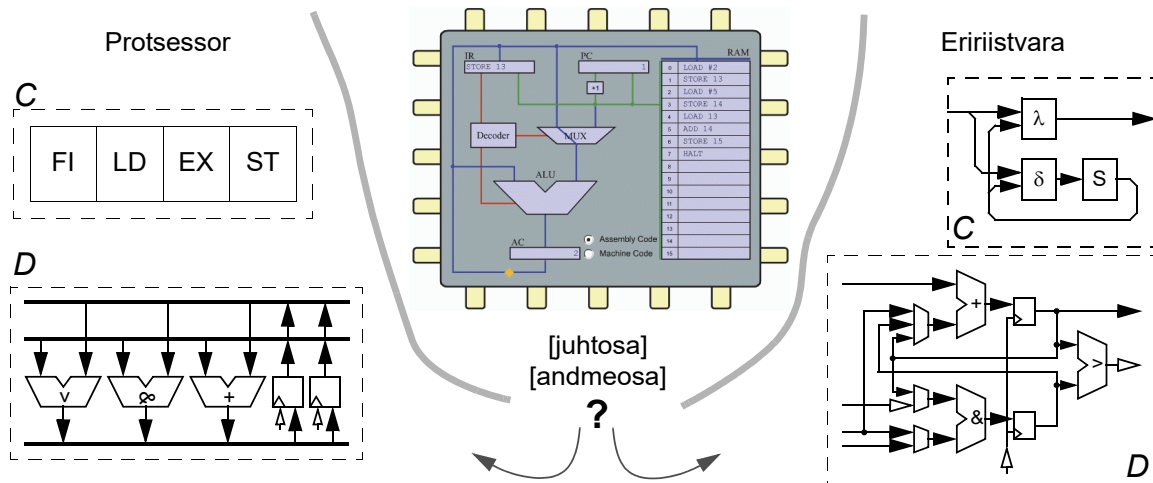
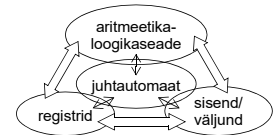


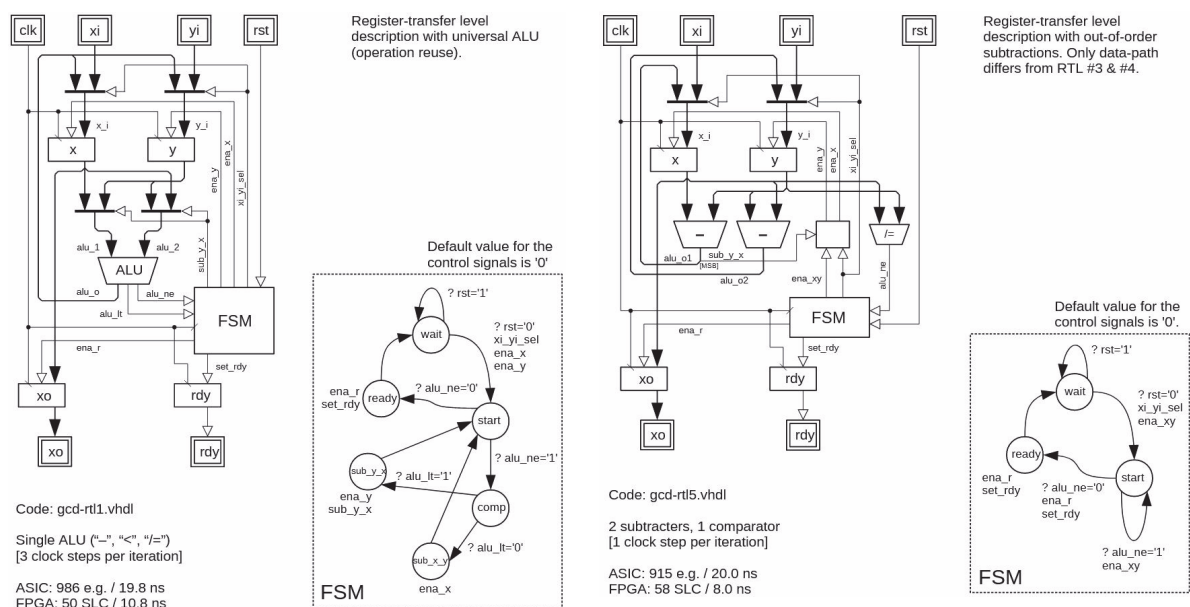
Digitaalsüsteem – struktuur & juhtautomaat



- **Protsessor – tarkvara**
 - universaalne andmetee – aeglane, suured mõõtmed, energianäljane
 - universaalne algoritm – äärmiselt paindlik, kuid nõuab palju ruumi (bitte)
- **Eriistvara**
 - fikseeritud andmetee – kiire, kompaktn, väike energiatarve
 - paindlikkus määratud juhtautomaadiga, algoritm sageli fikseeritud



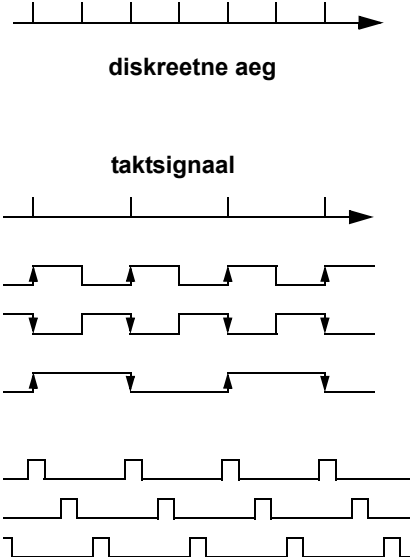
Digitaalsüsteem – andme- ja juhtosa Suurima ühisteguri leidmine (GCD)



- Sama algoritm, erinevad realisatsioonid [vt. ka <https://ati.ttu.ee/~lrv/gcd/>]

Diskreetne aeg

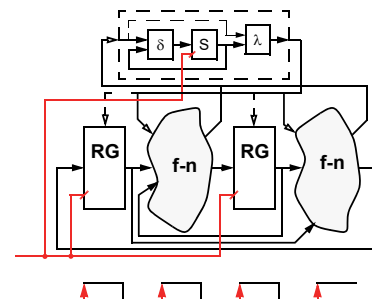
- Rangelt järjestatud sündmuste jada (ajamomentide jada)
 - üksiksündmuse kestus on 0
 - sündmuste vaheline ajavahemik pole oluline
- Taktsignaali (clock)
 - reaalne diskreetse aja esitusviis
 - üksiksündmus == taktsignaali front
 - tõusev või langev front – ühefaasiline taktsignaali (single phased clock)
 - tõusev ja langev front – kahefaasiline taktsignaali (double phased clock)
 - varasem mitmefaasiline taktsignaali – eri faasid tüürisid eri mäluelemente



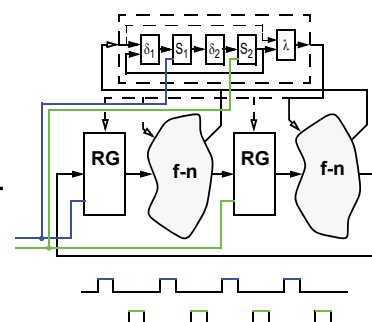
Takteerimine

- register → kombinatsiooniskeem → register → ...
- registri tüüp määrab taktsignaali faaside arvu ja lubatud tagasisideid
- flip-flop
 - frondist aktiveeruv triger
 - tagaside korral peab olema vähemalt üks triger ahelas
- latch
 - lukustuv triger (lukk-registri)
 - taktsignaali aktiivse nivoo korral on triger "läbipaistev" → tagaside selliselt, et vähemalt üks suletud triger ahelas

