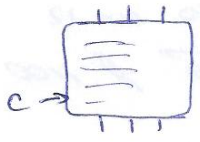
Procesoare

Arhitectura MIPS - ilustrare

Arhitecturi concrete - CISC (complex) - puține - durează mult - domeniu larg aplicații
- RISC (reduced) - multe - durează puțin - gamă restrânsă de prog



short - 16 biți
int
long - 32 biți

long a, b, c ; c=a+b;

Arhitectura intel este o arhitectură CISC (PC, laptop)

Procesoare RISC (calculatoare de la bordul avionului, telefoane, frigidere etc.) - MIPS (console playstation, notebook cu linux (Letux))

Limbajul de asamblare - MIPS - apropiat de hard
- mai înalt decât cel de mașină (hexa)

Mașină virtuală: PCSPIM (de instalat) cu procesor MIPS

Cartea: Peterson și ~~HANASY~~: Organizarea calc. pe interfața hardware - software / Bulevardul Brașov - B-dul Timișoara către, junglă Plaza ; depozit de carne Valea Cascadelor - materiale de construcții

Varianta engleză + CD

Examen scris - probleme

Laborator - proiect - MIPS → o probl. pt 5 ; 3 probl - nota 10 + intr, rulat etc.
- test după anul nou

d17d.txt - reprezentarea în calculator

Materiale - carte
- arhive
- slide

unsigned char - nr nat pe 8 biți

```
unsigned char x;
for (x=0; x<256; ++x) {...}
```

se repetă la infinit
• nr infinit de iterații

Reprezentarea în bază

12 bază 2 : $1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 \longrightarrow 1100$

1100   12 metalimbaj $(12)_2$ eb definit $\overline{1100}_2$ – reprezentarea nr. 12 // $\overline{1100}$ // 1100

Trecerea dintr-o bază într-alta

$( )_{10} \longrightarrow ( )_b$

• **Nr. întregi**: se ia modulul nr. se împ. nr. la bază, câtul la bază .... se iau resturile invers

R.: - împart cu rest ~~...~~ pe |x| la b, câtul la b etc până cât = 0
- iau resturile în ordine inv și le înlocuiesc cu cifre în baza b
- dacă x < 0

Ex: $(4235)_{16} = \overline{108B}$ $(105)_2 = \overline{1101001}$

4235 : 16 = 264, rest 11 (B); 264 : 16 = 16, rest 8; 16 : 16 = 1, rest 0; 1 : 16 = 0, rest 1

B 8 0 1 ←

105	1
52	0
26	0
13	1
6	0
3	1
1	1
0	

• **Nr reale**: ▭ . $c_1 c_2$

$0.f \text{ "} f_0 \text{"} \in [0,1)$

$0.f_0 \times b \in [0, b)$

$c_1 = [0.f_0 \times b] \in \{0, \dots, b-1\}$

$\{0.f_0 \times b\} = 0.f_1 \in [0,1)$

$0.f_1 \times b$ < $[\,] \longrightarrow c_2$; $\{\,\} - 0.f_2$

$f_0 \to c_1$, $f_1 \to c_2$, $f_2 \to c_3$, $f_3 \to c_4$, $f_4 \to c_5$

$(7.8)_5 =$

$(\sqrt{2})_5$ - cazul ...

R: - partea intr $|x|$ se repr ca la nr int

- Part frac $|x|$ se * cu 6 și se ia partea întreagă
- > parte frac $|x|$ se * cu 6
- etc până obt 0

Nr irațional — nu e algoritm, se termină procedeul

- Iau părțile int în ord directă și le înlocuiesc cu cifre în baza 6
- dacă $x < 0$ pun „-" în față

$(7.8)_2 = \overline{111.(1100)}$

7	1
3	1
1	1
0	1

$(7.8)_5 = \overline{12.4}$

7	2
1	1
0	

$0.8 \times 2 = 1.6$
$0.6 \times 2 = 1.2$
$0.2 \times 2 = 0.4$
$0.4 \times 2 = 0.8$
0.8×2

0.8	1
0.6	1
0.2	0
0.4	0
0.8	

$0.8 \times 5 = 4.0$
0×5

$(4.(3))_2 = \overline{100,(01)}$

4	0
2	0
1	1
0	

$0.(3) = \frac{3}{9} = \frac{1}{3}$

$\frac{1}{3} \times 2 = \frac{2}{3} = 0 + \frac{2}{3}$

$\frac{2}{3} \times 2 = \frac{4}{3} = 1 + \frac{1}{3}$

$\frac{1}{3} \times 2$

$(\ )_{10} \rightarrow (\ )_b$; $(\ )_b \rightarrow (\ )_{10}$

Regula $[(\ )_b \rightarrow (\ )_{10}]$
- înlocuiesc cifrele cu nr. care le reprez. și scrierea pozițională cu scrierea polinomială
- facem calculele în baza 10

$(\ )^{-1}$ – inversa
⇒ obț. doar pt nr.

