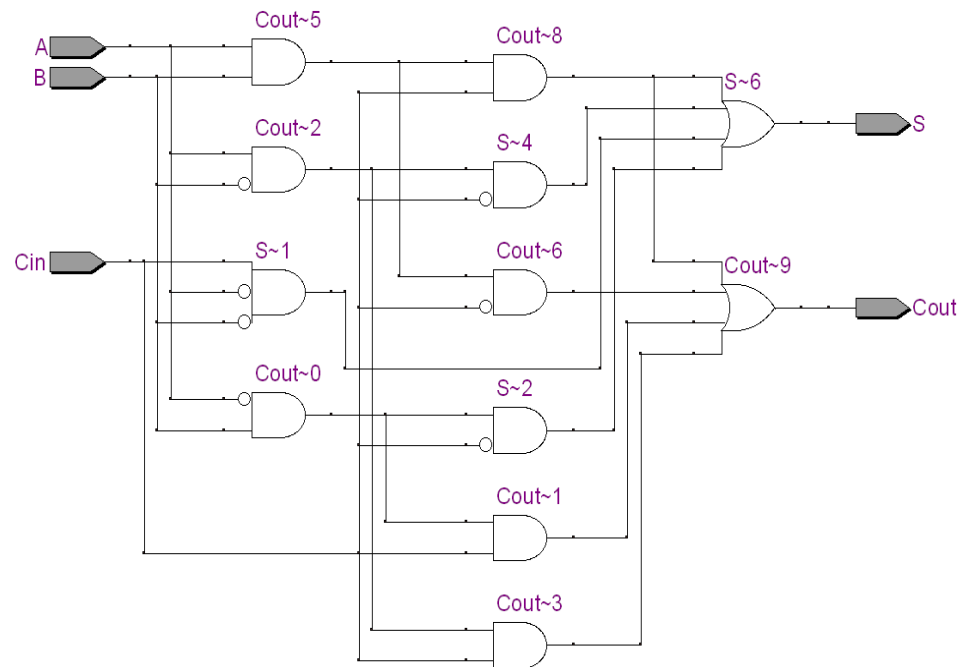
A	B	Cin	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Izstrādes gaita

- **Uzdevuma (problēmas) aprakstīšana**
- Iekārtas fizikālā projektēšana
- Testēšana

Definē uzvedības un/vai blokshēmu modeļus.

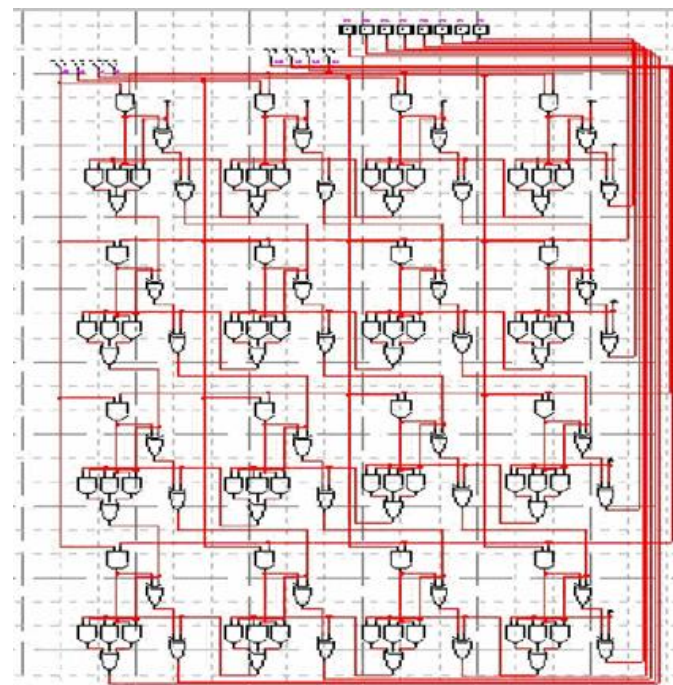
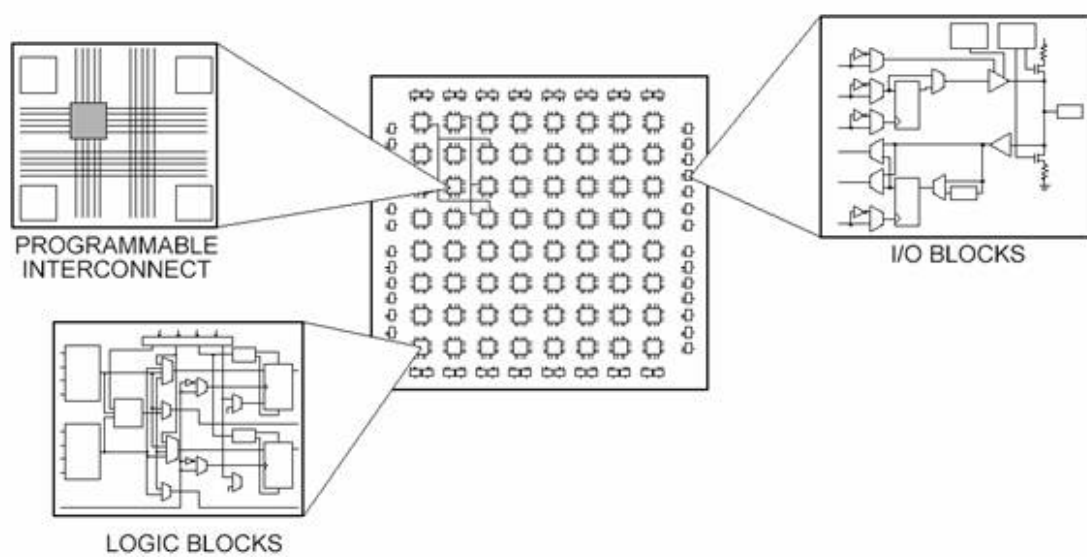
```
-- Sum
S <= '1' when (A='0' and B = '0' and Cin='1') or
               (A='0' and B = '1' and Cin='0') or
               (A='1' and B = '0' and Cin='0') or
               (A='1' and B = '1' and Cin='1')
           else '0';
-- Carry out
Cout <= '1' when (A='0' and B = '1' and Cin='1') or
                 (A='1' and B = '0' and Cin='1') or
                 (A='1' and B = '1' and Cin='0') or
                 (A='1' and B = '1' and Cin='1')
           else '0';
```