ühefaasiline taktsignaali – flip-flop registrid



kahefaasiline taktsignaali(id) – lukk-registrid



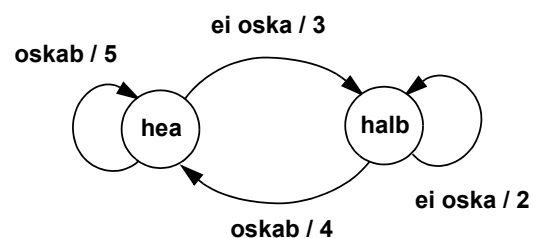
Abstraktne automaat

- **Kombinatoorne skeem**
 - skeemi väljund sõltub ainult skeemi sisendist
 - atsüklilise topoloogiaga skeem on kombinatoorne
 - tsükliga (tagasisidega) skeemid võivad olla kombinatoorsed
- **Mäluga skeem**
 - eksisteerivad mäluelemendid
 - asünkroonsetes skeemis võib mäluelement olla mitteilmutatud kujul
 - tagasiside on vajalik
- **Automaat – mäluga skeemi erijuht**
 - sisendid, väljundid ja olekud – lõplikud hulgad
 - abstraktne automaat, lõplik automaat
 - automaton (pl. automata), sequential machine, finite state machine (FSM)

Näide

- **Õppejõu käitumine eksamil**
 - kui õppejõud on heas tujus ja tudeng oskab, siis tudeng saab 5 ning õppejõu hea tuju säilib
 - kui õppejõud on heas tujus ja tudeng ei oska, siis tudeng saab 3 ning õppejõu tuju läheb halvaks
 - kui õppejõud on halvas tujus ja tudeng ei oska, siis tudeng saab 2 ning õppejõu halb tuju säilib
 - kui õppejõud on halvas tujus ja tudeng oskab, siis tudeng saab 4 ning õppejõu tuju läheb heaks

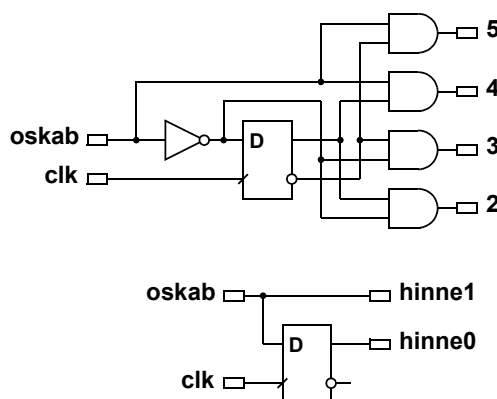
sisend	olek	väljund	uus olek
tudeng	õppejõud	hinne	õppejõud
oskab	hea tuju	5	hea tuju
ei oska	hea tuju	3	halb tuju
ei oska	halb tuju	2	halb tuju
oskab	halb tuju	4	hea tuju



Näide – õppejõu realisatsioon riistvaras

- Realisatsiooni efektiivsus sõltub kodeeringust!

- Sisend- ja väljundkodeering üldjul teada
- Olekute kodeerimine oluline
 - pindala – loogikalülide arv
 - viide – funktsioonide keerukus
 - võimsustarve – lülituste arv ajaühikus
- Realisatsioon #1
 sisend: ei oska - 0, oskab - 1
 väljund: 2 - 0001, 3 - 0010, 4 - 0100, 5 - 1000
 olek: hea - 0, paha - 1
- Realisatsioon #2
 sisend: ei oska - 0, oskab - 1
 väljund: 2 - 00, 3 - 01, 4 - 10, 5 - 11
 olek: hea - 1, paha - 0

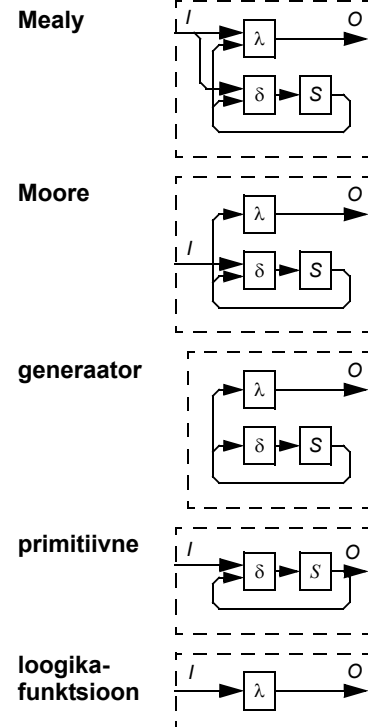


Abstraktne automaat – definitsioonid

- Automaat on viisik (quintuple) – $M = (S, I, O, \delta, \lambda)$
 - S : (sise)olekute hulk (states)
 - I : sisendite hulk (inputs)
 - O : väljundite hulk (outputs)
 - δ : siirdefunktsioon (transition) - $\delta: S \times I \rightarrow S$
 - λ : väljundfunktsioon - $\lambda: S \times I \rightarrow O$
- Hulgad on lõplikud ja (üldjuhul) mittetühjad
 - hulkade ja funktsioonide erijuhud – automaatide erijuhud
- Lähteolek s_0 – $M = (S, I, O, \delta, \lambda, s_0)$

Automaatide erijuhud

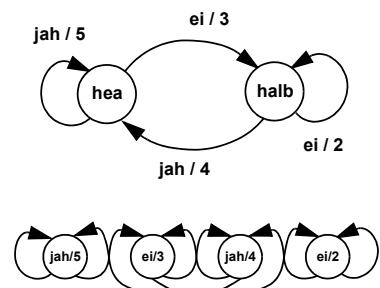
- **Mealy automaat** – $M = (S, I, O, \delta, \lambda)$
 - $S \neq \emptyset, I \neq \emptyset, O \neq \emptyset, \delta: S \times I \rightarrow S, \lambda: S \times I \rightarrow O$
- **Moore automaat** – $M = (S, I, O, \delta, \lambda)$
 - $S \neq \emptyset, I \neq \emptyset, O \neq \emptyset, \delta: S \times I \rightarrow S, \lambda: S \rightarrow O$
- **Primitiivne automaat** – $M = (S, I, \delta)$
 - $S \neq \emptyset, I \neq \emptyset, O = \emptyset (O \equiv S), \delta: S \times I \rightarrow S, \lambda \equiv \emptyset$
- **Generaator** – $M = (S, O, \delta, \lambda)$
 - $S \neq \emptyset, I = \emptyset, O \neq \emptyset, \delta: S \rightarrow S, \lambda: S \rightarrow O$
- **Loogikafunktsioon** – $M = (I, O, \lambda)$
 - $S = \emptyset, I \neq \emptyset, O \neq \emptyset, \delta = \emptyset, \lambda: I \rightarrow O$
- **Mikroprogramm automaat** – $M = (S, I, O, \delta, \lambda)$
 - $S \neq \emptyset, I = \{0,1\}^L, O = \{0,1\}^M, \delta: S \times I \rightarrow S, \lambda: S \times I \rightarrow O$



Esitusviisid

- **Tabel**
 - **veerud:**
sisend (i_t), jooksev olek (s_t),
väljund (o_t), uus olek (s_{t+1})
 - **read:** siire jooksvast olekust uude olekusse:
 $i_t \times s_t \rightarrow o_t \times s_{t+1}$
- **Olekudiagramm, olekugraaf (state graph)**
 - sõlmed: olekud
 - kaared: siirded
- **Siirdediagramm (transition graph)**
 - sõlmed: siirded
 - kaared: olekud

i_t	s_t	o_t	s_{t+1}
jah	hea	5	hea
ei	hea	3	halb
ei	halb	2	halb
jah	halb	4	hea



Automaatide omadusi

- **Osaliselt määratud automaadid**
 - leidub olekuid, kus siire pole mingi sisendi puhul määratud
 - lihtsustatud kirja pilt – vaikinisi jääb nt. samasse olekusse
 - määramatus tuleneb väliskeskkonna iseärasustes – mitte-eksisteerivad sisendkombinatsioonid
 - kahendkodeeritud olekud – osa kombinatsioone on kasutamata
 - automaadi minimeerimisel vabamad käed – osaliselt määratud loogikafunktsioonid
- **Mittedeterministlikud automaadid**
 - leidub olekute ja sisendite kombinatsioone, mille puhul on määratud rohkem kui üks järgmine olek
 - kompaktne meetod kirjeldamiseks, kui leidub rohkem kui üks legaalne reaktsioon mingile sisendkombinatsioonile (jadale)
 - matemaatilised mudelid
- **Isomorfism**
 - üksühene vastavus kahe automaadi komponentide $(S, I, O, \delta, \lambda)$ vahel
- **Homomorfism**
 - ühene vastavus kahe automaadi komponentide $(S, I, O, \delta, \lambda)$ vahel ~ “alam-automaat”

