

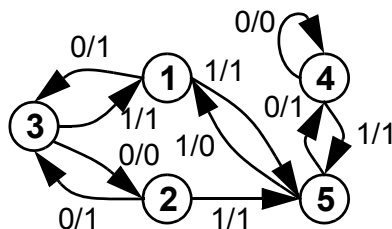


# Sünkroonsete loogikaskeemide optimeerimine

- **Oleku-põhised mudelid**
  - siirde diagrammid või tabelid (FSM)
  - ilmutatud olek / ilmutamata pindala ja viide
- **olekute arvu minimeerimine**
  
- **Struktuursed mudelid**
  - sünkroonsed loogikavõrkgraafid
  - ilmutamata olek / ilmutatud pindala ja viide
- **skeemi optimeerimine**
  
- **Sarnane mitmetasemeliste loogikafunktsioonide minimeerimisega**
  
- **Esitusviis – modifitseeritud loogikavõrkgraaf**
  - mäluelemendid / registrid

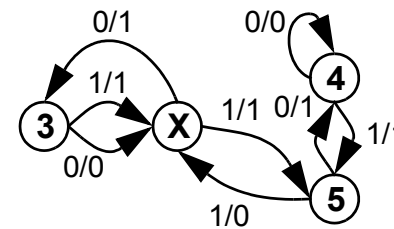
## Automaadi minimeerimine

- Automaadi minimeerimine – olekute arvu vähendamine
- Ekvivalentsed olekud – võrdsete sisendjadade korral väljundjadad katuivad



I: 001001100011...  
 O: 101101010111...  
 S: 1325445132315...

“must kast #1” == “must kast #2”

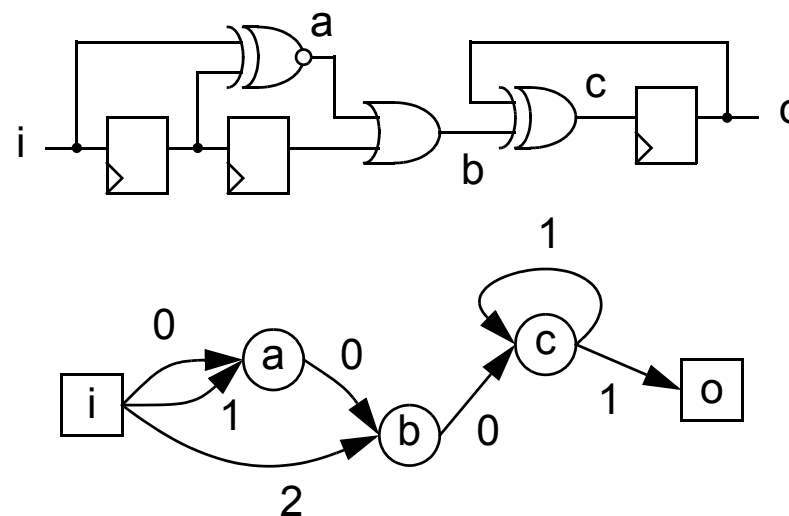


I: 001001100011...  
 O: 101101010111...  
 S: X3X5445X3X3X5...

- Intuitiivne lähenemine
  - kaks olekut on ekvivalentsed, kui nende siirded on identsed (ei pruugi alati töötada!)
- Süstemaatiline lähenemine
  - olekud 1 & 2 on ekvivalentsed
  - leida selline olekute hulga tükeldus, et ekvivalentsed olekud oleksid samas tükelduste plokis (ekvivalentsiklassis)
  - olekute hulk  $S = \{1, 2, 3, 4, 5\}$ , tükeldus  $P = \{ \overline{1, 2}; \overline{3}; \overline{4}; \overline{5} \}$  (S.P. tükeldus!)

## Automaadi skeemi minimeerimine

- **Loogikavõrkgraaf (logic network)**
  - omavahel ühendatud loogikafunktsioonid
  - kombineeritud struktuurne/käitumuslik mudel
- **Sünkroonne loogikavõrkgraaf**
  - sünkroonne viide
    - sõlmed == võrrandid == s/v, loogikalülid
    - servad == sõltuvused == ahelad
    - kaalud == sünkroonne viide == registrid
  - **Pindala ennustuse minimeerimine**
    - literaalide arv / funktsioonide/loogikalülide arv
    - arvestada tuleb viite piiranguid
  - **Suurima viite minimeerimine**
    - tee sügavus / loogikalülide mudelid / olulised teed
    - arvestada tuleb pindala (võimsustarbe) piiranguid
  - **Testitavuse maksimaliseerimine ja/või võimsustarbe minimeerimine**



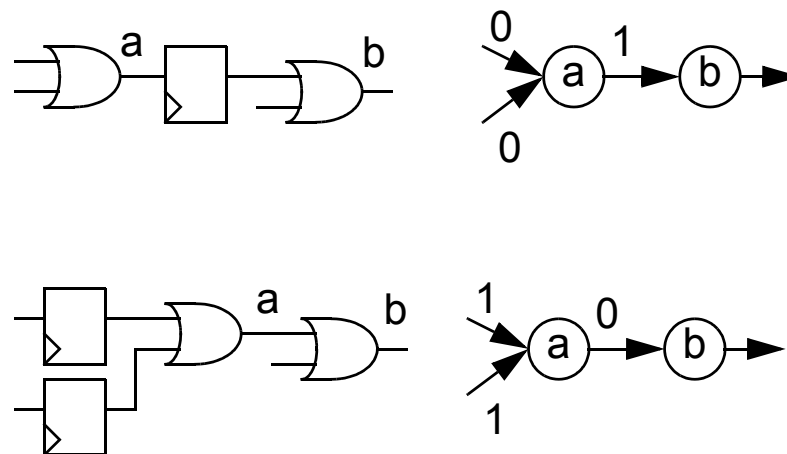


# Optimeerimine

- **Optimeeritakse ainult kombinatoorset loogikat**
  - loogika minimeerimine
    - kahe- ja/või mitme-tasemeline minimeerimine
- **Optimeeritakse ainult mälulementide / registrite asukohti**
  - *retiming* (ümberajastamine)
- **Optimeeritakse tervet skeemi**
  - sisendi/väljundi ajastuse muutmine
    - sünkroonsed teisendused – algebralised / Boole'i
- **Optimeerimis-strateegiad**
  - **Samm-sammuline parendamine**
    - teisendused (transformations) võrkgraafil
  - **Funktsionaalsus ei tohi muutuda**
  - **Erinevad meetodid**
    - erinevad teisenduste variandid / erinevad teisenduste rakendamise järjekorrad