* $(\overline{1101})_2^{-1} = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 13$

$(\overline{A2C})_{16}^{-1} = 10 \cdot 16 + 2 \cdot 16 + 12 \cdot 16 = 2604$

$(\overline{13,4})_5^{-1} = 1 \cdot 5^1 + 3 \cdot 5^0 + 4 \cdot 5^{-1} = \frac{38}{10} = 7,8$

$4/5$

$(\overline{10,1(011)})_2^{-1} = \left(\overline{10} + \frac{\overline{1011} - \overline{1}}{1110}\right)_2^{-1} = (\overline{10})_2^{-1} + \frac{(\overline{1011})_2^{-1} - (\overline{1})_2^{-1}}{(1110)_2^{-1}} =$

$2 + \frac{11-1}{14} = \frac{10}{7} = 2,(714285)$

* $(\ )_{b_1} \rightarrow (\ )_{b_2}$

Regula:
- aduc $(\ )_{b_1}$ în $(\ )_{10}$
- aduc $(\ )_{10}$ în $(\ )_{b_2}$

* $(\ )_{b^k} \rightarrow (\ )_b$; $(\ )_b \rightarrow (\ )_{b^k}$

Regula:
- înloc. fiecare cifră în $(\ )_{b^k}$ cu câte k cifre în baza b
- elimin 0-urile exterioare.

Regula: $(\ )_b \to (\ )_b$
- grupez câte 4 cifre de la virgulă spre stânga și dacă completând eventual cu 0
- înlocuiesc fiecare grup de 4 cifre în baza 2 cu câte o cifră în baza 16

$(1A,B)_{16} = 11010,1011_2$

0001 1010 , 1011

$11011,1011_2 = 1B,A_{16}$

0001 1011 , 1010

1 B , A

$110111001_2 = 0\cdot2^{11}+0\cdot2^{10}+0\cdot2^9+1\cdot2^8+1\cdot2^7+0\cdot2^6+1\cdot2^5+1\cdot2^4+1\cdot2^3+0\cdot2^2+0\cdot2^1+1\cdot2^0 =$

$= (0\cdot2^3+0\cdot2^2+0\cdot2^1+1\cdot2^0)\cdot16^2 + (0\cdot2^3+0\cdot2^2+1\cdot2^1+1\cdot2^0)\cdot16^1 +$

$+ (1\cdot2^3+0\cdot2^2+0\cdot2^1+1\cdot2^0)\cdot16^0$

$1 = 1$, $11 = B$, $9 = 9$

$= 1B9_{16}$

$b=2$

$1011 + 110 = 10001$

$b=16$

$9A - 9B = 17$

$b=2$

$110,11 - 1,001 = 11,011$

$11011 - 11,011 = 11,00111$

6=2

$$10000{,}01 : 11 = 101{,}01(10)$$

```
10000,01 | 11
11       |------------
 100     | 101,01(10)
   0
 100
  11
  101
   11
   (10)0
     11
     10
      0
    (10)
```

Performanța

Perf ≡ t.r. (timp răspuns)
- timpul consumat din calcule proq. ; SO, legat de prog → TUCP (timpul unității centrale) — C (cicli), F (frecvența)
- t. prog așt la resurse
- t cons. act. legat de alte prog

$$Perf. = \frac{1}{\text{Timp de răspuns}} \qquad F = \frac{1}{P(\text{perioada})}$$

X, Y - mașini

$$P_x > P_y \Leftrightarrow T_x < T_y$$
$$\frac{P_x}{P_y} = n \Leftrightarrow \frac{T_y}{T_x} = n \qquad (0)$$

$$TUCP = N_pC \times \Delta_C = \frac{N_rC}{F} \qquad (1)$$

(N_pC – nr cicli; Δ_C – durata ciclului)

1. Un prog durează pe calc A care durează 400 MHz 10 s. Vrem un calc B, F = ? unde prog să dureze 6 sec.

calc A (400 MHz) : 10 s
calc B (?) : 6 s.   1,2 ori mai mulți cicli

$F_A = 400\,MHz$

$T_A = 10\,s$

$T_B = 6\,s$

$\frac{C_B}{C_A} = 1,2$

$F_B = ?$

$$F_B = \frac{C_B}{T_B} = \frac{1,2 \cdot C_A}{T_B} = \frac{1,2 \cdot C_A}{6\,s} =$$

$$= \frac{1,2 \cdot T_A \cdot F_A}{6\,s} = \frac{\overset{0,2}{\cancel{1,2}} \cdot \cancel{10\,s} \cdot 400\,MHz}{\cancel{6\,s}} =$$

$$= 800\,MHz$$

$C_A = F_A \cdot T_A$

$Nr\,C = Nr\,I \times CPI$ (2)

(nr mediu de cicli pe instrucțiune; nr instrucțiuni)

$$TUCP = Nr\,I \times CPI \times \Delta c = \frac{Nr\,I \times CPI}{F}$$ (3) Ecuația elementară a performanței