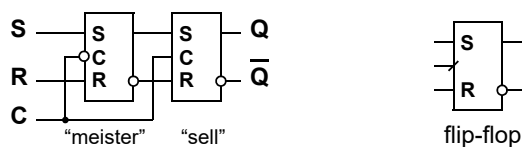
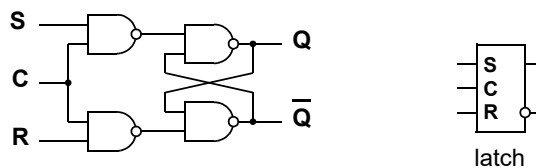
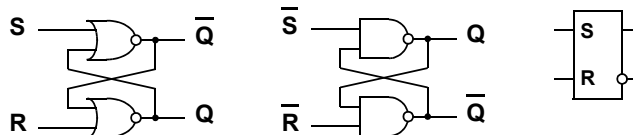
Mäluelemendid

- **Salvestavad olekukoodi**
- **Register**
 - kahendvektori salvestamiseks
 - sama tüüpi mäluelemendid
- **Mäluelementide tüübid**
 - funktsionaalsus – SR, JK, D- ja T-trigerid
 - takteerimine
 - asünkroone – takt puudub
 - latch (lukk-register/-triger) – läbipaistev kui takt on aktiivne
 - flip-flop (frondist aktiveeruv triger) – väljundis muutus ainult taktsignaali frondi korral
 - master-slave (meister-sell) – kaks järjestikust lukk-registrit (latch'i)
 - frondile reageerivad trigerid – spetsiaalne sise-ehitus

SR-triger (set/reset)

S	R	Q_{t+1}	$\bar{Q}_{t+1}$
0	0	Q_t	$\bar{Q}_t$
0	1	0	1
1	0	1	0
1	1	?	?

Q_t	Q_{t+1}	S	R
0	0	0	-
0	1	1	0
1	0	0	1
1	1	-	0



JK-triger

J	K	Q_{t+1}	$\bar{Q}_{t+1}$
0	0	Q_t	$\bar{Q}_t$
0	1	0	1
1	0	1	0
1	1	$\bar{Q}_t$	Q_t

Q_t	Q_{t+1}	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0



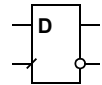
- Määramatus võimaldab loogikafunktsioone efektiivsemalt minimeerida
- Kaks sisendit → kaks loogikafunktsiooni

D-triger (delay)

- Kõige enam kasutusel
 - lihtne sise-ehitus
 - väike sisendite arv → vähem loogikafunktsioone

D	Q_{t+1}	$\bar{Q}_{t+1}$
0	0	1
1	1	0

Q_t	Q_{t+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

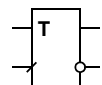


T-triger

- Sobiv loendurites kasutamiseks

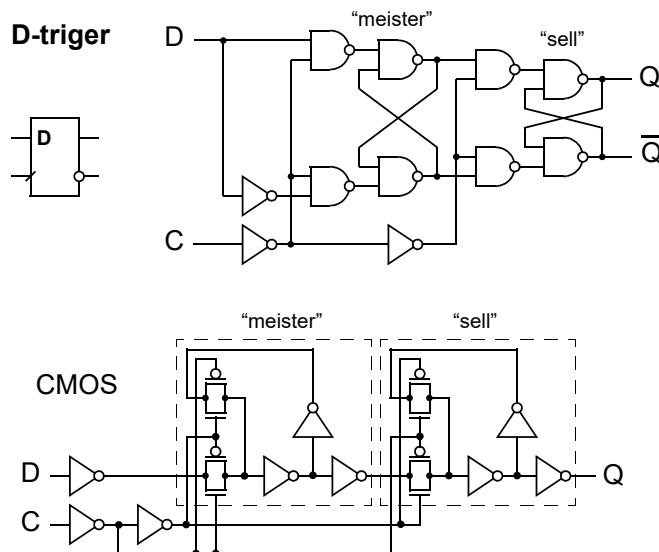
T	Q_{t+1}	$\bar{Q}_{t+1}$
0	Q_t	$\bar{Q}_t$
1	$\bar{Q}_t$	Q_t

Q_t	Q_{t+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

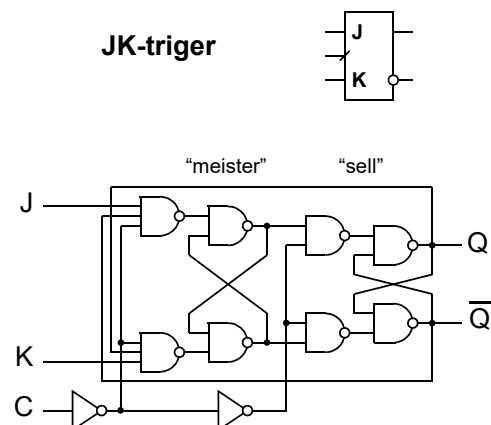


Trigerite sise-ehitus

D-triger



JK-triger

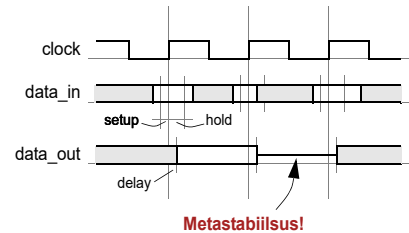


Trigerite simulaatorid

- <http://www.falstad.com/circuit/> – Sequential Logic → Flip-Flops → ...
- <http://tams-www.informatik.uni-hamburg.de/applets/cmos/cmosdemo.html>
vt. CMOS D-latch

Trigerite ajalised parameetrid

- Sisemised ahelad erinevate viidetega
 - nt. sisendist mäluelementideni võrreldes tagasisidega
 - seadeaeg (setup) – nõutav valmisoleku aeg
 - sisend peab olema stabiilne mingi aeg enne taktsignaali aktiivset fronti
 - hoideaeg (hold) – nõutav stabiilsuse aeg
 - sisend peab olema stabiilne mingi aeg pärast taktsignaali aktiivset fronti
 - nõuete rikkumise korral metastabiilsuse oht – väljundi '0' ja '1' vahel

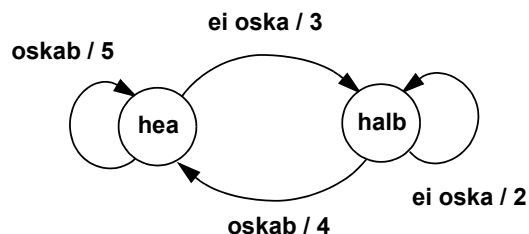


Juhtautomaatide süntees

- **Automaadi olekudiagrammi / tabeli genereerimine (süntees)**
 - tabeli süntees plokk-skeemist (plokk-diagrammist)
 - plokk-skeemi genereerimine kõrgtaseme keeltest
 - riistvara kirjelduskeel (hardware description language)
 - vt. automaadi näiteid VHDL-i loengumaterjalides
- **Automaadi süntees olekudiagrammist / tabelist**
 - eesmärk – automaadi efektiivne realisatsioon
 - suurus, kiirus, energiatarve, testitavus jne.
 - sisendite / väljundite kodeerimine
 - olekute kodeerimine
 - siirde- ja väljundfunktsiooni süntees ja minimeerimine
 - sõltub valitud mäluelementide tüübist (tüüpidest)