Izstrādes gaita

- Uzdevuma (problēmas) aprakstīšana
- Iekārtas fizikālā projektēšana
- Testēšana

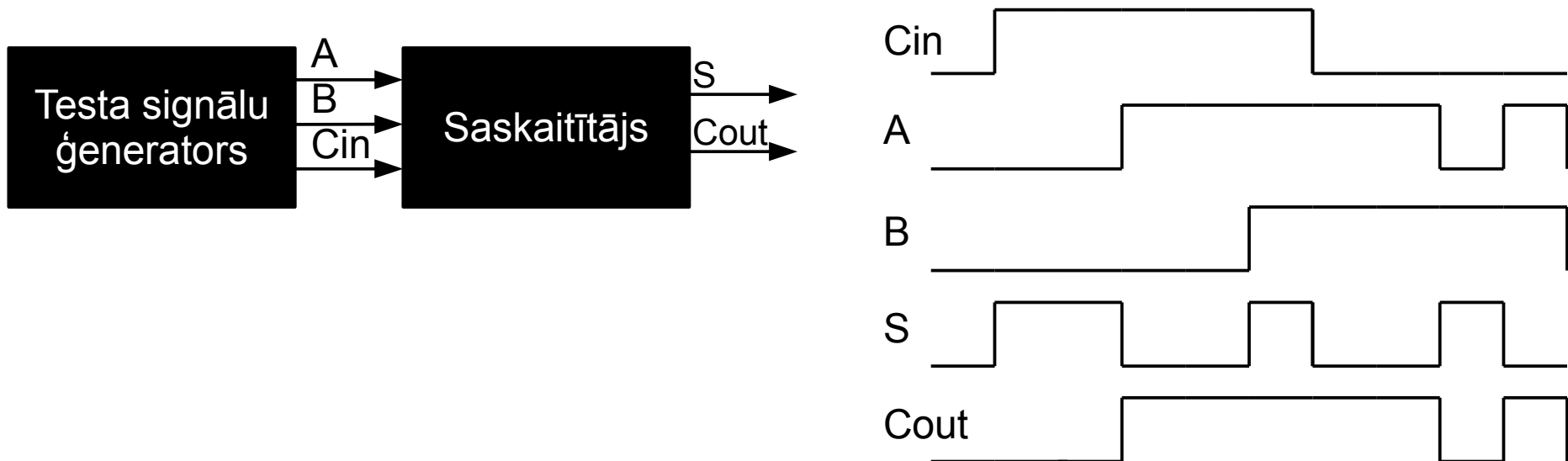
FPGA – loģisko elementu interkonekciju veic programmatūra.