Clase de instrucțiuni (cls de i)

$CPI_i = CPI$ cls i

C_i = nr de instr executate dintr cls i

$$Nr\,C = \sum_{i=1}^{n} (CPI_i \times C_i)$$ (4)

2. Avem o mașină cu trei cls de instr:

cl	CPI_{cls}
A	1
B	2
C	3

Două programe:

Pr	Nr inst executate dintr fiecare cls (C_i)		
	A	B	C
1	2	1	2
2	4	1	1

a) Care prog execută cele mai multe instr?

b) _____ este mai rapid?

c) CPI pt fiecare prog.

a) $I_1 = 2 + 1 + 2 = 5i$

$I_2 = 4 + 1 + 1 = 6i$

b) $I_1 = 5i$

(4) $\Rightarrow C_1 = 1 \cdot 2 + 1 \cdot 2 + 3 \cdot 2 = 10c$

$C_2 = 1 \cdot 4 + 1 \cdot 2 + 3 \cdot 1 = 9c$ $\Rightarrow$ I_2 este mai rapid

c) $cpi = \frac{C}{I}$

$cpi_1 = \frac{C_1}{I_1} = \frac{10c}{5i} = 2\frac{c}{i} = 2 cpi$

$cpi_2 = \frac{C_2}{I_2} = \frac{9c}{6i} = \frac{3}{2} cpi = 1,5 cpi$ = cel mai bun

$TJ = \frac{Nr\ I}{Timp}$ (frecventa instr.) | $MIPS = \frac{NR\ I}{Timp \times 10^6}$ (5)

Ex

P_r	IC_i		
	A	B	C
1	5	1	1
2	10	1	1

1 masina $-$ 500 MHz

$MHz = M\frac{c}{s} = \frac{c}{s} \cdot 10^6$

a) Timpii = ?

b) Mipsii = ?

a) $C_1 = 1 \cdot 5 + 2 \cdot 1 + 3 \cdot 1 = 10c$

$C_2 = 1 \cdot 10 + 2 \cdot 1 + 3 \cdot 1 = 15c$

(1) $\Rightarrow T_1 = \frac{C_1}{500 MHz} = \frac{10c}{500 MHz} = \frac{0.2\ c}{100\ MHz/s} = \frac{0.2\ s}{10^7} = 2 \cdot 10^{-8} s$

$T_2 = \frac{C_2}{500 MHz} = \frac{15c}{500 MHz} = \frac{3s}{100 \cdot 10^6} = 3 \cdot 10^{-8} s$

b) $MIPS_1 = \frac{\#i}{2 \cdot 10^{-8} s \cdot 10^6} = \frac{\#i}{2 \cdot 10^{-2} s} = \frac{\#i}{2s} \cdot 10^2 = 3.50 \frac{i}{s} = 350 mips$

$MIPS_2 = \frac{4 \cancel{00} i}{\cancel{5} \cdot 10^{-8} s \cdot 10^6} = 4 \cdot 10^2 \frac{i}{s} = 400 \frac{i}{s} = \cancel{400} mips$ - mai bun

Ex.

2 masini
2 prog

	M_1 (200 MHz)	M_2 (300 MHz)
Prog 1	10 s 200×10^6 i	5 s 160×10^6 i
Prog 2	3 s	4 s

a) Ce masina e mai rapida pt fiecare prog si cu cat?

b) Frecventa de executare a prog pe fiecare masina

$F_{i\ M_1}^{P_1}$, $F_{i\ M_2}^{P_2}$

c) CPI prog 1 pe masina 1 si 2

a) $P_1 \longrightarrow M_2$

$P_2 \longrightarrow M_1$

$\frac{P_{M_2}^{P_1}}{P_{M_1}^{P_1}} = \frac{T_{M_1}^{P_1}}{T_{M_2}^{P_1}} = \frac{10s}{5s} = 2$ perf.

$\frac{P_{M_1}^{P_2}}{P_{M_2}^{P_2}} = \frac{T_{M_2}^{P_2}}{T_{M_1}^{P_2}} = \frac{4s}{3s} = 1,(3)$

b) $F_{i\ M_1}^{P_1} = \frac{200 \times 10^6 i}{10 s} = 2 \cdot 10^7 \frac{i}{s} = 20 \cdot 10^6 \frac{i}{s}$ (mr instr)

$F_{i\ M_2}^{P_1} = \frac{160 \times 10^6 i}{5s} = 32 \cdot 10^6 \frac{i}{s}$ (timp)

