

Latvijas Universitāte
Datorikas fakultāte

Kombinatoriskā ciparu elektronika

Kurss "ievads digitālajā projektēšanā"
Lekcija 14.09.2012

Autors: Rinalds Ruskuls

Lekcijas saturs

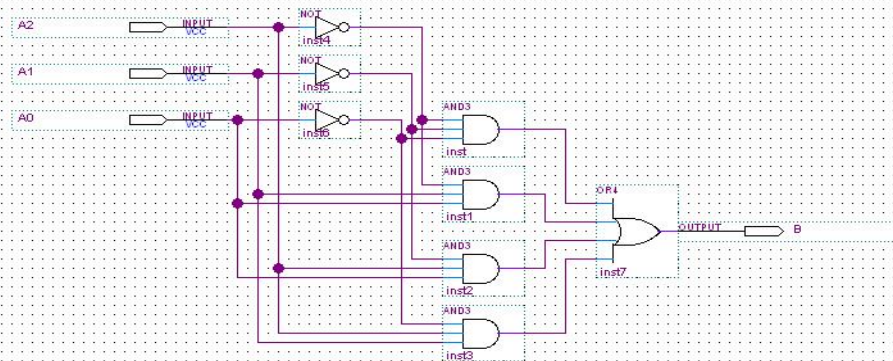
- **Ievads ciparu elektronikā**
- No kā veido ciparu elektronikas pamatelementus?
- Kādā veidā veido ciparu pamatelementus?
- Karno kartes
- Skaitļošanas sistēmas

Ciparu elektronika

- Divas apakšgrupas
 - Kombinatoriskā ciparu elektronika
 - Secīgā ciparu elektronika

Kas ir kombinatoriskā ciparu elektronika?

- Sastāv no loģiskiem elementiem
- Loģisko elementu kombinācijas
- Izejas signāls ir atkarīgs no ieejas signāliem dotajā laika momentā



Lekcijas saturs

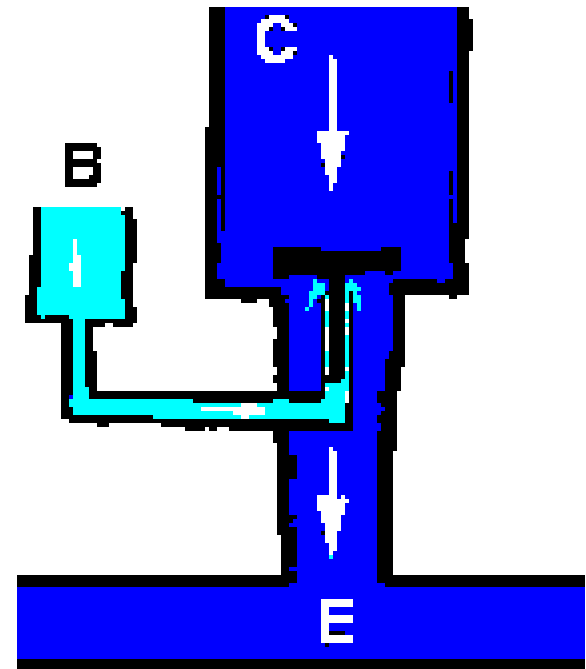
- Kombinatoriskā ciparu elektronika
- **No kā veido ciparu elektronikas pamatelementus?**
- Kādā veidā veido ciparu pamatelementus?
- Karno kartes
- Skaitļošanas sistēmas

Tranzistori

- Iedalās
 - BJT – bipolārie tranzistori (NPN, PNP)
 - FET – lauka efekta tranzistori (N, P kanāla)
- Iespējami 3 darbības režīmi
 - Nogriešanas
 - Lineārais
 - Piesātināšanas

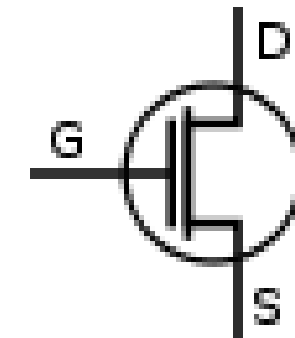
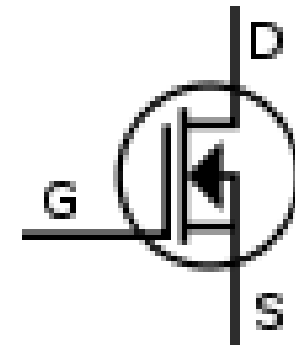
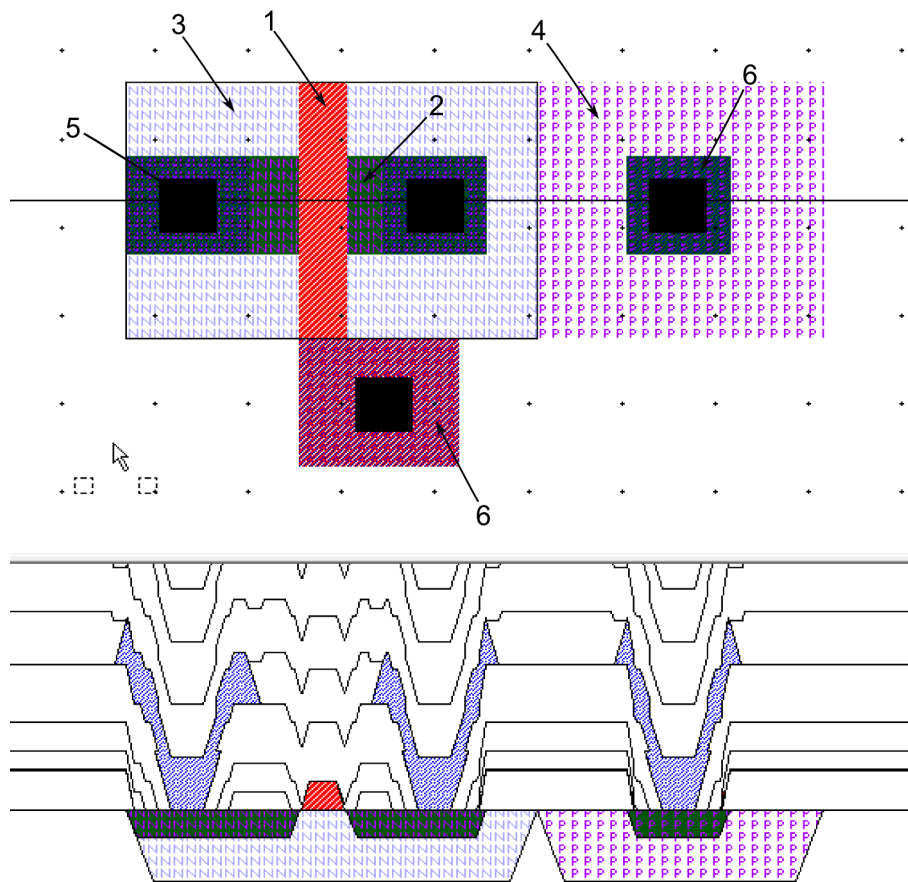
Vienkāršots tranzistoru modelis