## Retiming (ümberajastamine) – optimeerimise etapid

- Registrid eraldatakse kombinatoorsest osast
- Kombinatsiooniskeemide minimeerimine
  - avaldiste modifitseerimine
  - graafi struktuuri modifitseerimine
- Registrid ühendatakse tagasi
- Registrate asukohti muudetakse
  - kombinatoorne osa ei muutu
  - muutuvad kriitilised teed
- Võrkgraafi struktuur ei muutu
  - muutuvad kaalud
  - graafi struktuur ei muutu





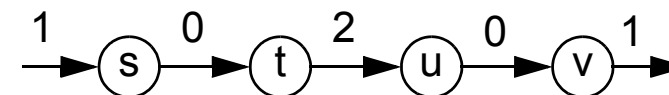
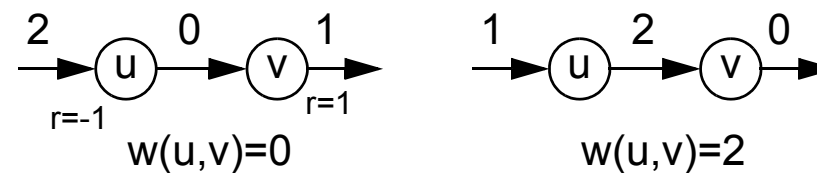
# Retiming

- **Globaalne optimeerimistehnika**
  - mõjutab pindala – registrite arv muutub
  - mõjutab viidet – teede pikkused registrite vahel muutuvad
  - lahendatav polünomaalse ajaga
- **Eeldused**
  - sõlme viide on konstantne
  - väljundite arvust sõltuvat viite komponenti ei arvestata
  - graafi topoloogia on invariantne
    - ei teostata loogika teisendusi
  - sünkroonne realisatsioon
    - tsüklitel on positiivsed kaalud
    - servadel on mitte-negatiivsed kaalud
  - arvestatakse ainult topoloogilisi teid
    - “false path” analüüsi ei teostata

## Retiming – definitsioone ja omadusi

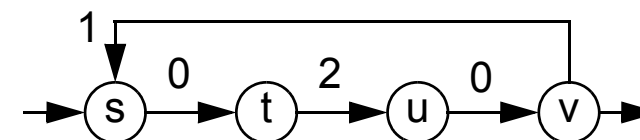
- **Sõlme retiming**
  - täisarvuline teisendus, register nihutatakse väljundist sisendisse
- **Skeemi retiming**
  - sõlmede retiming'ute vektor
- **Ekvivalentsete skeemide kogum**
  - originaalne skeem + retiming vektorid
- **Definitsioonid**
  - $w(v_i, v_j)$  – serva  $(v_i, v_j)$  kaal
  - $(v_i, \dots, v_j)$  – tee sõlmest  $v_i$  sõlmeni  $v_j$
  - $d(v_i, \dots, v_j)$  – tee  $(v_i, \dots, v_j)$  viide (sõlmest  $v_i$  sõlmeni  $v_j$ )
- **Omadused**
  - serva retiming:  $\bar{w}_{ij} = w_{ij} + r_j - r_i$
  - tee retiming:  $\bar{w}(v_i, \dots, v_j) = w(v_i, \dots, v_j) + r_j - r_i$
  - tsüklite kaalud on invariantseid

*serva retiming*



*tee*

$$w(s, t, u, v) = 2$$



*tsükkel*

$$w(\text{tsükkel}) = 3$$



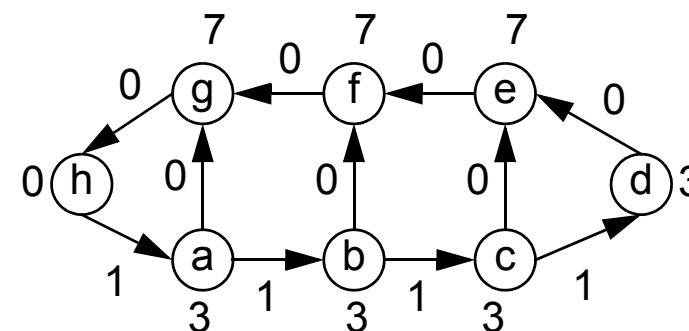
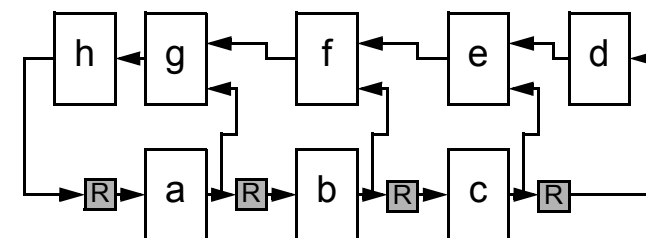
## Legaalne retiming

- Taktsignaali periood  $\phi$
- Retiming'u vektor peab tagama:
  - ühegi serva kaal ei ole negatiivne:  $\bar{w}_{ij} = w_{ij} + r_j - r_i \geq 0 \quad \forall i, j$
  - iga tee  $(v_i, \dots, v_j)$ , mille puhul  $d(v_i, \dots, v_j) \geq \phi$ , omab vähemalt ühte registrit:  
$$\bar{w}(v_i, \dots, v_j) = w(v_i, \dots, v_j) + r_j - r_i \geq 1 \quad \forall i, j$$
- Originaalne graaf ei oma tsükleid negatiivse kaaluga, seega ka uus graaf ei oma tsükleid negatiivse kaaluga



## Retiming – optimeerimisülesanne

- Lühim registrite tee –  $W(v_i, v_j) = \min w(v_i, \dots, v_j)$   
[kõik teed  $v_i$  ja  $v_j$  vahel]
- Kriitiline viide –  $D(v_i, v_j) = \max d(v_i, \dots, v_j)$   
[kõik teed  $v_i$  ja  $v_j$  vahel kaaluga  $W(v_i, v_j)$ ]
- Leida selline minimaalne taktsignaali periood  $\phi$ ,  
et eksisteeriks retiming vektor, mille puhul
- $r_i - r_j \leq w_{ij} \quad \forall (v_i, v_j) \in E$
- $r_i - r_j \leq W(v_i, v_j) - 1 \quad \forall v_i, v_j \mid D(v_i, v_j) > \phi$



sõlmed: a & e

teed: (a,b,c,e) & (a,b,c,d,e)