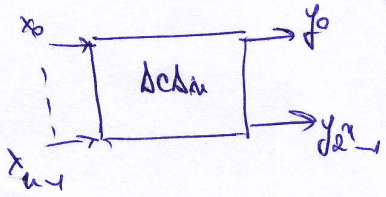
c) [illegible]

$$CPI^{P_1}_{M_1} = \frac{C^{P_1}_{M_1}}{I^{P_1}_{M_1}} = \frac{F_1 \cdot T^{P_1}_{M_1}}{I^{P_1}_{M_1}} = \frac{200MHz \cdot 10s}{200 \cdot 10^6 i} = \frac{10^9 c}{10^6 i} = 10 \text{ cpi}$$

$$CPI^{P_1}_{M_2} = \frac{F_2 \cdot T^{P_1}_{M_2}}{I^{P_1}_{M_2}} = \frac{300MHz \cdot 5s}{160 \cdot 10^6 i} = \frac{75 \cdot 10^8 c}{8 \cdot 10^6 i} = \frac{75}{8} \frac{c}{i} =$$

$= 9,375$ cpi

1) Decodificatorul



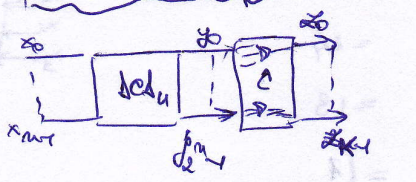
$(x_{n-1} \dots x_0) = k \in \overline{0, 2^n - 1}$

$\Downarrow$

$(0 \dots 1 \dots 0)$

y_k

2) Circuitul complet



$(x_{n-1}, \dots x_0) = k \in \overline{0, 2^n - 1}$

$\Downarrow$

$(0 \dots 1 \dots 0)$

y_k

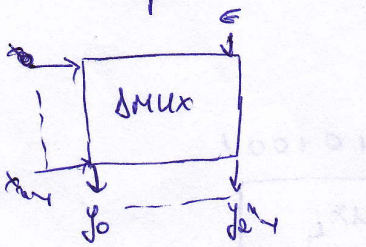
$\Downarrow$

$(z_{k-1} \dots z_0)$

$z_i (x_{n-1} \dots, x_0)$

$x_i (k)$

3) Demultiplexor



$(x_{n-1} \dots x_0) = k \in \overline{0, 2^n - 1}$

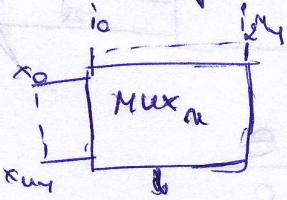
$e \in \{0, 1\}$

$\Downarrow$

$(0 \dots e \dots 0)$

y_k

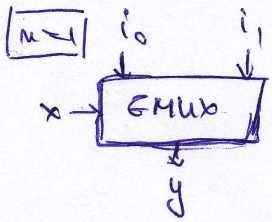
4) Multiplexor

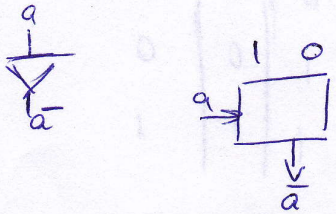


$(x_{n-1} \dots x_0) = k \in \overline{0, 2^n - 1}$

$(i_0 \dots i_{2^n-1}) \Rightarrow i_k$

4a) Multiplexoral elementar



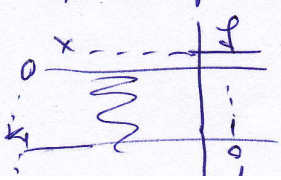


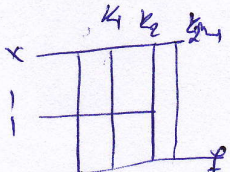
Ap:

1) a) un codificator care pt o secv de 4 biți scoate bitul majoritar

b) C care pt o secv. de 3 biți scoate $\Leftrightarrow$ nr. ~~biți~~ x este div cu 3

c) Mux pt implem. func. SAU de 3 b.