Tabel → skeem – Mealy automaat

<i>sisend</i>	<i>olek</i>	<i>väljund</i>	<i>uus olek</i>
tudeng	õppejõud	hinne	õppejõud
oskab	hea tuju	5	hea tuju
ei oska	hea tuju	3	halb tuju
ei oska	halb tuju	2	halb tuju
oskab	halb tuju	4	hea tuju



- Sisendite kodeerimine: oskab – “1”, ei oska – “0”
- Väljundite kodeerimine: 5 – “11”, 4 – “10”, 3 – “01”, 2 – “00”
- Olekute kodeerimine: hea tuju – “1”, halb tuju – “0”

Kodeeritud tabel

<i>sisend</i>		<i>olek</i>		<i>väljund</i>		<i>uus olek</i>	
tudeng	i_t	õppejõud	s_t	hinne	o_t	õppejõud	s_{t+1}
oskab	1	hea tuju	1	5	11	hea tuju	1
ei oska	0	hea tuju	1	3	01	halb tuju	0
ei oska	0	halb tuju	0	2	00	halb tuju	0
oskab	1	halb tuju	0	4	10	hea tuju	1

- Ainult koodid vajalikud
- Siirded sorteeritud (lähte)olekute järgi

i_t	s_t	s_{t+1}	o_t
0	0	0	00
1	0	1	10
0	1	0	01
1	1	1	11

Funktsioonide süntees

- Kodeeritud tabel
- Mäluelemendi tabel
 - vana olek & uus olek → sisendkombinatsioon(id)
- D-trigger

i_t	s_t	s_{t+1}	o_t
0	0	0	00
1	0	1	10
0	1	0	01
1	1	1	11

Q_t	Q_{t+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

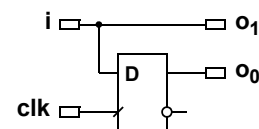
- Funktsioonide minimeerimine

	i
s	0 1
	0 1
	d_0

	i
s	0 0
	1 1
	o_0

	i
s	0 1
	0 1
	o_1

$$\begin{aligned} d_0 &= i \\ o_0 &= s \\ o_1 &= i \end{aligned}$$



Funktsioonide süntees (variant)

- Kodeeritud tabel
- Mäluelemendi tabel
 - vana olek & uus olek → sisendkombinatsioon(id)
- JK-trigger

i_t	s_t	s_{t+1}	o_t
0	0	0	00
1	0	1	10
0	1	0	01
1	1	1	11

Q_t	Q_{t+1}	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

- Funktsioonide minimeerimine

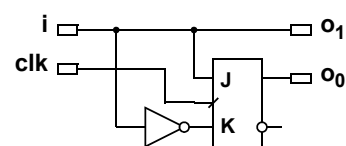
	i
s	0 1
	- -
	j_0

	i
s	- -
	1 0
	k_0

	i
s	0 0
	1 1
	o_0

	i
s	0 1
	0 1
	o_1

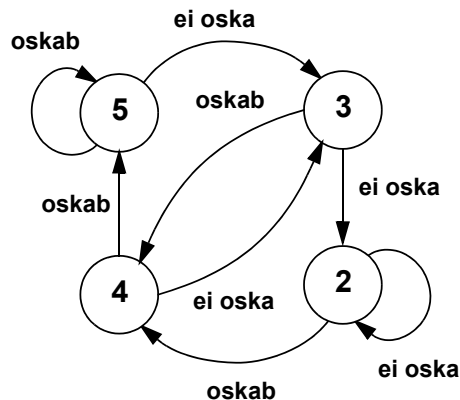
$$\begin{aligned} j_0 &= i \\ k_0 &= \bar{i} \\ o_0 &= s \\ o_1 &= i \end{aligned}$$



Moore automaadi süntees

- Põhimõtteliselt sama
 - olekuid rohkem
 - väljundfunktsioon lihtsam

sisend	olek	väljund	uus olek
ei oska	halb tuju & 2	2	halb tuju & 2
oskab			hea tuju & 4
ei oska	halb tuju & 3	3	halb tuju & 2
oskab			hea tuju & 4
ei oska	hea tuju & 4	4	halb tuju & 3
oskab			hea tuju & 5
ei oska	hea tuju & 5	5	halb tuju & 3
oskab			hea tuju & 5



Moore automaadi süntees (järg)

- Kodeerimine
 - sisend ja väljundkoodid samad
 - olekud – halb tuju & 2 – “00”, halb tuju & 3 – “01”, hea tuju & 4 – “10”, hea tuju & 5 – “11”

i_t	s_t	s_{t+1}	o_t
0	00	00	00
1		10	
0	01	00	01
1		10	
0	10	01	10
1		11	
0	11	01	11
1		11	

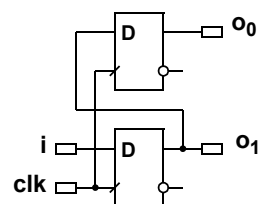
Q_t	Q_{t+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

- Funktsioonide minimeerimine

	s_0	s_1
i	0	0
	1	1
	d_0	

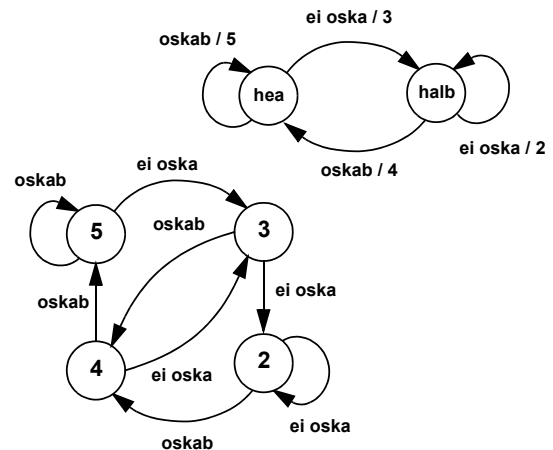
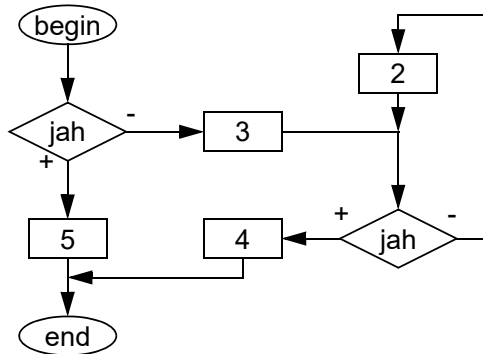
	s_0	s_1
i	0	0
	1	1
	d_1	

$$\begin{aligned} d_0 &= s_1 \\ d_1 &= i \\ o_0 &= s_0 \\ o_1 &= s_1 \end{aligned}$$

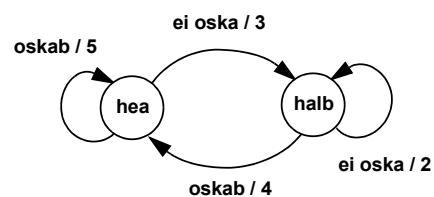
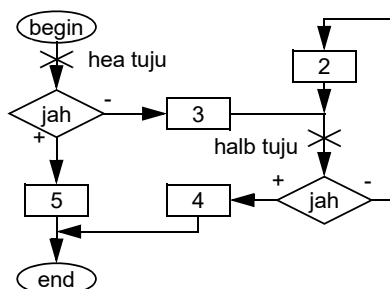