$W(a,e)=2$  &  $D(a,e)=16$

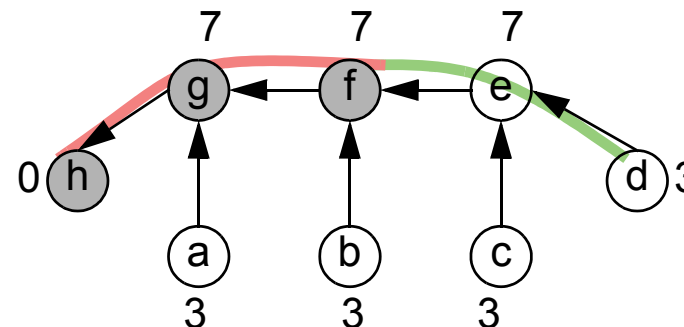
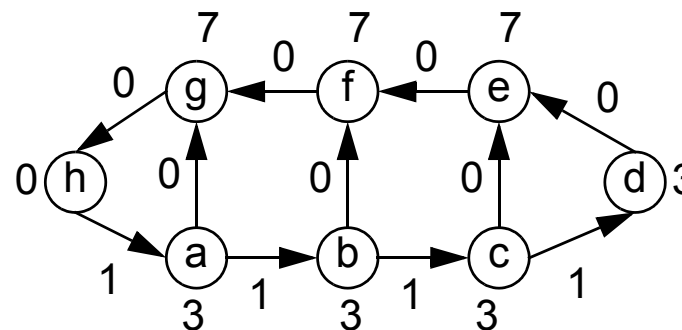


## Relaktsioonil põhinev algoritm

- **Otsib suure viitega teid**
- **Lühendab selliseid teid otsimise registri “koomale tõmbamise” abil**
  - mõni teine teed võib muutuda liiga pikaks
  - eriti need teed, mille “saba” liikus
- **Kasutab iteratiivset lähenemist**
  
- **Iga sõlme jaoks leitakse andmete valmisoleku aeg (data ready time)**
  - koguviide registrist alates
- **Iteratiivne lähenemine**
  - leiab sõlmed, mille andmete valmisoleku aeg  $> \phi$
  - sellistel sõlmedel teostatakse retiming 1 võrra (register väljundist sisendisse)
- **Omadused**
  - legalne retiming leitakse  $|N|$  iteratsiooniga (kui üldse leidub)

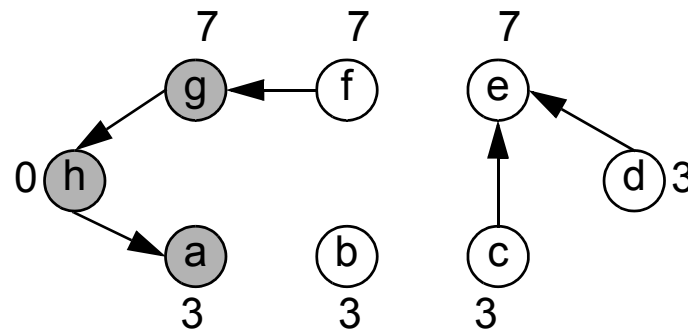
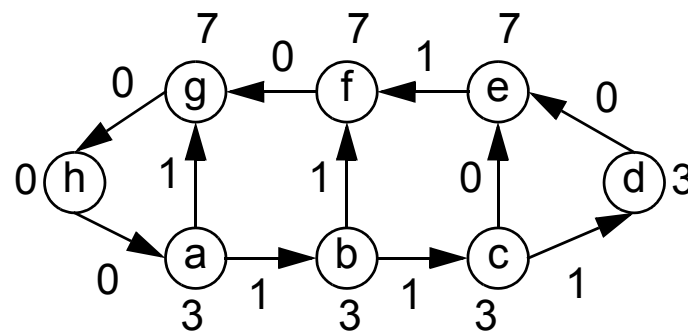
## Retiming – näide

- Ülemine gaaf – originaalne skeem
- Alumine graaf – registrid eemaldatud
  - kombinatsioonskeem
- $\phi = 13$
- andmete valmisoleku ajad:
  - $t_a=3, t_b=3, t_c=3, t_d=3, t_e=10, t_f=17, t_g=24, t_h=24$
- sõlmedele f, g ja h tehakse retiming 1 võrra



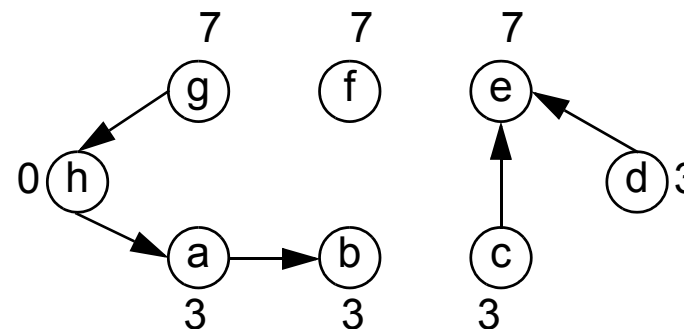
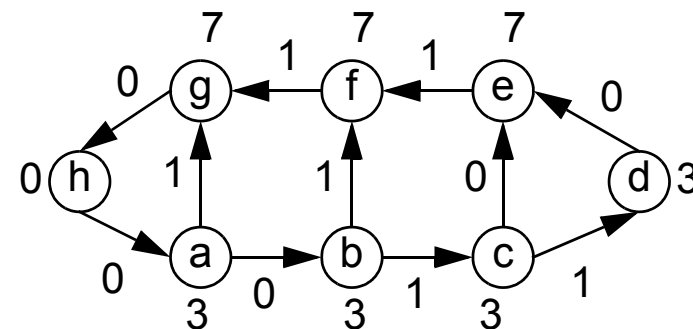
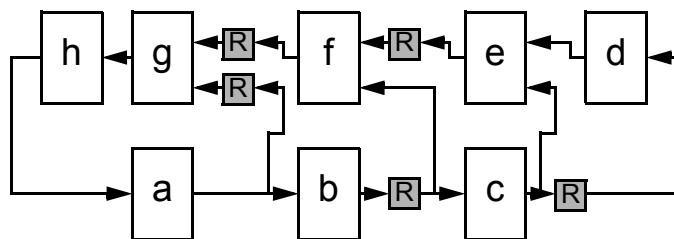
## Retiming – näide (järg)