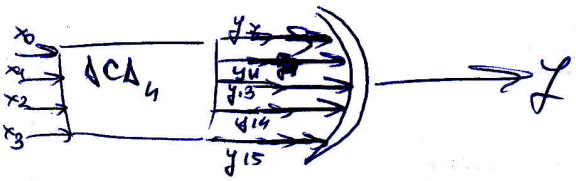
1 1 1 1 = 15

1 1 1 0 = 14

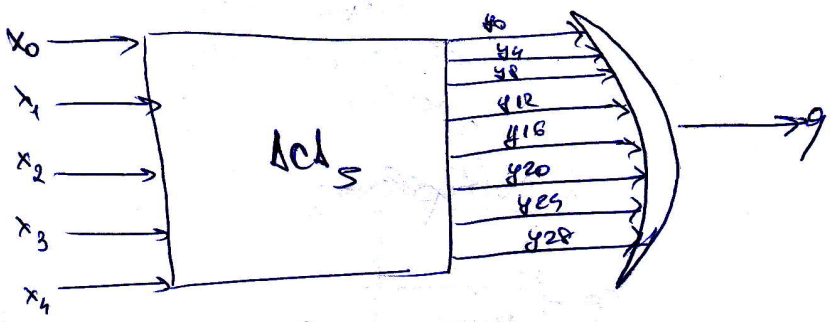
1 1 0 1 = 13

1 0 1 1 = 11

0 1 1 1 = 7



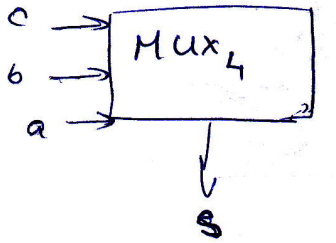
b) 0, 4, 8, 12, 16, 20, 24, 28

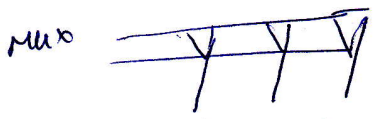


a)

a	b	c	S
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

0 1 1 0 1 0 0 1





2.) $f: B_2^5 \to B_2^3$, $f(x,y,z) = (\underbrace{x+\bar{y}z}_{f_1}, \underbrace{\bar{x}y+\bar{x}\bar{y}z}_{}, \underbrace{y+\bar{x}z}_{f_3})$

a) Implem f cu C

b) —▷— EMUX-uri

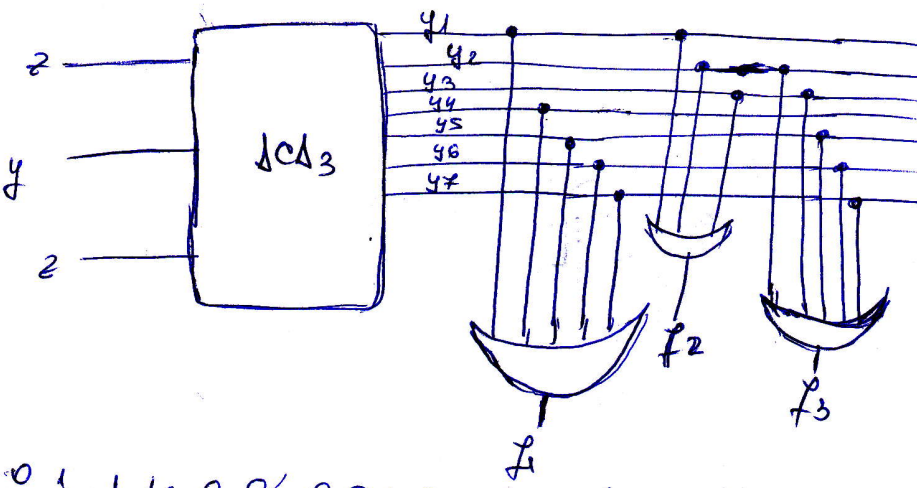
Red la max nr de EMUX

	x	y	z	$\bar{x}$	$\bar{y}$	$\bar{z}$	$\bar{y}z$	xz	$y+z$	f_1	f_2	f_3
0	0	0	0	1	1	1	0	0	0	0	0	0
1	0	0	1	1	1	0	1	0	1	1	1	0
2	0	1	0	1	0	1	0	0	1	0	1	1
3	0	1	1	1	0	0	0	0	0	0	1	1
4	1	0	0	0	1	1	0	0	0	1	0	0
5	1	0	1	0	1	0	1	1	1	1	0	1
6	1	1	0	0	0	1	0	0	1	1	0	1
7	1	1	1	0	0	0	0	1	0	0	0	1