Tabeli / olekudiagrammi süntees plokskeemist

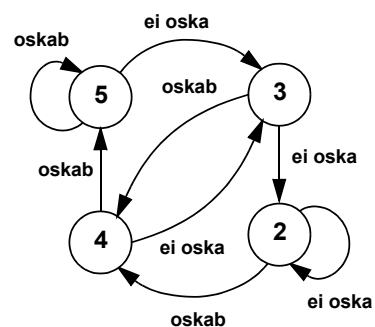
- Algoritmi graaf-skeem (GSA)
- Behavioral FSM



Tabeli / olekudiagrammi süntees plokskeemist – reeglid



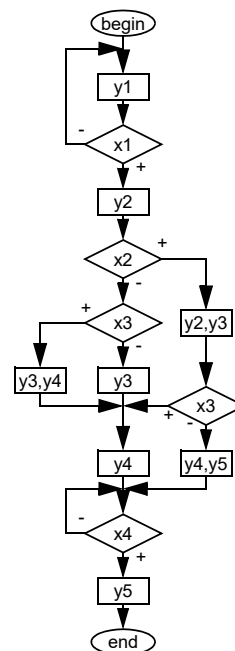
- Mealy automaat
 - kahe väljundplokki vahel peab olema olek
 - tsüklil peab sisaldama olekut
- Moore automaat
 - väljundplokk on olek
 - tsüklil peab sisaldama olekut



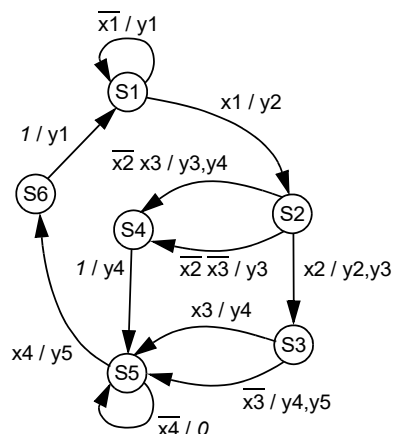
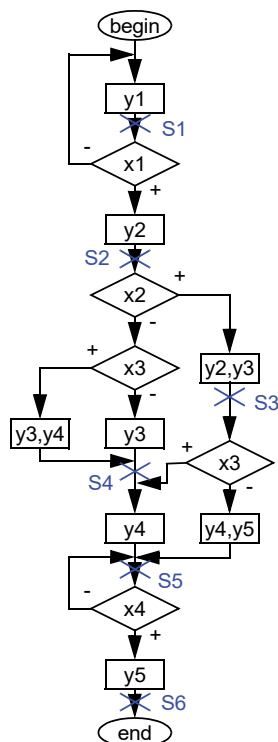
Plokskeemi genereerimine

- Automaatne genereerimine kõrgtaseme kirjeldusest
 - programmeerimiskeel või riistvara kirjelduskeel

```
process fsm (x1,x2,x3,x4,y1,y2,y3,y4,y5)
  bit in x1,x2,x3,x4; bit out y1,y2,y3,y4,y5;
{
  while (!x1) out (y1);
  out (y2);
  if (x2) {
    out (y2,y3);
    if (!x3) { out (y4,y5); goto L1; }
  }
  else { if (x3) out (y3,y4); else out (y3); }
  out (y4);
L1: while (!x4);
  out (y5);
}
```



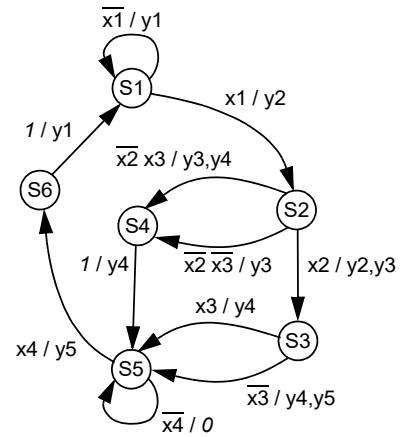
Mealy automaat



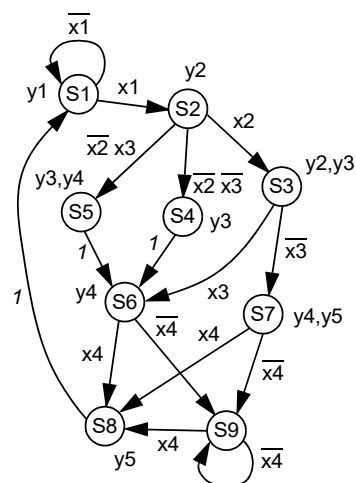
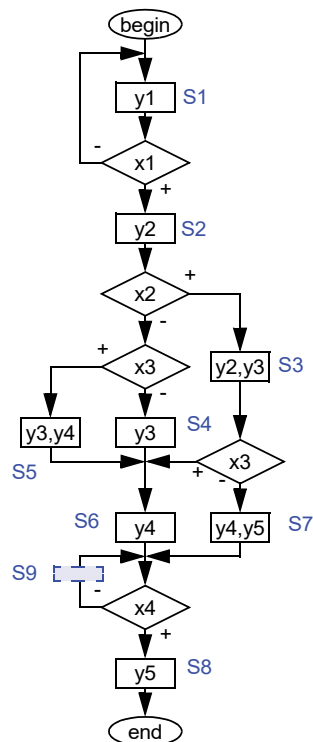
Mealy automaat – tabel

i^t	s^t	s^{t+1}	o^t
$\overline{x1}$	S1	S1	y1
x1		S2	y2
$\overline{x2} \ \overline{x3}$	S2	S4	y3
$\overline{x2} \ x3$		S4	y3, y4
x2		S3	y2, y3
$\overline{x3}$	S3	S5	y4, y5
x3		S5	y4
1	S4	S5	y4
$\overline{x4}$	S5	S5	0
x4		S6	y5
1	S6	S1	y1

i^t	s^t	s^{t+1}	o^t
0---	S1	S1	10000
1---		S2	01000
-00-	S2	S4	00100
-01-		S4	00110
-1--		S3	01100
--0-	S3	S5	00011
--1-		S5	00010
----	S4	S5	00010
---0	S5	S5	00000
---1		S6	00001
----	S6	S1	10000



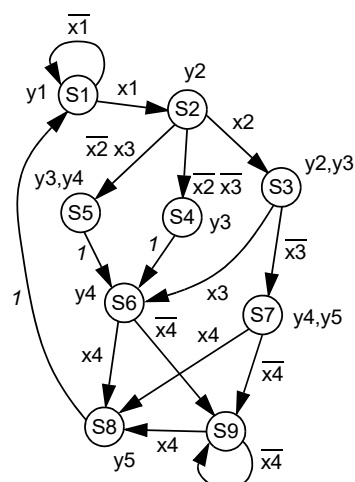
Moore automaat



Moore automaat – tabel

i^t	s^t	s^{t+1}	o^t
$\overline{x1}$	S1	S1	y1
x1		S2	
$\overline{x2} \overline{x3}$	S2	S4	y2
$\overline{x2} x3$		S5	
x2		S3	
$\overline{x3}$	S3	S7	y2, y3
x3		S6	
1	S4	S6	y3
1	S5	S6	y3, y4
$\overline{x4}$	S6	S9	y4
x4		S8	
$\overline{x4}$	S7	S9	y4, y5
x4		S8	
1	S8	S1	y5
$\overline{x4}$	S9	S9	0
x4		S8	

i^t	s^t	s^{t+1}	o^t
0---	S1	S1	10000
1---		S2	
-00-	S2	S4	01000
-01-		S5	
-1--		S3	
--0-	S3	S7	01100
--1-		S6	
----	S4	S6	00100
----	S5	S6	00110
----0	S6	S9	00010
----1		S8	
---0	S7	S9	00011
---1		S8	
----	S8	S1	00001
---0	S9	S9	00000
---1		S8	