- andmete valmisoleku ajad:
- $t_a=17$ ,  $t_b=3$ ,  $t_c=3$ ,  $t_d=3$ ,  $t_e=10$ ,  $t_f=7$ ,  $t_g=14$ ,  $t_h=14$
- sõlmedele a, g ja h tehakse retiming 1 võrra



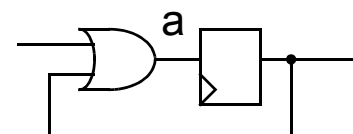
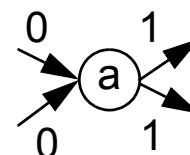
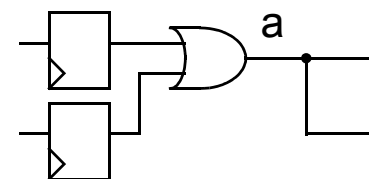
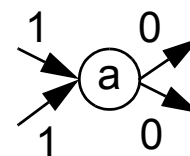
## Retiming – näide (järg)

- andmete valmisoleku ajad:
  - $t_a=10, t_b=13, t_c=3, t_d=3, t_e=10, t_f=7, t_g=7, t_h=7$
- lõpp –  $\forall t_i \leq \phi$



## Retiming – rakendatavus

- **Registrite arvu minimeerimine**
  - sünkroonse viite nihutamine sisenditest väljundisse
- **Kontroll-osa**
  - abstrakne automaadi mudel peidetud võrkgraafi sisse
- **Andme-osa**
  - loogikafunktsioonid
  - mälu elemendid / registrid
- **Mitte ainult loogikaskeemid vaid ka kõrgemad abstraktsiooni tasemed**
  - sõlmed – funktsioonid
  - sünkroonsed viited – nt. kriitilised ressursid (mälud)





# Digitaalsüsteemide testimise alused

## Funktsioonide teisendused – kahendmeetodid

- **Kasutavad fakti, et osa sisendmuutujaid ja/või võrgugraafi sõlmi ei mõjuta osasid funktsioone**
  - toob sisse täiendavad määramatused
  - võimalik kasutada lokaalset kahendminimeerimist
- ***Juhitavus* (controllability)**
  - sisendkombinatsioonid, mida ei esine võrgu sisendis (ümbritsevast keskkonnast tingituna)
- ***Jälgitavus* (observability)**
  - sisendkombinatsioonid, mille korral väljund ei ole vaadeldav keskkonna poolt
  - suhteline iga väljundi jaoks
- **Seotud funktsioonide testitavusega**



## Kahendmeetodid – sisemised määramatused

- **Võrgu sise-ehitusest sõltuvad**
  - **Juhitavus** – kombinatsioonid, mida ei esine alamvõrgu sisendis
  - **Jälgitavus** – kombinatsioonid, mille korral alamvõrgu väljundid ei ole vaadeldavad
- **Näide #1**
  - $x = \bar{a} + b; \quad y = a b x + \bar{a} c x;$
  - **juhitavus** –  $y$  sisendis ei saa esineda kombinatsiooni  $a \bar{b} x + \bar{a} \bar{x} + b \bar{x}$
  - **minimeerides** –  $y = a x + \bar{a} c$
- **Näide #2**
  - $v = \bar{a} d + b d + \bar{c} d + a\bar{e}; \quad (\text{dekompositsioon}) \quad j = \bar{a} + b + \bar{c}; \quad v = jd + a\bar{e};$
  - $a=0 \rightarrow j=1$  : seega on võimatu kombinatsioon:  $a=0 \ \& \ j=0 \rightarrow v=''$
  - **minimeerides** –  $v = jd + a\bar{e}$  (alati ei muutu lihtsamaks)



$$x = \bar{a} + b; \quad y = abx + \bar{a}cx;$$

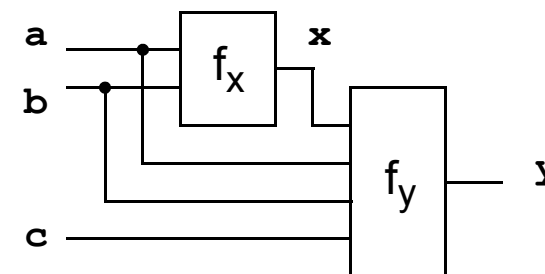
- $x=0 \rightarrow a=1 \ \& \ b=0$
- $x=1 \rightarrow a=0 \ \vee \ b=1$
- **Võimatud kombinatsioonid:**
  - $x=0 \ \& \ !(a=1 \ \& \ b=0) \rightarrow x=0 \ \& \ (a=0 \ \vee \ b=1)$
  - $x=1 \ \& \ !(a=0 \ \vee \ b=1) \rightarrow x=1 \ \& \ a=1 \ \& \ b=0$

- **Määramatused:**

- $x=0 \ \& \ a=0; \ x=0 \ \& \ b=1; \ x=1 \ \& \ a=1 \ \& \ b=0$

- **Tulemus**

$$y = \bar{a}c + ax$$



		<u>a</u>		<u>b</u>	
		0	1	0	1
c	x	0	0	0	0
		0	0	0	0
		1	0	1	1
		0	0	1	0

		<u>a</u>		<u>b</u>	
		0	1	0	1
c	x	-	0	-	-
		-	0	-	-
		1	-	1	1
		0	-	1	0

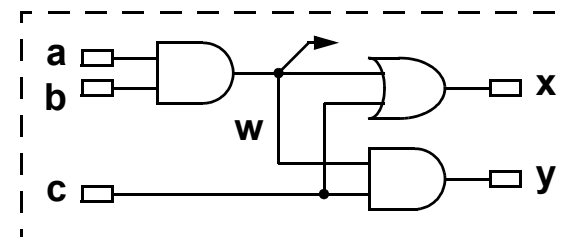
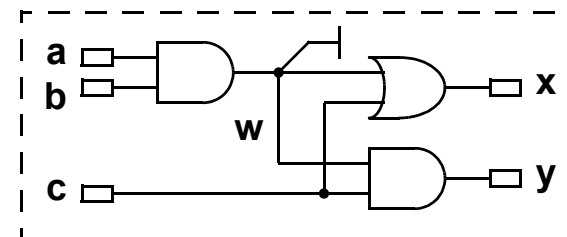
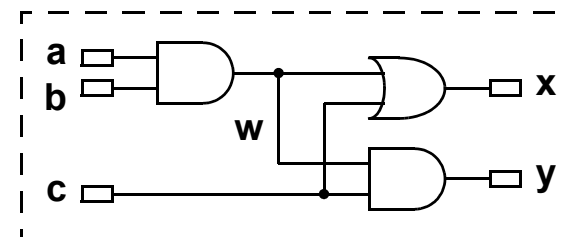