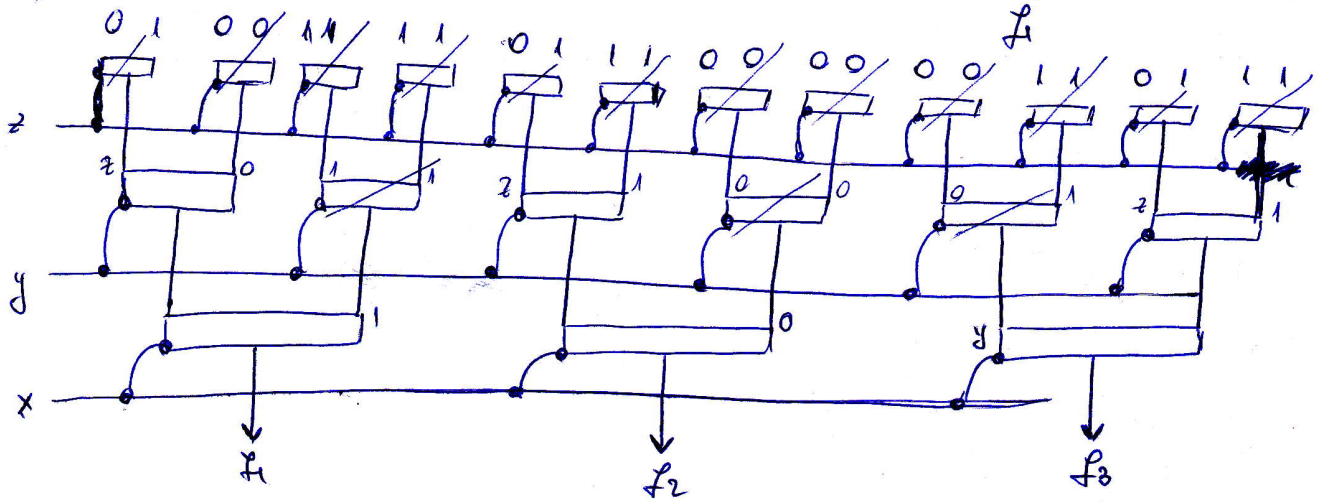
a) f_1: 1, 4, 5, 6, 7

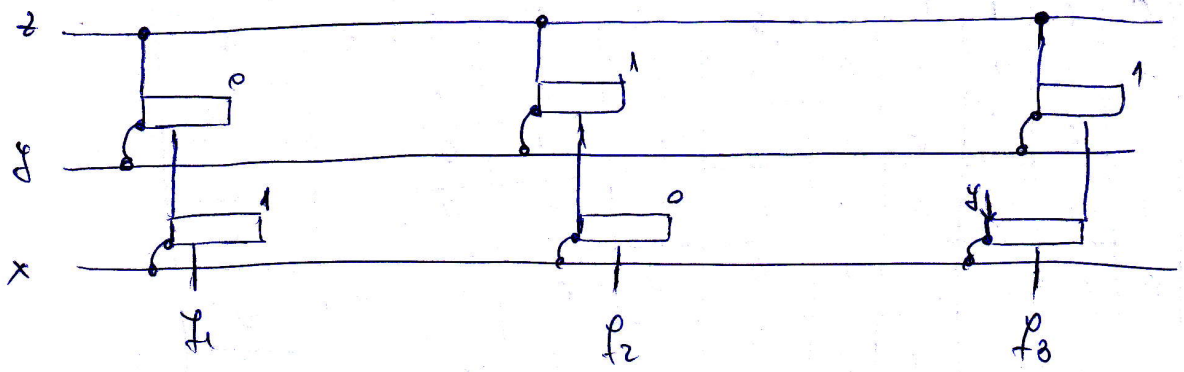
f_2: 1, 2, 3

f_3: 2, 3, 5, 6, 7

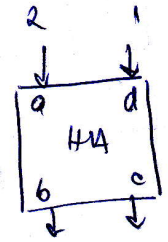


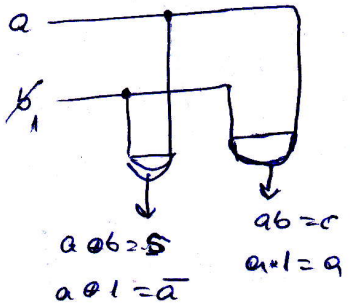
b)

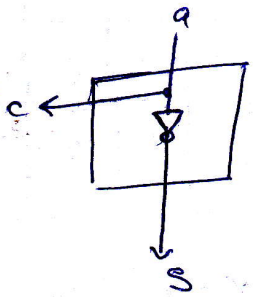


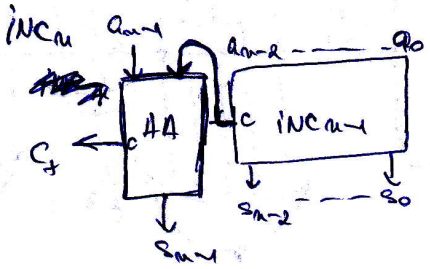


3) INC +1
INC$_0$

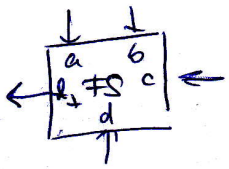








4) FS (Full subtract)