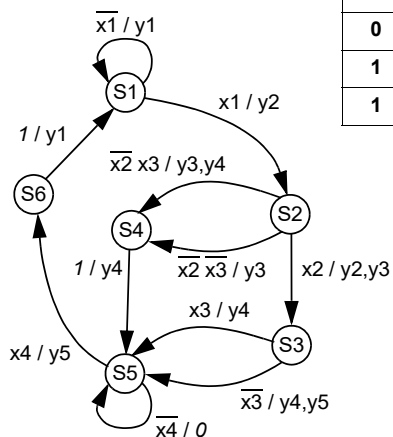
Olekute kodeerimine

- Leida olekutele sellised koodid, et siirde- ja väljundfunktsioonid oleksid minimaalsed
- osaliselt määratud funktsioonid aitavad sellele kaasa
- Heuristilised reeglid
 - Kõik mäluelemendid
 - olekud, mis samade sisendtingimuste korral viivad samasse olekusse, peaksid olema naaberkoodidega – võimaldab kaks konjunktsiooni kokku kleepida
 - D-trigerid – mida rohkem siirdeid mingisse olekusse, seda vähem peaks selle kood sisaldama 1
 - iga siire vastab ühele konjunktsioonile
 - olek, millesse toimub kõige rohkem siirdeid, peaks olema ainult 0-dest koosneva koodiga
 - SR-, JK- & T-trigerid – siirdele vastava jooksva ja järgneva oleku koodid peaksid võimalikult kokku langema
 - ideaaljuhul naaberkoodid
 - sarnane sobivusgraafi servakatte leidmise ülesandega

Mealy automaat – olekute kodeerimine

i^t	s^t	s^{t+1}	o^t
0---	S1	000	S1 000 10000
1---			S2 100 01000
-00-	S2	100	S4 110 00100
-01-			S4 110 00110
-1--			S3 101 01100
--0-	S3	101	S5 111 00011
--1-			S5 111 00010
----	S4	110	S5 111 00010
---0	S5	111	S5 111 00000
---1			S6 011 00001
----	S6	011	S1 000 10000

Q_t	Q_{t+1}	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0



- Naabrid: S1-S2, S2-S3, S2-S4, S3-S5, S4-S5, S5-S6, S6-S1

Mealy automaat – funktsioonide süntees

i^t	s^t	s^{t+1}	JK^{t+1}	o^t
0---	S1	000	S1 000 0- 0- 0-	10000
1---			S2 100 1- 0- 0-	01000
-00-	S2	100	S4 110 -0 1- 0-	00100
-01-			S4 110 -0 1- 0-	00110
-1--			S3 101 -0 0- 1-	01100
--0-	S3	101	S5 111 -0 1- -0	00011
--1-			S5 111 -0 1- -0	00010
----	S4	110	S5 111 -0 -0 1-	00010
---0	S5	111	S5 111 -0 -0 -0	00000
---1			S6 011 -1 -0 -0	00001
----	S6	011	S1 000 0- -1 -1	10000

Q_t	Q_{t+1}	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

JK^{t+1} - JK
trigerite
sisendid

Kasutamata
olekukoodid:
001, 010

Mealy automaat – funktsioonide minimeerimine

funktsioonid

```
xxxx qq | jkjkjk yyy
1234 123 | 112233 12345
-----+-----
0--- 000 | 0-0-0- 10000
1--- 000 | 1-0-0- 01000
-00- 100 | -01-0- 00100
-01- 100 | -01-0- 00110
-1-- 100 | -00-1- 01100
--0- 101 | -01--0 00011
--1- 101 | -01--0 00010
---- 110 | -0-01- 00010
---0 111 | -0-0-0 00000
---1 111 | -1-0-0 00001
---- 011 | 0--1-1 10000
---- 001 | -----
---- 010 | -----
```

espresso sisend

```
.i 7
.o 11
0---000 0-0-0-10000
1---000 1-0-0-01000
-00-100 -01-0-00100
-01-100 -01-0-00110
-1--100 -00-1-01100
--0-101 -01--000011
--1-101 -01--000010
----110 -0-01-00010
---0111 -0-0-000000
---1111 -1-0-000001
----011 0--1-110000
----001 -----
----010 -----
.e
```

espresso väljund

```
.i 7
.o 11
#.phase 11011101101
---1111 01000000001
--0--00 00000000010
--0--01 00000000001
1---0-0 10000011000
----100 00000000100
-----11 00000000010
-1--100 00101001010
-----1- 00001000000
----1-- 00000010000
----0-- 00110100010
.e
```

Mealy automaat – skeem

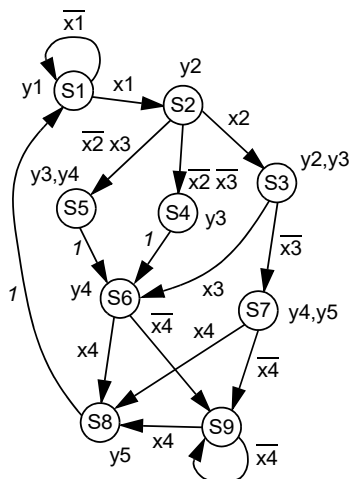
espresso väljund

```
.i 7
.o 11
#.phase 11011101101
---1111 01000000001
--0--00 00000000010
--0--01 00000000001
1---0-0 10000011000
----100 00000000100
-----11 00000000010
-1--100 00101001010
-----1- 00001000000
----1-- 00000010000
----0-- 00110100010
.e
```

- $j1 = x1 \overline{q1} \overline{q3}$
- $k1 = x4 q1 q2 q3$
- $\overline{j2} = x2 q1 \overline{q2} \overline{q3} + \overline{q1}$
- $k2 = k3 = \overline{q1}$
- $j3 = x2 q1 \overline{q2} \overline{q3} + q2$
- $y1 = j1 + q1$
- $y2 = j1 + x2 q1 \overline{q2} \overline{q3}$
- $y3 = q1 \overline{q2} \overline{q3}$
- $y4 = \overline{x3} \overline{q2} \overline{q3} + q2 q3 + x2 q1 \overline{q2} \overline{q3} + \overline{q1}$
- $y5 = k1 + x3 \overline{q2} q3$

i^t	s^t	s^{t+1}	o^t
0---	S1	0000	10000
1---		S2	0011
-00-	S2	0011	01000
-01-		S5	0101
-1--		S3	0110
--0-	S3	0110	01100
--1-		S6	0001
----	S4	1100	00100
----	S5	0101	00110
---0	S6	0001	00010
---1		S8	0010
---0	S7	1000	00011
---1		S8	0010
----	S8	0010	00001
---0	S9	0100	00000
---1		S8	0010

Moore automaat – olekute kodeerimine



Q_t	Q_{t+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

Siirete arv:
 S1 - 2, S2 - 1,
 S3 - 1, S4 - 1,
 S5 - 1, S6 - 3,
 S7 - 1, S8 - 3,
 S9 - 3

i ^t	s ^t		s ^{t+1}		D ^{t+1}	o ^t
0---	s1	0000	s1	0000	0000	10000
1---			s2	0011	0011	
-00-	s2	0011	s4	1100	1100	01000
-01-			s5	0101	0101	
-1--			s3	0110	0110	
--0-	s3	0110	s7	1000	1000	01100
--1-			s6	0001	0001	
----	s4	1100	s6	0001	0001	00100
----	s5	0101	s6	0001	0001	00110
---0	s6	0001	s9	0100	0100	00010
---1			s8	0010	0010	
---0	s7	1000	s9	0100	0100	00011
---1			s8	0010	0010	
----	s8	0010	s1	0000	0000	00001
---0	s9	0100	s9	0100	0100	00000
---1			s8	0010	0010	