- Palielinot ūdens daudzumu bāzes apgabalā, lielāka daļa ūdens sāk tecēt no kolektora uz emiteru.
- Ar mazāku enerģiju, var kontrolēt lielākus enerģijas apjomus



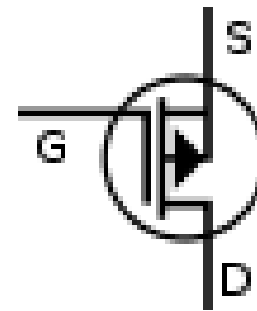
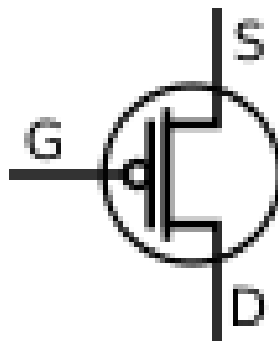
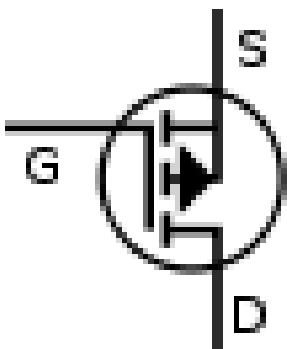
Lauktranzistori (CMOS) N-kanāls

- Pielieto ciparu elektronikā – CMOS tehnoloģija!
- CMOS – **C**omplementary **M**etal **O**xide **S**emiconductor



Lauktranzistori (CMOS) P-kanāls

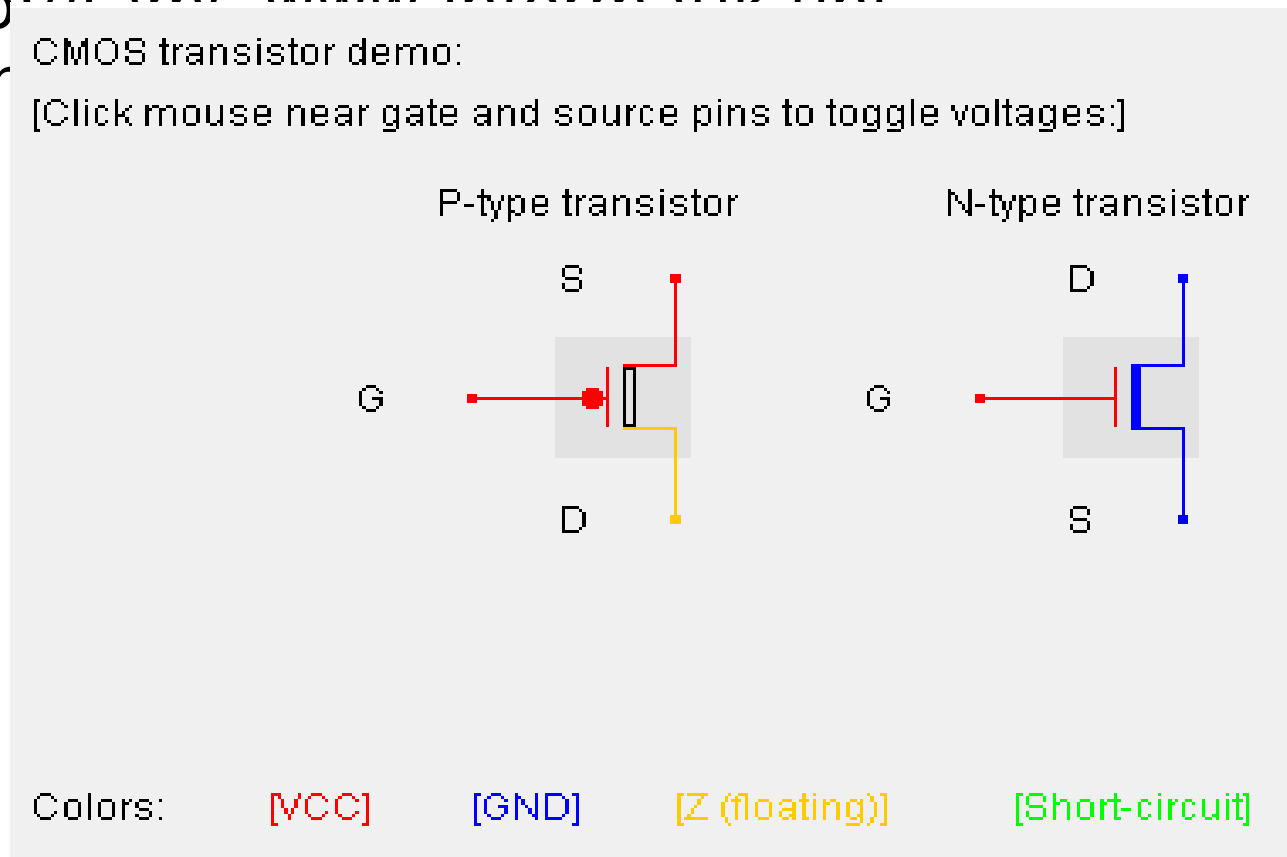
- Pielieto ciparu elektronikā – CMOS tehnoloģija!
- CMOS – **C**omplementary **M**etal **O**xide **S**emiconductor
- Ja U_{gs} spriegums ir negatīvs – tranzistors ir atvērts