$a - b - t \Rightarrow d$
$\searrow t_+$

$a + \bar{b} = 1 = 2 - 1$

$a - b = a + \bar{b} + 1 - 2$

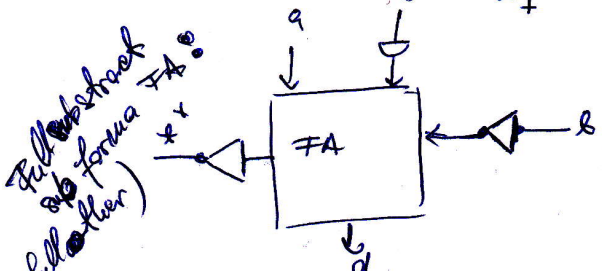
$a - b - t = a + \bar{b} + \bar{t} - 2$

$= FA_s(a, \bar{b}, \bar{t}) + FA_c(a, \bar{b}, \bar{t}) \cdot 2 - 2 =$

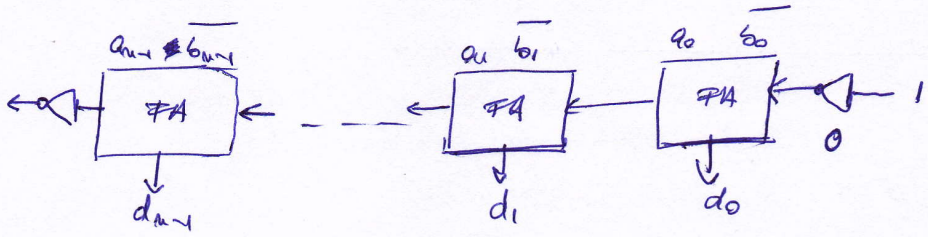
$= \underbrace{FA_s(a, \bar{b}, \bar{t})}_{d} - 2 \cdot \underbrace{\overline{FA_c(a, \bar{b}, \bar{t})}}_{t_+}$

$x + y = s(x,y) + c(x,y) \cdot 2$

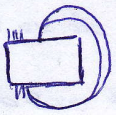
$1 - x = \bar{x}$



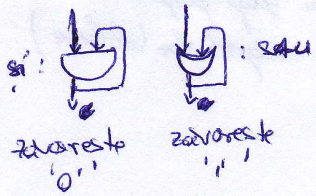
Construcția serială



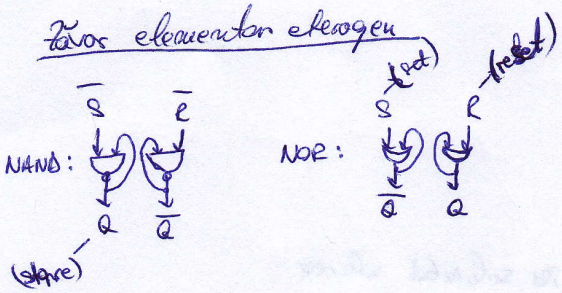
1-DS



Tăvoare elementare

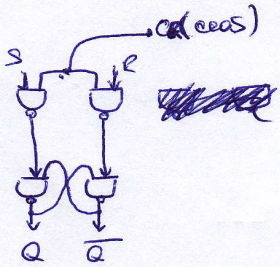


Zăvor elementar ebergen

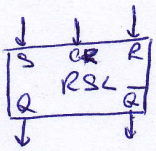


S	R	Q_{UM} (starea nouă)
0	0	Q_1
1	0	1 (set)
0	1	0 (reset)
1	1	nedefinit

Zăvor elementar cu ceas

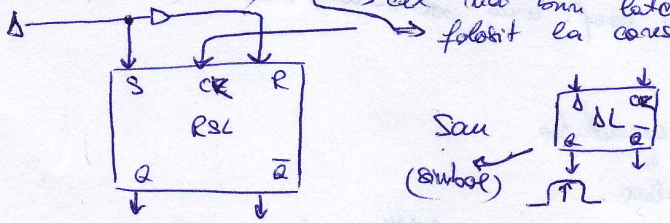


Reset set latch (RSL)

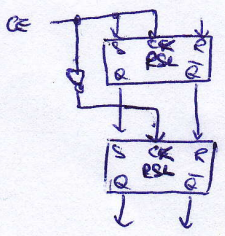


 - nivel de ceas (sensibil când e pe 1)

DL (Data Latch) → elimină hardware inutare (1,1)
→ cel mai bun latch
→ folosit la construcția RAM-ului

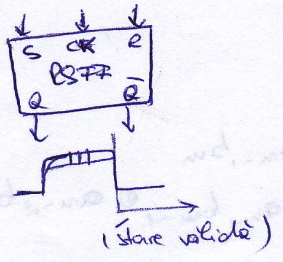


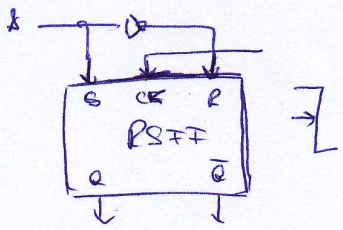
Struct. master slave



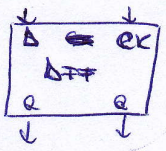
Când prea ceasul avem starea nouă

se transcrie → Reset-set flip-flop

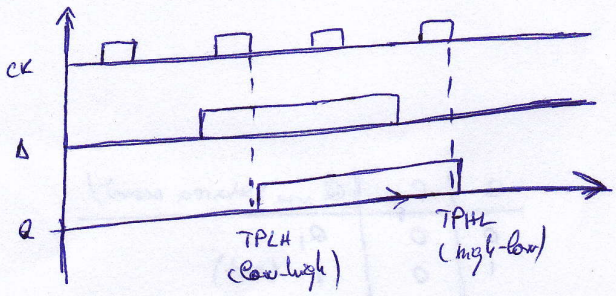




de Gay flip-flop

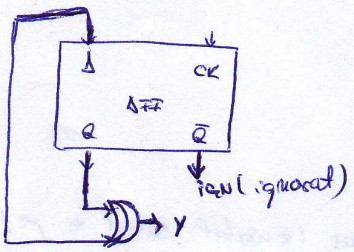


- iese D pe Q (cu întârziere de un front)
- la primul ceasul de ceas
- la primul ~~front~~ ceasul de ceas de după sch. lui D
- cel mai bun flip-flop
- construcția registrelor (folosit)