Izstrādes gaita

- Uzdevuma (problēmas) aprakstīšana
- Iekārtas fizikālā projektēšana
- **Testēšana – formālā**

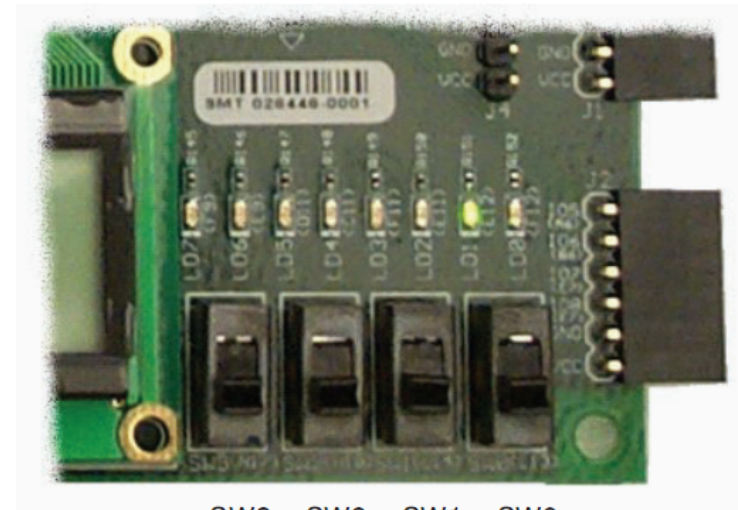
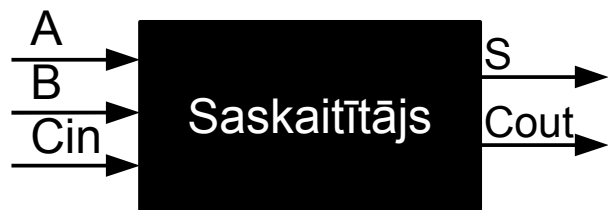
Definē testa signālus, uzdodot grafiski vai programmas kodā.



Izstrādes gaita

- Uzdevuma (problēmas) aprakstīšana
- Iekārtas fizikālā projektēšana
- **Testēšana – fizikālā**

Piesaista izveidoto shēmu fiziskai ierīcei, definējot ieejas un izejas.



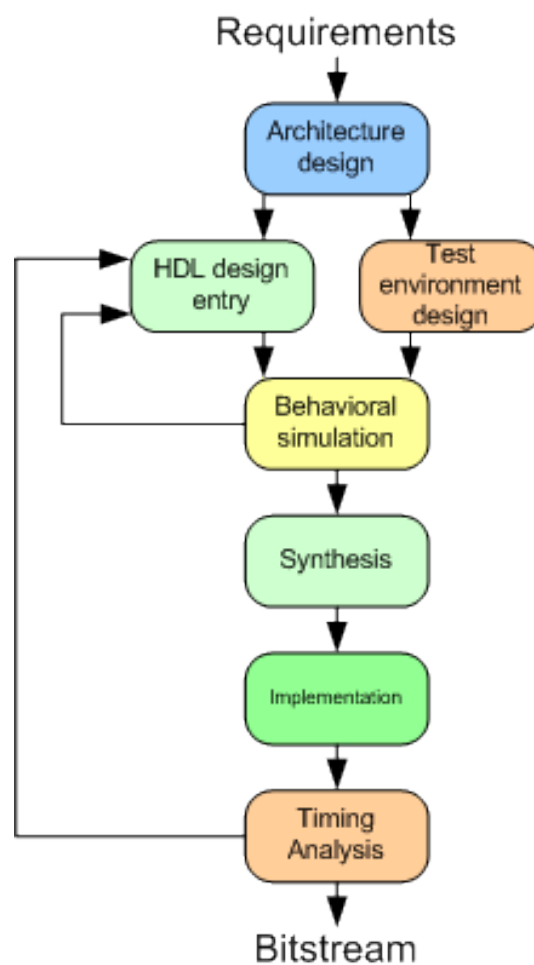
SW3 SW2 SW1 SW0
(N17) (H18) (L14) (L13)

HIGH
↑
LOW

UG230_c2_01_021206

Figure 2-1: Four Slide Switches

Izstrādes gaita (algoritms)



Programmatūras ierobežojumi

- EDA (Electronic Design Automation)
- Programmatūra nespēj automātiski pārvērst nepilnīgi aprakstītu dizainu – pabeigtā.
- Tā var veikt tikai lokālas optimizācijas

Paldies par uzmanību!
Jautājumi?