Tranzistori

- Iespēja virtuāli patestēt tranzistorus un to slēgumus:

- http://toms-www.informatik.uni-hannover.de/~toms/teaching/2004/01/01/01_cmos_demo.html

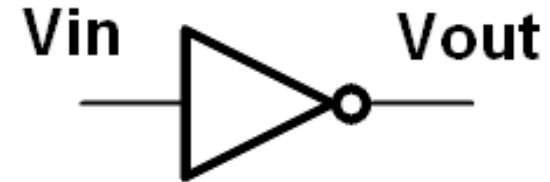


Lekcijas saturs

- Kombinatoriskā ciparu elektronika
- No kā veido ciparu elektronikas pamatelementus?
- **Kādā veidā veido ciparu pamatelementus?**
- Ciparu elektronikas pamatelementi
- Karno kartes
- Skaitļošanas sistēmas

Invertors

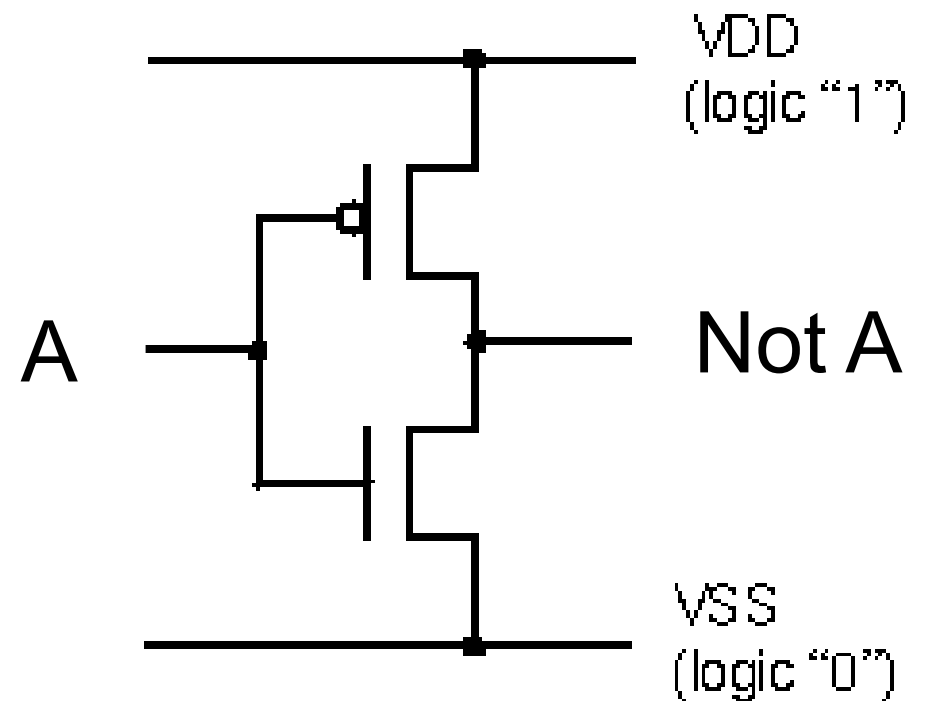
- Vienkāršākais ciparu elektronikas pamatelements
- Invertē ieejas signālu
- $A = \tilde{A}$



Vin	Vout
0	1
1	0

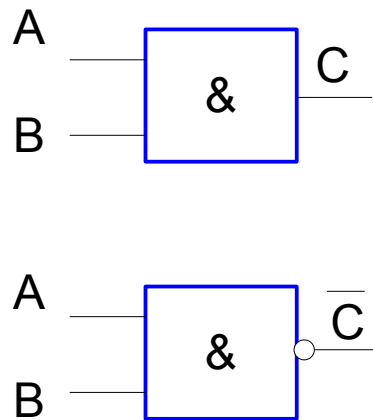
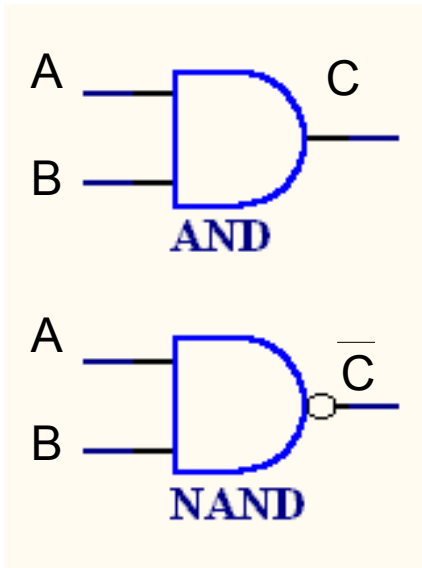
Invertors no tranzistoriem

- VDD – barošanas spriegums
- Augšējais tranzistors atvērs, tad kad $A = V_{ss}$ (zeme)
- Apakšējais tranzistors atvērts, ja $A = VDD$