## Testitavuse alused

- **Rikke mudel – mingi ahela *lühis* 0 või 1-ga (stuck-at-0/stuck-at-1)**
- **Ahela w kontroll lühisele 0-ga**
  - sisendkombinatsiooniga seatakse vastav ahel 1-ks
  - võrreldakse väljundeid – vigase skeemi väljund on erinev soovitud
- **Ahela w kontroll lühisele 1-ga**
  - sisuliselt sama, kuid jälgitav ahel seatakse 0-ks
- **Ahel w peab olema juhitud ja jälgitud**
  - testitavus sõltub skeemi struktuurist
- **Süntees testitavust silmas pidades**
  - võimalikult suur osa sisemisi ahelaid peaks olema jälgitud ja juhitud
  - liiasuste eemaldamine mitmetasemelisel loogikafunktsioonide minimeerimisel

## Näide

- Skeem:  $w=ab$ ;  $x=w+c$ ;  $y=wc$ ;
- Lühis 0-ga  
 $w=0$ ;  $\rightarrow$   $x=c$ ;  $y=0$ ;
- Lühis 1-ga  
 $w=1$ ;  $\rightarrow$   $x=1$ ;  $y=c$ ;
- Kas  $w$  on jälgitav?
  - Millised väljundid sõltuvad  $w$ -st?
- Kas  $w$  on juhitav?
  - Kas leidub sisendkombinatsioon, mis lubab  $w$ -le seada soovitud väärtuse?





- **Jälgitavus - Boole'i diferentsiaal -  $\partial f/\partial x_i = f_{x_i} \oplus f_{x_i'}$** 
  - Kas väljund  $x$  ( $x=w+c$ ) sõltub  $w$ -st? Kas väljund  $y$  ( $y=wc$ ) sõltub  $w$ -st?
  - Kofaktorid --  $x_w=1$ ;  $x_w'=c$ ;  $y_w=c$ ;  $y_w'=0$ ;
  - $\partial x/\partial w = x_w \oplus x_w' = 1 \oplus c = c'$  (sõltub siis, kui  $c=0$ )
  - $\partial y/\partial w = y_w \oplus y_w' = c \oplus 0 = c$  (sõltub siis, kui  $c=1$ )
  - Nii  $x$  kui ka  $y$  sõltuvad  $w$ -st, kuid erinevatel  $c$  väärtustel  
→  $w$  on jälgitav väljunditel  $x$  ja  $y$  (teatud mõõndustega)
- **Juhitavus - soovitud väärtuse seadmine  $w$ -l**
  - $w=a$  b
  - lühis 0-ga kontrollimiseks →  $w=1$  →  $a=1$  ja  $b=1$
  - lühis 1-ga kontrollimiseks →  $w=0$  →  $a=0$  või  $b=0$
  - BDD'd (või muud otsustus diagrammid) sobivad selleks suurepäraselt



- **Ahel  $w$  peab olema jälgitav ja juhitav**
  - Peab leiduma ühisosa jälgitavust ja juhitavust määravate sisend-kombinatsioonide vahel, vastasel korral pole mõni riketest määratav
  - Funktsionaalne test – ainult töö õigsuse kontroll, rike ei pruugi olla määratav
  - Diagnostika – konkreetse rikke (või isegi mitme rikke) täpne määramine
- **Konsensus ( $C^X_w = x_w \cdot x_w'$ ) – milline osa ei sõltu  $w$ -st**
  - $C^X_w = x_w \cdot x_w' = 1 \cdot c = c$ ;  $C^Y_w = y_w \cdot y_w' = c \cdot 0 = 0$ ;
- **Häiritus (perturbation)**
  - **Lühis 0-ga** -  $\delta_w = w \cdot (\partial x / \partial w) = x \oplus x_w'$ 
    - $x$  -  $\delta^x_w = w \cdot (\partial x / \partial w) = x \oplus x_w' = w \cdot c'$
    - $y$  -  $\delta^y_w = w \cdot (\partial y / \partial w) = y \oplus y_w' = w \cdot c$
  - **Lühis 1-ga** -  $\delta_w = w' \cdot (\partial x / \partial w) = x' \oplus x_w$ 
    - $x$  -  $\delta^x_w = w' \cdot (\partial x / \partial w) = x' \oplus x_w = w' \cdot c'$
    - $y$  -  $\delta^y_w = w' \cdot (\partial y / \partial w) = y' \oplus y_w = w' \cdot c$

## Näide – vektorid

- **Lühis 0-ga**

- $\delta_{w'}^x = w \cdot c'$  &  $\delta_{w'}^y = w \cdot c$

- **sisend - abc == 110**

- väljund x - 0 (peab olema 1)
  - väljund y - 0 (peab olema 0)

- **sisend - abc == 111**

- väljund x - 1 (peab olema 1)
  - väljund y - 0 (peab olema 1)

- **Lühis 1-ga**

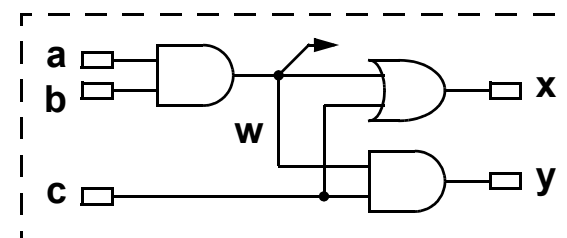
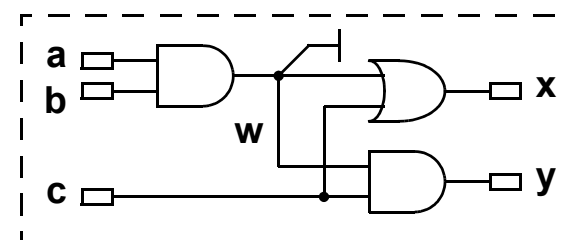
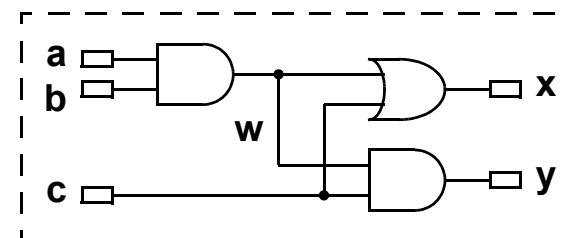
- $\delta_{w'}^x = w' \cdot c'$  &  $\delta_{w'}^y = w' \cdot c$