Moore automaat – funktsioonide süntees

Q_t	Q_{t+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

D^{t+1} - D-trigerite sisendid

Kasutamata olekukoodid:
 0111, 1001, 1010,
 1011, 1101, 1110,
 1111

xxxx qqqq dddd 1234 1234 1234	siirde- funktsioon		.i 8 .o 4	siirde- funktsioon
-----+-----		espresso sisend:	0--0000 0000	
0--- 0000 0000			1--0000 0011	
1--- 0000 0011			-00-0011 1100	
-00- 0011 1100	väljund- funktsioon		-01-0011 0101	väljund- funktsioon
-01- 0011 0101			-1--0011 0110	
-1-- 0011 0110	qqqq yyyy 1234 12345		--0-0110 1000	.i 4
--0- 0110 1000	-----+-----		--1-0110 0001	.o 5
--1- 0110 0001	0000 10000		----1100 0001	0000 10000
---- 1100 0001	0011 01000		----0101 0001	0011 01000
---- 0101 0001	0110 01100		---00001 0100	0110 01100
---1 0001 0010	1100 00100		---10001 0010	1100 00100
---0 0001 0100	0101 00110		---01000 0100	0101 00110
---1 1000 0010	0001 00010		---11000 0010	0001 00010
---0 1000 0100	1000 00011		---00100 0000	1000 00011
---0 0010 0000	0010 00001		---00100 0100	0010 00001
---1 0100 0010	0100 00000		---10100 0010	0100 00000
---0 0100 0100	0111 -----		----0111 ----	0111 ----
---- 0111 -----	1001 -----		----1001 ----	1001 ----
---- 1001 -----	1010 -----		----1010 ----	1010 ----
---- 1010 -----	1011 -----		----1011 ----	1011 ----
---- 1011 -----	1101 -----		----1101 ----	1101 ----
---- 1101 -----	1110 -----		----1110 ----	1110 ----
---- 1110 -----	1111 -----		----1111 ----	1111 ----
---- 1111 -----			.e	.e

.i 8 .o 4	siirde- funktsioon		
0--0000 0000		espresso sisend	
1--0000 0011			
-00-0011 1100	väljund- funktsioon		
-01-0011 0101			
-1--0011 0110			
--0-0110 1000	.i 4		
--1-0110 0001	.o 5		
----1100 0001	0000 10000		
----0101 0001	0011 01000		
---00001 0100	0110 01100		
---10001 0010	1100 00100		
---01000 0100	0101 00110		
---11000 0010	0001 00010		
----0010 0000	1000 00011		
---00100 0100	0010 00001		
---10100 0010	0100 00000		
----0111 ----	0111 ----		
----1001 ----	1001 ----		
----1010 ----	1010 ----		
----1011 ----	1011 ----		
----1101 ----	1101 ----		
----1110 ----	1110 ----		
----1111 ----	1111 ----		
.e	.e		

Moore automaat – funktsioonide minimeerimine	
	espresso väljund
.i 8 .o 4	siirde- funktsioon
#.phase 0011	
1--0000 0011	väljund- funktsioon
---10100 0110	
-01---11 1001	.i 4
---1-001 0110	.o 5
-1---11 1010	#.phase 10011
--1--11- 1001	0000 10000
---110-- 0110	0-00 00100
----00-0 1100	-010 01001
-----10 0100	--01 00010
----11-- 0101	10-- 00011
-----1-1 0101	--0- 01000
-----0- 1000	-0-- 00100
.e	.e

Moore automaat – skeem

```
.i 8
.o 4
#.phase 0011
1---0000 0011
---10100 0110
-01---11 1001
---1-001 0110
-1---11 1010
--1--11- 1001
---110-- 0110
-----00-0 1100
-----10 0100
----11-- 0101
-----1-1 0101
-----0- 1000
.e
```

siirde-
funktsioon

$$\begin{aligned} \overline{d1} &= \overline{x2} \ x3 \ q3 \ q4 + \overline{x2} \ \overline{q3} \ q4 + \\ &\quad + x3 \ q2 \ q3 + \overline{q1} \ \overline{q2} \ \overline{q4} + \overline{q3} \\ \overline{d2} &= x4 \ \overline{q1} \ \overline{q2} \ \overline{q3} \ \overline{q4} + x4 \ \overline{q2} \ \overline{q3} \ \overline{q4} + \\ &\quad + x4 \ q1 \ q2 + \overline{q1} \ \overline{q2} \ \overline{q4} + \overline{q3} \ \overline{q4} + \\ &\quad + \overline{q1} \ q2 \ q4 \\ \overline{d3} &= x1 \ \overline{q1} \ \overline{q2} \ \overline{q3} \ \overline{q4} + \\ &\quad + x4 \ \overline{q1} \ \overline{q2} \ \overline{q3} \ \overline{q4} + x4 \ \overline{q2} \ \overline{q3} \ q4 + \\ &\quad + x2 \ q3 \ q4 + x4 \ q1 \ \overline{q2} \\ \overline{d4} &= x1 \ \overline{q1} \ \overline{q2} \ \overline{q3} \ \overline{q4} + \overline{x2} \ x3 \ q3 \ q4 + \\ &\quad + x3 \ q2 \ q3 + \overline{q1} \ q2 + \overline{q2} \ q4 \end{aligned}$$

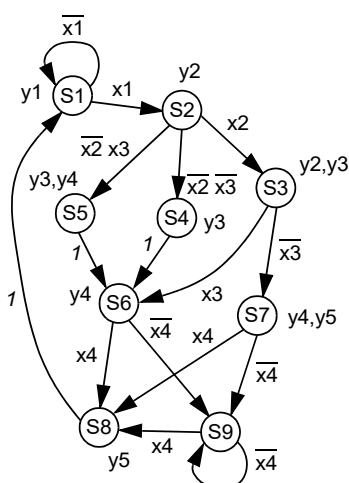
```
.i 4
.o 5
#.phase 10011
0000 10000
0-00 00100
-010 01001
--01 00010
10-- 00011
--0- 01000
-0-- 00100
.e
```

väljund-
funktsioon

$$\begin{aligned} y1 &= \overline{q1} \ \overline{q2} \ \overline{q3} \ \overline{q4} \\ y2 &= \overline{q2} \ q3 \ \overline{q4} + \overline{q3} \\ y3 &= \overline{q1} \ \overline{q3} \ \overline{q4} + \overline{q2} \\ y4 &= \overline{q3} \ q4 + \overline{q1} \ \overline{q2} \\ y5 &= \overline{q2} \ q3 \ \overline{q4} + \overline{q1} \ \overline{q2} \end{aligned}$$

Moore automaat – olekute kodeerimine #2

i^t	s^t	s^{t+1}	o^t
0---	S1	0000	S1 0000 10000
1---		S2 0001	
-00-	S2	0001	S4 0101 01000
-01-		S5 0011	
-1--		S3 1001	
--0-	S3	1001	S7 1010 01100
--1-		S6 0110	
----	S4	0101	S6 0110 00100
----	S5	0011	S6 0110 00110
---0	S6	0110	S9 1110 00010
---1		S8 1100	
---0	S7	1010	S9 1110 00011
---1		S8 1100	
----	S8	1100	S1 0000 00001
---0	S9	1110	S9 1110 00000
---1		S8 1100	



Q_t	Q_{t+1}	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

Võimalikult vähe
muutuvad koodid:
S1 - 0000, S2 - 0001,
S3 - 1001, S4 - 0101,
S5 - 0011, S6 - 0110,
S7 - 1010, S8 - 1100,
S9 - 1110