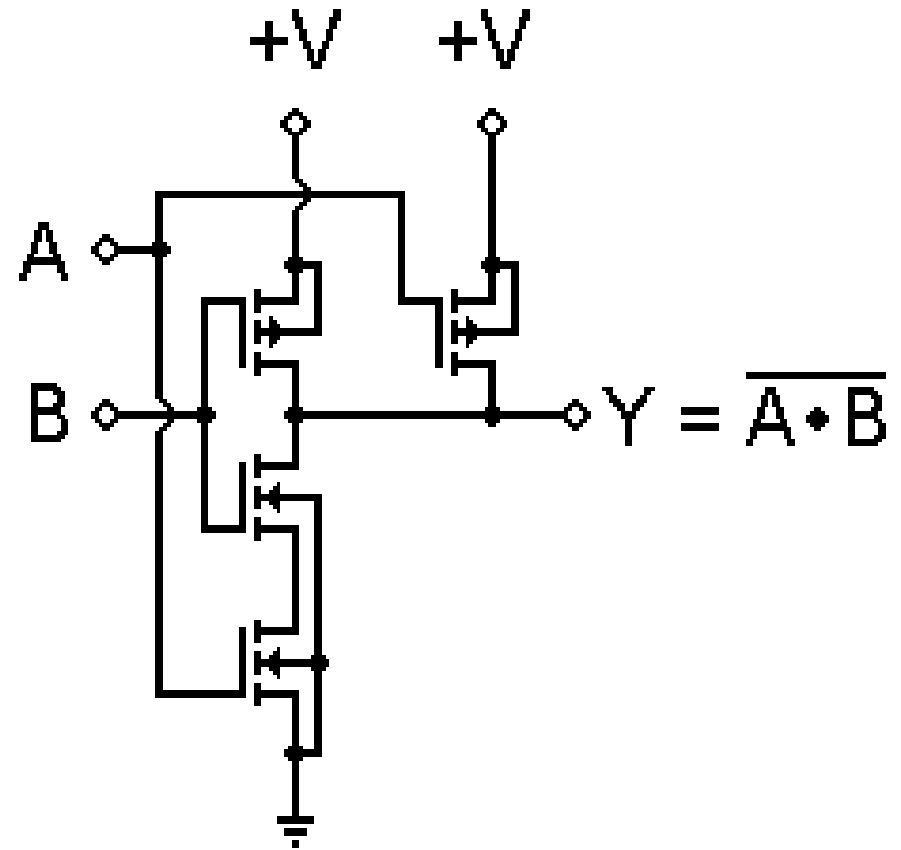
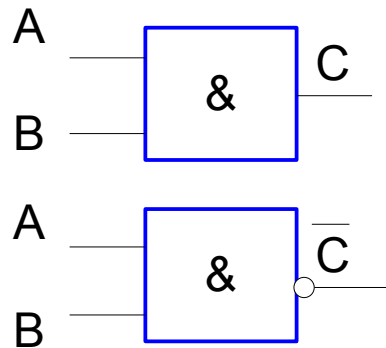
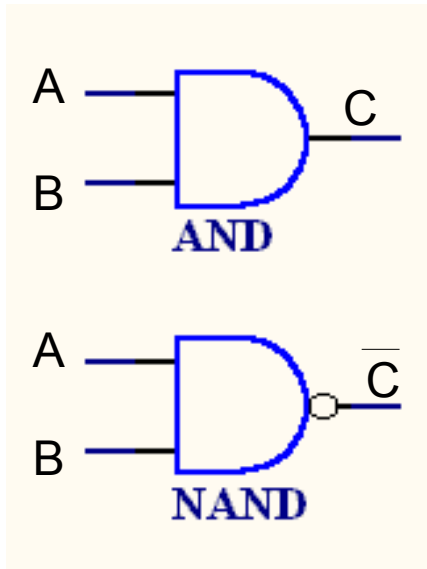
UN, UN-NE (AND, NAND)

$$C = A * B$$



A	B	C	Not C
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

UN, UN-NE (AND, NAND)



A	B	C	Not C
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Kāda izskatītos UN (AND) shēma?
Vai tā būtu sarežģītāka par UN-NE (NAND)?

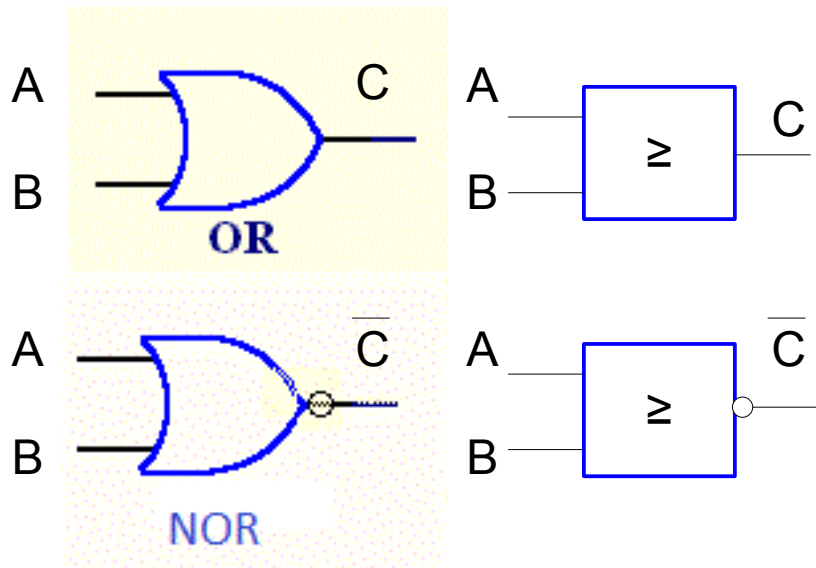
Mājas darbs – pirmā daļa

- Izveidot no CMOS tranzistoriem sekojošus loģiskos elementus
 - NAND, AND
 - NOR, OR
 - XOR

VAI, VAI-NE (OR, NOR)

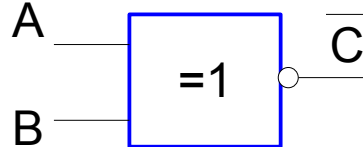
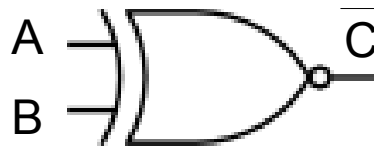
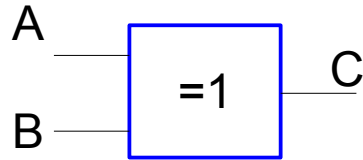
$$C = A + B$$

$$\hat{C} = \overline{A + B}$$



A	B	C	Not C
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Izslēdzošais VAI (XOR)