- **sisend - abc == 000**

- väljund x - 1 (peab olema 0)
  - väljund y - 0 (peab olema 0)

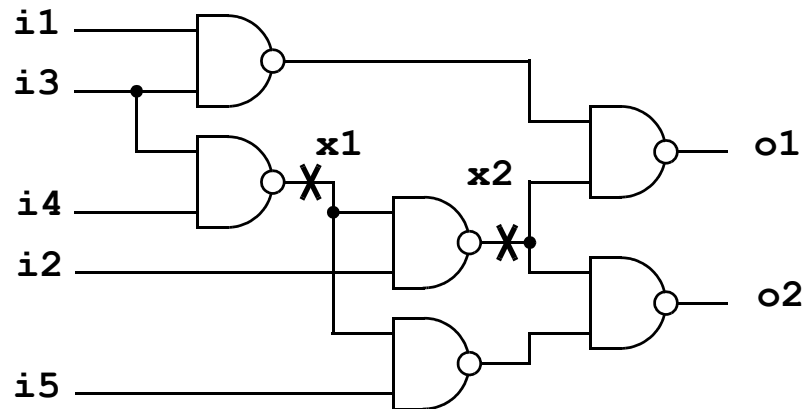
- **sisend - abc == 001**

- väljund x - 1 (peab olema 1)
  - väljund y - 1 (peab olema 0)



## Näide #2

- $x1 = (i3 \& i4)'$
- $o1 = ((i1 \& i3)' \& (i2 \& x1)')'$
- $o2 = ((i5 \& x1)' \& (i2 \& x1)')'$
- $x2 = (i2 \& (i3 \& i4)')'$
- $o1 = ((i1 \& i3)' \& x2)'$
- $o2 = (x2 \& (i5 \& (i3 \& i4)')')'$

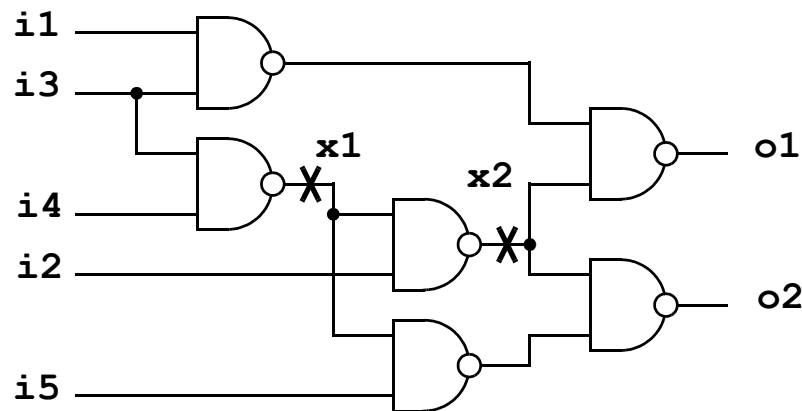


- Lühis 0-ga -  $\delta_w^x = w \cdot (\partial x / \partial w) = x \oplus x_w$       Lühis 1-ga -  $\delta_w^x = w' \cdot (\partial x / \partial w) = x' \oplus x_w$

- $\delta_{x1}^{o1} = x1 \cdot (\partial o1 / \partial x1)$ ;     $\partial o1 / \partial x1 = o1_{x1} \oplus o1_{x1'}$ ;     $o1_{x1} = i1 \ i3$ ;     $o1_{x1'} = i1 \ i3 + i2$
- $\partial o1 / \partial x1 = (i1 \ i3) \oplus (i1 \ i3 + i2) = (i1 \ i3) (i1 \ i3 + i2)' + (i1 \ i3)' (i1 \ i3 + i2)$
- $\partial o1 / \partial x1 = i1 \ i3 \ i2' (i1' + i3') + (i1' + i3') (i1 \ i3 + i2) = i1' \ i2 + i2 \ i3'$
- $\delta_{x1}^{o1} = x1 \cdot (\partial o1 / \partial x1) = (i3 \ i4)' (i1' \ i2 + i2 \ i3') = (i3' + i4') (i1' \ i2 + i2 \ i3') = i2 \ i3' + i1' \ i2 \ i4'$

## Näide #2

- $x1 = (i3 \& i4)'$
- $o1 = ((i1 \& i3)' \& (i2 \& x1)')'$
- $o2 = ((i5 \& x1)' \& (i2 \& x1)')'$
- $x2 = (i2 \& (i3 \& i4)')'$
- $o1 = ((i1 \& i3)' \& x2)'$
- $o2 = (x2 \& (i5 \& (i3 \& i4)')')'$



• Lühis 0-ga -  $\delta_{w'}^x = w \cdot (\partial x / \partial w) = x \oplus x_w$

• Lühis 1-ga -  $\delta_{w'}^x = w' \cdot (\partial x / \partial w) = x' \oplus x_w$

•  $\delta_{x1}^{o1} = x1 \cdot (\partial o1 / \partial x1) = i2 \bar{i3} + \bar{i1} i2 \bar{i4}$  ;

•  $\delta_{x1}^{o1} = \bar{x1} \cdot (\partial o1 / \partial x1) = \bar{i1} i2 i3 i4$

•  $\delta_{x1}^{o2} = x1 \cdot (\partial o2 / \partial x1) = i2 \bar{i3} + i2 \bar{i4} + \bar{i3} i5 + \bar{i4} i5$  ;

•  $\delta_{x1}^{o2} = \bar{x1} \cdot (\partial o2 / \partial x1) = i2 i3 i4 + i3 i4 i5$