Moore automaat – funktsioonide süntees #2

i ^t	s ^t		s ^{t+1}		JK ^{t+1}	o ^t
0---	S1	0000	S1	0000	0-0-0-0-	10000
1---			S2	0001	0-0-0-1-	
-00-	S2	0001	S4	0101	0-1-0--0	01000
-01-			S5	0011	0-0-1--0	
-1--			S3	1001	1-0-0--0	
--0-	S3	1001	S7	1010	-00-1--1	01100
--1-			S6	0110	-11-1--1	
----	S4	0101	S6	0110	0--01--1	00100
----	S5	0011	S6	0110	0-1--0-1	00110
---0	S6	0110	S9	1110	1--0-00-	00010
---1			S8	1100	1--0-10-	
---0	S7	1010	S9	1110	-01--00-	00011
---1			S8	1100	-01--10-	
----	S8	1100	S1	0000	-1-10-0-	00001
---0	S9	1110	S9	1110	-0-0-00-	00000
---1			S8	1100	-0-0-10-	

Q_t	Q_{t+1}	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

JK^{t+1} - JK-trigerite sisendid

Kasutamata olekukoodid:
0010, 0100, 0111, 1000, 1011, 1101, 1111

Moore automaat – funktsioonide minimeerimine #2

```
.i 8
.o 8
0---0000 0-0-0-0-
1---0000 0-0-0-1-
-00-0001 0-1-0--0
-01-0001 0-0-1--0
-1--0001 1-0-0--0
--0-1001 -00-1--1
--1-1001 -11-1--1
----0101 0--01--1
----0011 0-1--0-1
---00110 1--0-00-
---10110 1--0-10-
---01010 -01--00-
---11010 -01--10-
----1100 -1-10-0-
---01110 -0-0-00-
---11110 -0-0-10-
----0010 -----
----0100 -----
----0111 -----
----1000 -----
----1011 -----
----1101 -----
----1111 -----
.e
```

siirde-funktsioon
espresso sisend

väljund-funktsioon

espresso väljund

```
siirde-funktsioon
.i 8
.o 8
#.phase 11110110
-00-00-1 00101000
-1--0001 10001000
--1-1--1 01100000
1---00- 00000010
----000- 00000001
---1---0 00000100
-----10 10000000
-----00 01011000
-----1- 00100000
.e

väljund-funktsioon
.i 4
.o 5
#.phase 00010
0-1- 00010
000- 00101
-11- 00001
-01- 11010
---0 01100
---1 10001
-1-- 11000
.e
```

Moore automaat – skeem #2

<pre>.i 8 .o 8 #.phase 11110110 -00-00-1 00101000 -1--0001 10001000 --1-1--1 01100000 1----00- 00000010 ----000- 00000001 ---1---0 00000100 -----10 10000000 -----00 01011000 -----1- 00100000 .e</pre>	siirde-funktsioon $j1 = x2 \overline{q1} \overline{q2} \overline{q3} \overline{q4} + q3 \overline{q4}$ $k1 = \overline{x3} \overline{q1} \overline{q4} + \overline{q3} \overline{q4}$ $j2 = \overline{x2} \overline{x3} \overline{q1} \overline{q2} \overline{q4} + \overline{x3} \overline{q1} \overline{q4} + q3$ $k2 = \overline{q3} \overline{q4}$ $j3 = \overline{x2} \overline{x3} \overline{q1} \overline{q2} \overline{q4} + \overline{x2} \overline{q1} \overline{q2} \overline{q3} \overline{q4} + \overline{q3} \overline{q4}$ $k3 = \overline{x4} \overline{q4}$ $j4 = \overline{x1} \overline{q2} \overline{q3}$ $k4 = \overline{q1} \overline{q2} \overline{q3}$	<pre>.i 4 .o 5 #.phase 00010 0-1- 00010 000- 00101 -11- 00001 -01- 11010 ---0 01100 ---1 10001 -1-- 11000 .e</pre> väljund-funktsioon $\overline{y1} = \overline{q2} \overline{q3} + \overline{q4} + q2$ $\overline{y2} = \overline{q2} \overline{q3} + \overline{q4} + q2$ $\overline{y3} = \overline{q1} \overline{q2} \overline{q3} + \overline{q4}$ $\overline{y4} = \overline{q1} \overline{q3} + \overline{q2} \overline{q3}$ $\overline{y5} = \overline{q1} \overline{q2} \overline{q3} + q2 \overline{q3} + q4$
---	---	--

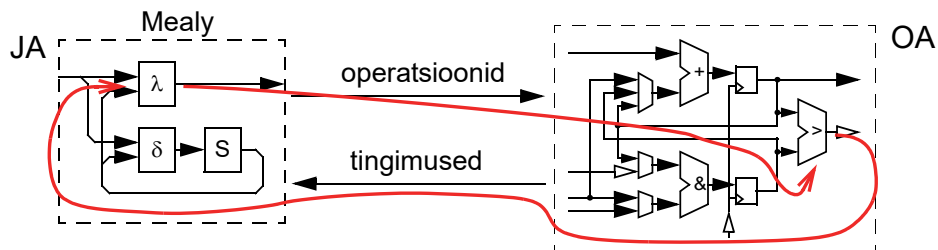
Olekute kodeerimise olulisus – võrdlus

Automaat	Kodeering #1	Kodeering #2	Kodeering #3
x s s+ y	"loendur"	D-trigeri heuristika	1,2,3,4,5,6
0 1 4 0			1,6,2,5,3,4
1 1 3 0			
0 2 6 0	1:000 2:001	1: 1 2: 2	1:110 2:101
1 2 3 0	3:010 4:011	3: 3 4: 3	3:100 4:000
0 3 5 0	5:100 6:101	5: 2 6: 1	5:001 6:010
1 3 2 0			
0 4 2 1	faas 1011	ülesanne	faas 1111
1 4 5 1	0001 1100	0110 0010	-000 0001
0 5 1 0	000- 0010	1110 0000	00-- 1000
1 5 4 0	01-0 0100	0010 1010	11-- 1000
0 6 3 0	11-- 0010	1010 0000	0--1 0100
1 6 4 0	1-10 0110	0000 1000	--00 0010
	0-10 1100	1000 0100	
	1-11 1101	0001 0101	
	0-11 0111	1001 1001	
		0100 1100	
		1100 0010	
		0101 0000	
		1101 0010	
		-011 ----	
		-111 ----	
			$y = \overline{q1} \overline{q2} \overline{q3}$
			$d_1 = \overline{x} \overline{q1} + x q_1$
			$d_2 = \overline{x} q_3$
			$d_3 = \overline{q2} \overline{q3}$



Mealy või Moore?

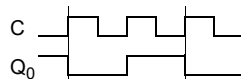
- **Mealy**
 - vähem olekuid → vähem mälulemente
 - keerulisemad väljundfunktsioonid
 - puhverdamata väljundid → asünkroonse tagasiside oht
- **Moore**
 - rohkem olekuid → rohkem mälulemente
 - lihtsamad väljundfunktsioonid
 - “puhverdatud” väljundid



Loendur kui automaat ...

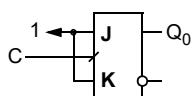
- Loendur – primitiivne generaator – $S \neq \emptyset$, $I = \emptyset$, $O = \emptyset$ ($O \equiv S$), $\delta: S \rightarrow S$, $\lambda \equiv \emptyset$

2-nd loendur
Jada – 0, 1

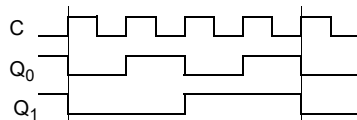


s_t	s_{t+1}	JK_{t+1}
0	1	1-
1	0	-1

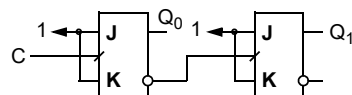
$J=K=1$



4-nd loendur
Jada – 0, 1, 2, 3



Kaks loendurit järjest?



Takti hilistumine!!!

Automaat?
Olekud – 00, 01, 10, 11 [Q1 Q0]

JK-triger



Q_t	Q_{t+1}	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

s_t	s_{t+1}	JK_{t+1}
00	01	0- 1-
01	10	1- -1
10	11	-0 1-
11	00	-1 -1

$J_0=K_0=1$

$J_1=K_1=Q_0$