A	B	C	Not C
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

Mājas darbs – otrā daļa

- Izmantojot loģiskos elementus, NAND, NOR, NOT izveidot elementu XOR (2 varianti)
 - Vadi drīkst krustoties
 - Vadi nekrustojas

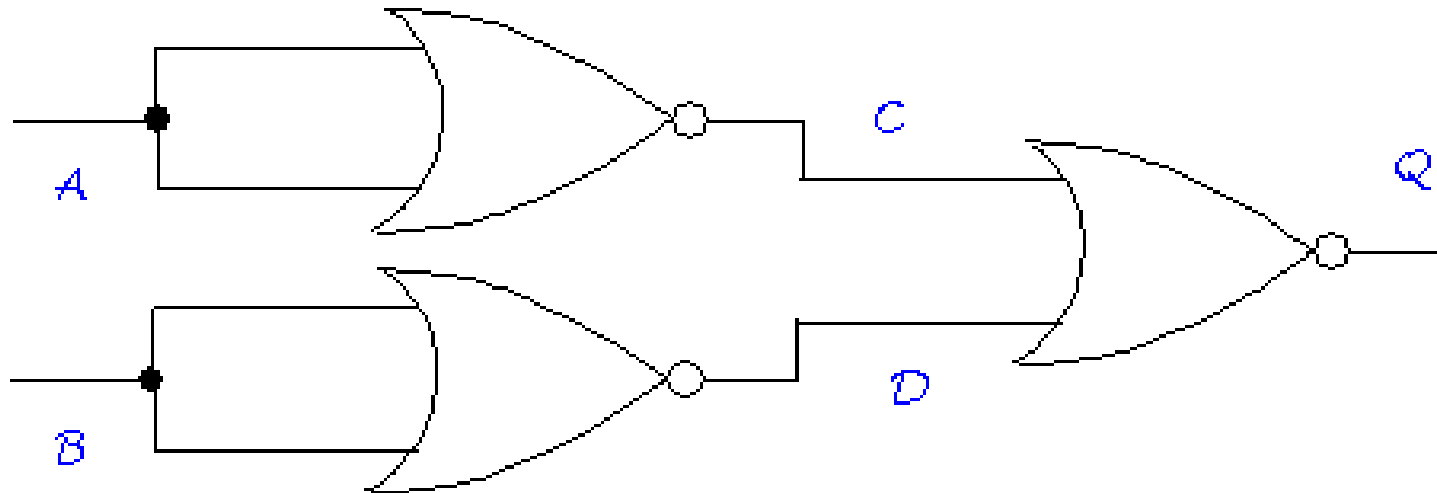
Uzdevums Nr. 1

- Izveidot no VAI-NE (NOR) elementiem UN (AND) loģisko elementu!

De Morgana likums

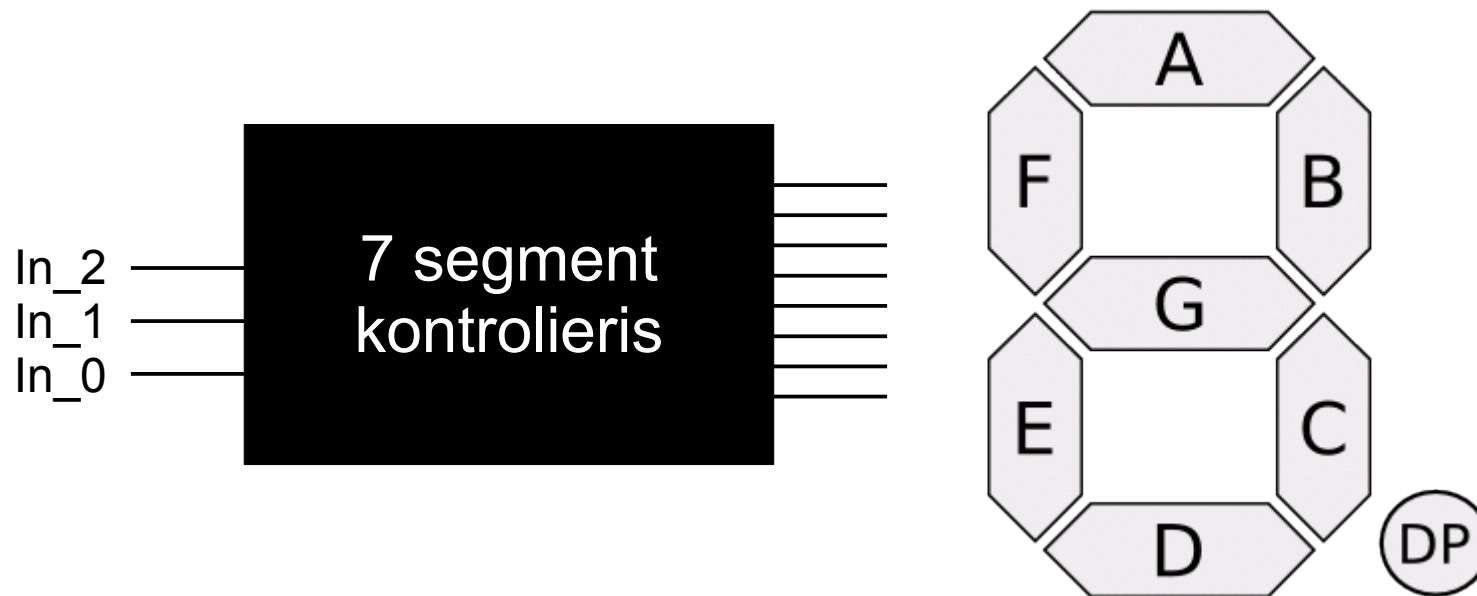
$$A * B = \neg(\neg A + \neg B)$$

$$A + B = \neg(\neg A * \neg B)$$



Ciparu shēmu veidošana un optimizēšana

- Izveidot bin to "seven segment" shēmu.
 - Ieejā – trīs biti
 - Uz seven segment displeja jāattēlo skaitlis