•  $\delta_{x2}^{o1} = x2 \cdot (\partial o1 / \partial x2) = \bar{i1} \bar{i2} + \bar{i2} \bar{i3} + \bar{i1} i3 i4$  ;

•  $\delta_{x2}^{o1} = \bar{x2} \cdot (\partial o1 / \partial x2) = \bar{i1} i2 \bar{i4} + i2 \bar{i3} \bar{i4}$

•  $\delta_{x2}^{o2} = x2 \cdot (\partial o2 / \partial x2) = \bar{i2} \bar{i5} + i3 i4$  ;

•  $\delta_{x2}^{o2} = \bar{x2} \cdot (\partial o2 / \partial x2) = i2 \bar{i3} \bar{i5} + i2 \bar{i4} \bar{i5}$



## Näide #2 – vektorid

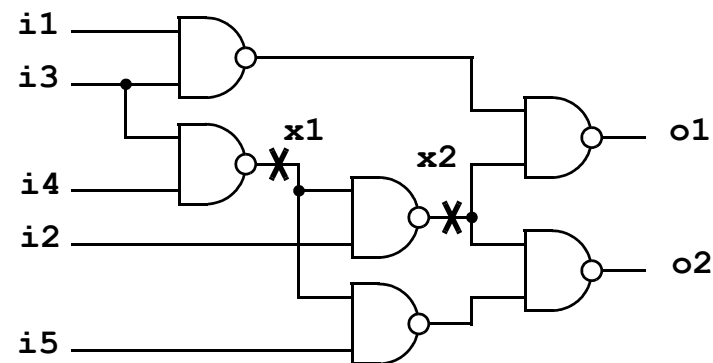
- $\delta_{x1}^{o1} = x1 \cdot (\partial o1 / \partial x1) = i2 \bar{i3} + \bar{i1} i2 \bar{i4}$  ;
  - $\delta_{x1}^{o2} = x1 \cdot (\partial o2 / \partial x1) = i2 \bar{i3} + i2 \bar{i4} + \bar{i3} i5 + \bar{i4} i5$  ;
  - $\delta_{x2}^{o1} = x2 \cdot (\partial o1 / \partial x2) = \bar{i1} \bar{i2} + \bar{i2} \bar{i3} + \bar{i1} i3 i4$  ;
  - $\delta_{x2}^{o2} = x2 \cdot (\partial o2 / \partial x2) = \bar{i2} \bar{i5} + i3 i4$  ;
- $\delta_{x1}^{o1} = \bar{x1} \cdot (\partial o1 / \partial x1) = \bar{i1} i2 i3 i4$
  - $\delta_{x1}^{o2} = \bar{x1} \cdot (\partial o2 / \partial x1) = i2 i3 i4 + i3 i4 i5$
  - $\delta_{x2}^{o1} = \bar{x2} \cdot (\partial o1 / \partial x2) = \bar{i1} i2 \bar{i4} + i2 \bar{i3} \bar{i4}$
  - $\delta_{x2}^{o2} = \bar{x2} \cdot (\partial o2 / \partial x2) = i2 \bar{i3} \bar{i5} + i2 \bar{i4} \bar{i5}$

i1...5	<u>x1</u>	<u>o1</u>	<u>o2</u>
-10--	1/0	1/0	1/0
01-0-	1/0	1/0	1/0
0111-	0/1	0/1	0/1

i1...5	<u>x2</u>	<u>o1</u>	<u>o2</u>
00---	1/0	0/1	?/1
-00--	1/0	0/1	?/1
0-11-	1/0	0/1	0/1
01-0-	0/1	1/0	1/?
-100-	0/1	1/0	1/?

i1...5	<u>x1</u>	<u>o1</u>	<u>o2</u>
-10--	1/0	1/0	1/0
-1-0-	1/0	?/?	1/0
--0-1	1/0	?/0	1/0
---01	1/0	?/?	1/0
-111-	0/1	?/1	0/1
--111	0/1	?/?	0/1

i1...5	<u>x2</u>	<u>o1</u>	<u>o2</u>
-0--0	1/0	?/1	0/1
--11-	1/0	?/1	0/1
-10-0	0/1	1/0	1/0
-1-00	0/1	1/?	1/0



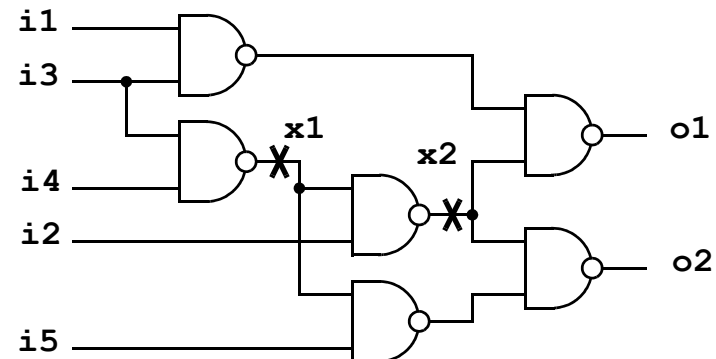
## Näide #2 – vektorite pakkimine

$i1\dots5$	$\underline{x1}$	$\underline{o1}$	$o2$
-10--	1/0	1/0	1/0
01-0-	1/0	1/0	1/0
0111-	0/1	0/1	0/1

$i1\dots5$	$\underline{x1}$	$o1$	$\underline{o2}$
-10--	1/0	1/0	1/0
-1-0-	1/0	?/?	1/0
--0-1	1/0	?/0	1/0
---01	1/0	?/?	1/0
-111-	0/1	?/1	0/1
--111	0/1	?/?	0/1

$i1\dots5$	$\underline{x2}$	$\underline{o1}$	$o2$
00---	1/0	0/1	?/1
-00--	1/0	0/1	?/1
0-11-	1/0	0/1	0/1
01-0-	0/1	1/0	1/?
-100-	0/1	1/0	1/?

$i1\dots5$	$\underline{x2}$	$o1$	$\underline{o2}$
-0--0	1/0	?/1	0/1
--11-	1/0	?/1	0/1
-10-0	0/1	1/0	1/0
-1-00	0/1	1/?	1/0



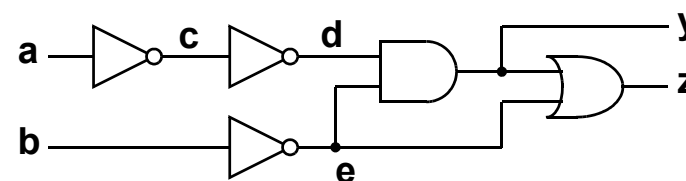
$i1\dots5$	0	$\underline{x1}$	1	0	$\underline{x2}$	1
01000	*	.	.	.	*	.
0111-	.	*	*	*	.	.