Ciparu shēmu veidošana un optimizēšana

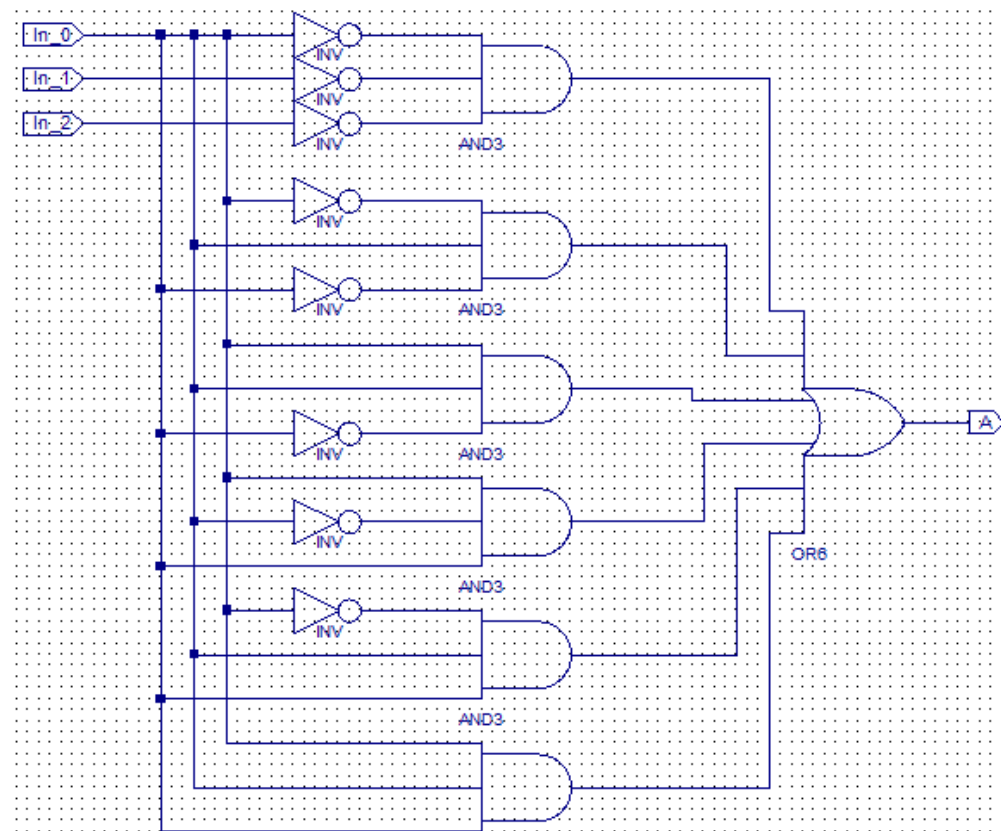
- Sastāda shēmas funkcionalitāti – īstenības tabula

In2	In1	In0	A	B	C	D	E	F	G	Output
0	0	0	1	1	1	1	1	1	0	0
0	0	1	0	1	1	0	0	0	0	1
0	1	0	1	1	0	1	1	0	1	2
0	1	1	1	1	1	1	0	0	1	3
1	0	0	0	1	1	0	0	1	1	4
1	0	1	1	0	1	1	0	1	1	5
1	1	0	1	0	1	1	1	1	1	6
1	1	1	1	1	1	0	0	0	0	7

Ciparu shēmu veidošana un optimizēšana

- Apskata tikai vienu izeju, piemēram, A
- Apskata tikai gadījumus, kad izejā ir '1'

In2	In1	In0	A
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

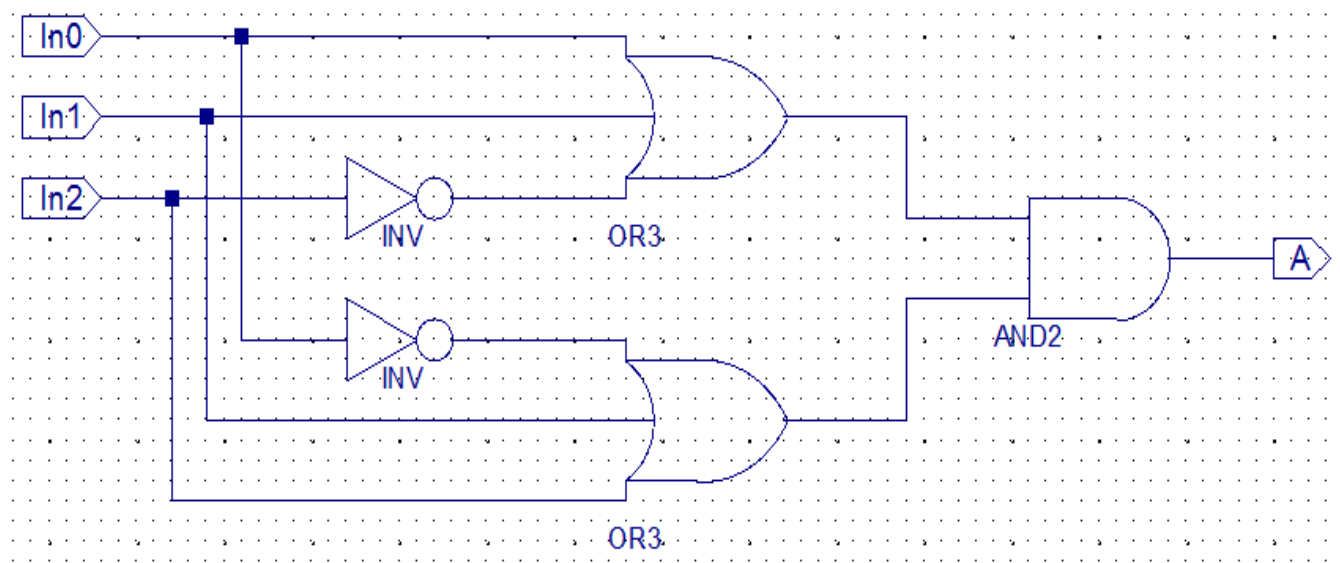


Ciparu shēmu veidošana un optimizēšana

Apskata tikai vienu izeju, piemēram, A

Apskata tikai gadījumus, kad izejā ir '0'

In2	In1	In0	A
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Secinājumi

- Viena no pieejām jāizvēlas pēc vieninieku vai nulļu skaita – kuru mazāk, to jāizvēlas
- Jāsaprot atšķirība starp shēmu veidošanu nulļu un vieninieku gadījumā!

Ciparu shēmu veidošana un optimizēšana – Karno kartes

Ieejas funkcijas uz vienas ass var atšķirties, piemēram, uz x ass AB, bet uz y ass - C.

AB, CD – ieejas funkcijas argumenti

Tiek kārtoti pēc Greja koda

Kvadrāta iekšienē – funkcijas izejas vērtības

Atzīmē vieniniekus 2^n pakāpē

		AB			
		00	01	11	10
CD	00	0	0	1	1
	01	0	0	1	1
	11	0	0	0	1
	10	0	1	1	1

$$f(A,B,C,D) = E(6,8,9,10,11,12,13,14)$$

$$F = AC' + AB' + BCD' + AD'$$

$$F = (A+B)(A+C)(B'+C'+D')(A+D')$$

7 segment optimizācija

- Atzīmē pa grupām vieniniekus
- Skatās, kādi no ieejas signāliem ietekmē izejas signālu
- Shēmas veido ar **UN** elementiem, bet apvieno ar **VAI** elementu

		In1 In2			
		00	01	11	10
In0	0	1	0	1	1
	1	0	1	1	1

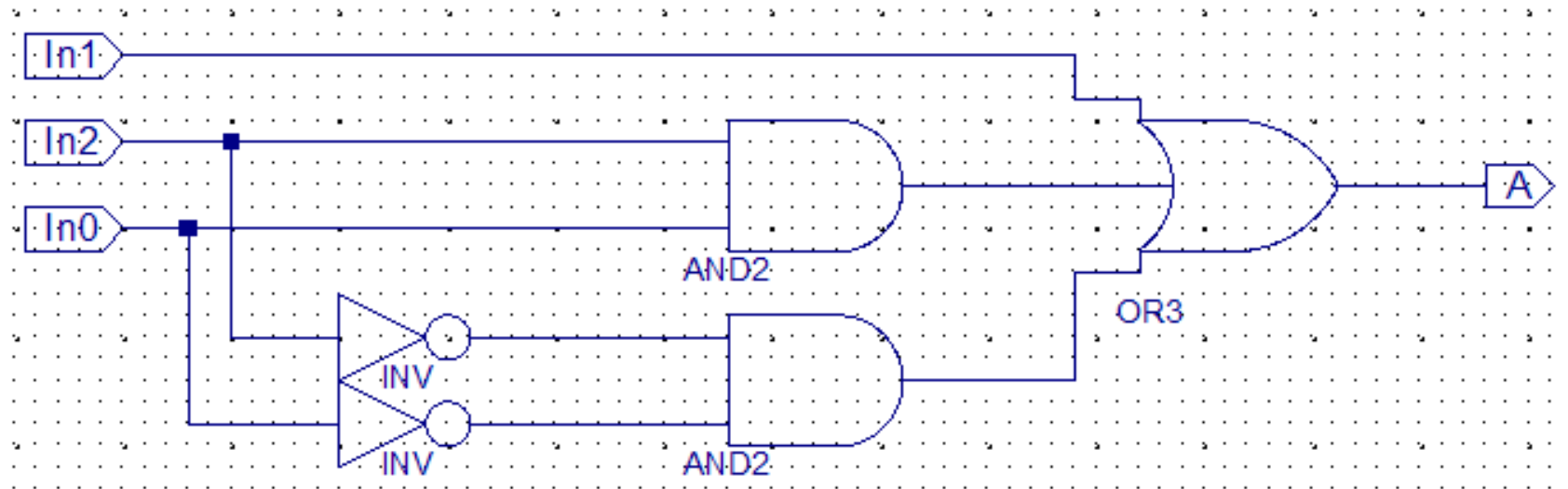
		In1 In2			
		00	01	11	10
In0	0	1	0	1	1
	1	0	1	1	1