- **Testimine – vähim arv (osaliselt) kattuvaid vektoreid, et katta võimalikult palju rikkeid**
  - Veel üks katte leidmise ülesanne!
- **Diagnostika – unikaalsed vektorid rikete identifitseerimiseks**
  - Ei leidu...

## Viite minimeerimine ja testimine

### Väär topoloogiline kriitiline tee - näide

- Kõikidel loogikalülidel ühikviide
- Kõik sisendid valmis ajahetkel 0
- Pikim topoloogiline tee
  - (a,c,d,y,z) – viide 4
- Tõeline kriitiline tee
  - (a,c,d,y) – viide 3

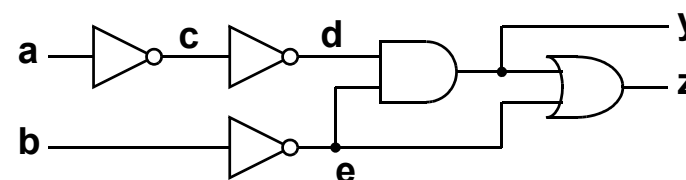


### Tundlik kriitiline tee

- Sündmus levib algusest lõpuni
- Mitte-tundlikud kriitilised teed on väärad (ja neid võib ignoreerida)

## Dünaamiline tundlikkuse määramine

- Tee –  $P = (v_0, v_1, \dots, v_m)$
- Sündmus levib mööda teed, kui  $\partial f_{x_i} / \partial x_{i-1} = 1 \quad \forall i=1,2,\dots,m$ 
  - Kõrvalsisendid (side-inputs) - sisendid, mis pole teel
  - Boole'i diferentsiaal on funktsioon kõrval-sisenditest (mille väärtused võivad muutuda)
  - Boole'i diferentsiaal peab olema tõene ajal, mil sündmus levib
- Tee - (a,c,d,y,z)
  - $\partial f_y / \partial d = e = 1$  ajahetkel 2
  - $\partial f_z / \partial y = e' = 1$  ajahetkel 3
  - ei ole dünaamiliselt tundlik, sest e stabiliseerub ajahetkel 1
- Alternatiiv – staatiline tundlikkuse määramine
  - Lihtsustatud mudel – ajalisi tingimusi Boole'i diferentsiaali väärtuse jaoks ei eksisteeri
  - Oht – liiga väikeste viidete ennustamine





## Automaatide testitavuse alused

- **Loogikafunktsioonid**
  - sisendid seatakse soovitud väärtustele
  - väljundeid võrreldakse eeldatud väärtustega
- **Mäluelemendid**
  - sisendid seatakse soovitud väärtustele
  - väljundeid võrreldakse eeldatud väärtustega
- **Probleem!**
  - mäluelementide sisendid/väljundid pole üldjuhul otseselt seatavad/nähtavad
  - kaudne seadmine ja võrdlemine



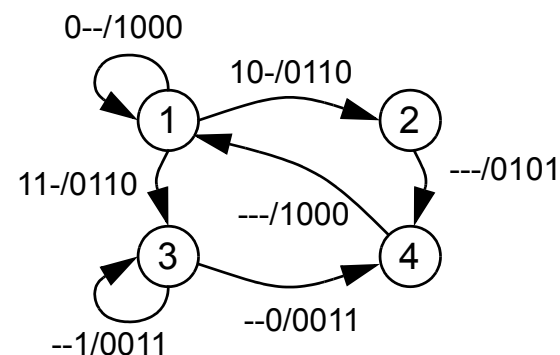
## Seade- ja kontrolljada

- **Automaadi seadejada**
  - **sisendkombinatsioonide jada, mis viiks soovitud olekusse**
    - lähteolekust soovitud olekusse
    - suvalisest olekust soovitud olekusse
  - **võimalik lahendus**
    - suvalisest olekust lähteolekusse
    - lähteolekust soovitud olekusse
- **Automaadi kontrolljada**
  - **sisendkombinatsioonide jada, mis viitaks üheselt, et automaat oli mingis kindlas olekus (läbis mingit kindlat olekut)**
    - eri olekud võivad genereerida sarnaseid väljundsignaale
    - eristatavad väljundkombinatsioonide jadad

## Seadejada – näide

- **Soovitud olek – 2**
  - 1 → 2 : 1 takt, sisend “10-”
  - 1 → 1 : 0 takti
  - 2 → 1 : 2 takti, sisendjada “---”, “---”
  - 3 → 1 : 2 takti, sisendjada “--0”, “---”
  - 4 → 1 : 1 takt, sisendjada “---”
- **Ühepikkused jasad vajalikud (ootab mõne takti olekus 1)**
  - 1 → 1 : 2 takti, sisendjada “0--”, “0--”
  - 4 → 1 : 2 takti, sisendjada “---”, “0--”
- **Ühepikkuste sisendjadade ühisosa, pluss soovitud olekusse minekuks vajalik jada: “0-0”, “0--”, “10-”**
  - 1 → 1 → 1 → 2, 2 → 4 → 1 → 2,  
3 → 4 → 1 → 2, 4 → 1 → 1 → 2

I	$s^t$	$s^{t+1}$	O
0 --	1	1	1 0 0 0
1 0 -		2	0 1 1 0
1 1 -		3	0 1 1 0
---	2	4	0 1 0 1
-- 0	3	4	0 0 1 1
-- 1		3	0 0 1 1
---	4	1	1 0 0 0





## Sisse-ehitatud testitavus

- **Lisavahendid mälulementide otseseks seadmiseks**
  - nihkeregistrid
  - spetsiaalsed testjadade sisendid ja väljundid
- **Kasutusel nii kontroll-osa kui ka andme-osa puhul**
- **Täiendav riistvara**
  - suureneb pindala
  - suureneb viide
  - suureneb volutarve
  - ka täiendavaid osi tuleks testida
- **Ainult osa mälulemente on otseselt seatavad ja kontrollitavad**
- **BIST – Built-In Self-Test**