		In1 In2			
		00	01	11	10
In0	0	1	0	1	1
	1	0	1	1	1

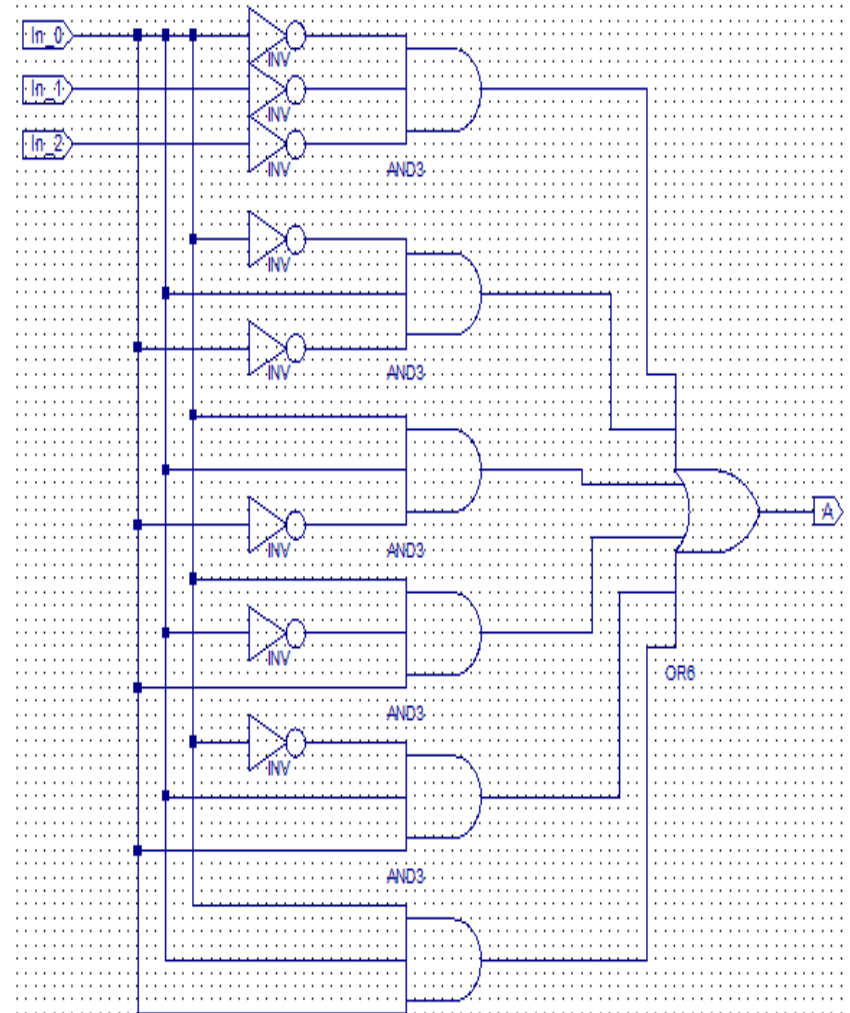
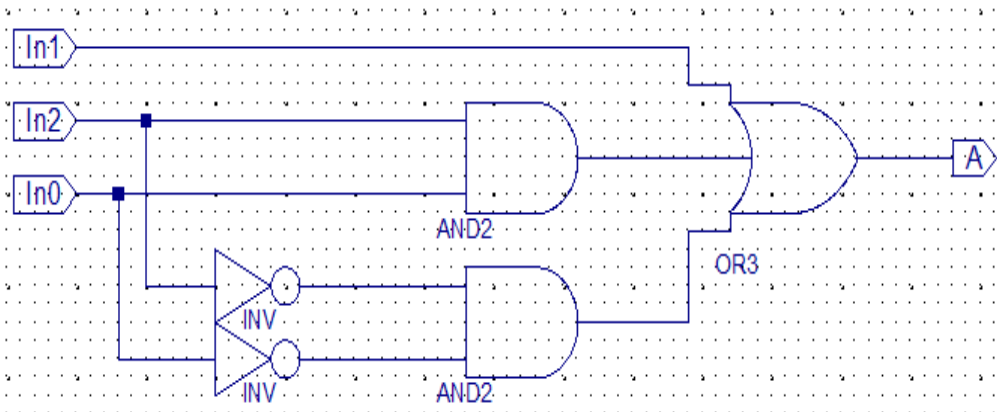
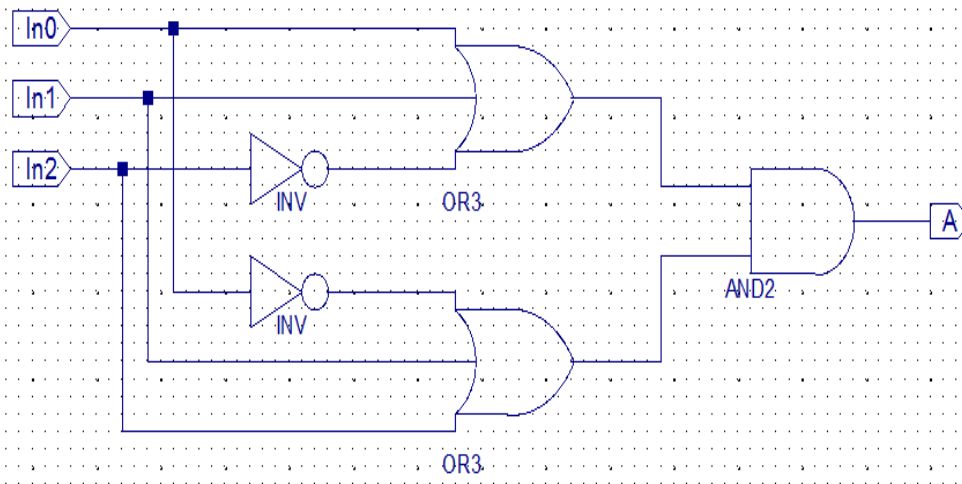
		In1 In2			
		00	01	11	10
In0	0	1	0	1	1
	1	0	1	1	1

7 segment optimizācija

- A ir atkarīgs tikai no In1 signāla – dzeltenais
- A ir atkarīgs no In0 un In2 signāliem – violetais
- A ir atkarīgs no invertētiem In0 un In2 signāliem - zaļais



Shēmu salīdzinājums



Uzdevums Nr. 3

- Katram izvēlēties un optimizēt kādu no 7segment segmentiem (B, C, D, E, F, G, H)
- Optimizēto shēmu uzzīmēt uz tāfeles
- Pilno 7segment kontrolieri realizēt Xilinx ISE
- Nosimulēt izveidoto shēmu

Papildkods

- Negatīvo skaitļu pieraksts binārajā sistēmā
 - Ir divi veidi:
 - papildināšana ar 1
 - Papildināšana ar 2
- Papildināšana ar 2
 - Invertē doto bināro skaitli, tad pieskaita klāt '1'
- Binārais skaitlis – 01010110
- Inversais binārais skaitlis – 10101001
- Pieskaita klāt '1' -> 10101010
- $no - 2^{(N-1)} līdz + 2^{(N-1)} - 1$

Mājas darbs

- Izveidot no CMOS tranzistoriem sekojošus loģiskos elementus
 - NAND, AND
 - NOR, OR
 - XOR
- Izmantojot loģiskos elementus, NAND, NOR, NOT izveidot elementu XOR (2 varianti)
 - Vadi krustojas
 - Vadi nekrustojas

Paldies par uzmanību!
Jautājumi?