Exerc: 1) DFF. Circuit care schimbă 1 $\rightsquigarrow$ DFF își schimbă starea

0 0 1 1 0
0 0 1 0 1



2) Circuit de înmulțit polinoame

Fie $a(x) = \sum_{i=0}^{n} a_i x^i \in \mathbb{Z}_2[x]$ (evident $a_n = 1$)

Circuit care primește succesiv în coef. unui polinom $b(x) \in \mathbb{Z}_2[x]$ și scoate succesiv coef. prod. $a(x) \cdot b(x)$

Observăm că în $\mathbb{Z}_2$ $\begin{cases} \cdot = AND \\ + = XOR \end{cases}$ comutative și distr.

Fie $b(x) = \sum_{i=0}^{m} b_i x^i$ ($b_m = 1$) at $a(x) \cdot b(x) = \sum_{i=0}^{n+m} c_i x^i$, unde $\forall p$,

$$c_p = \sum_{\substack{i \in \overline{0,n} \\ j \in \overline{0,m} \\ i+j=p}} a_i b_j = \bigoplus_{\substack{i=\overline{0,n} \\ j \in \overline{0,m} \\ i+j=p}} a_i \cdot b_j$$

$c_{n+m} = a_n b_m = 1$

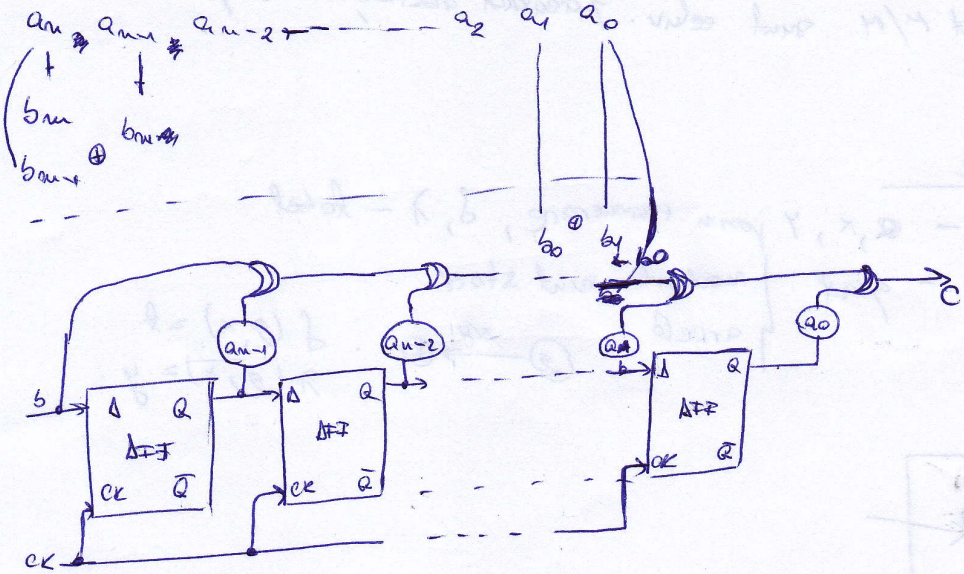
$c_{n+m-1} = a_n b_{m-1} \oplus a_{n-1} b_m$

$c_{n+m-2} = a_n b_{m-2} \oplus a_{n-1} b_{m-1} \oplus a_{n-2} b_m$

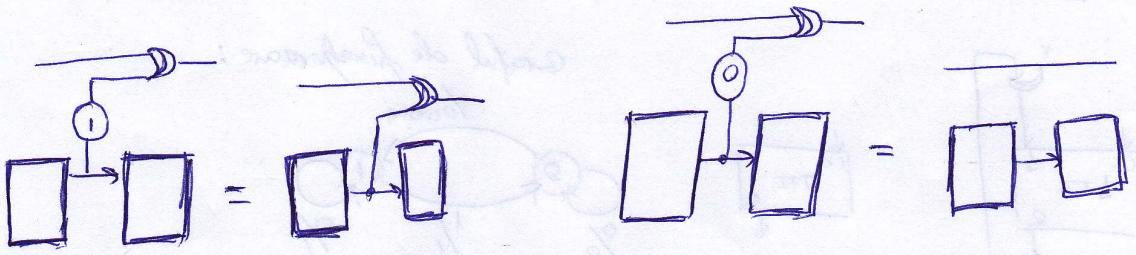
$\bar{c}_2 = \bar{a}_2 b_0 \oplus a_1 b_1 \oplus a_0 b_2$

$c_1 = a_1 b_0 \oplus a_0 b_1$

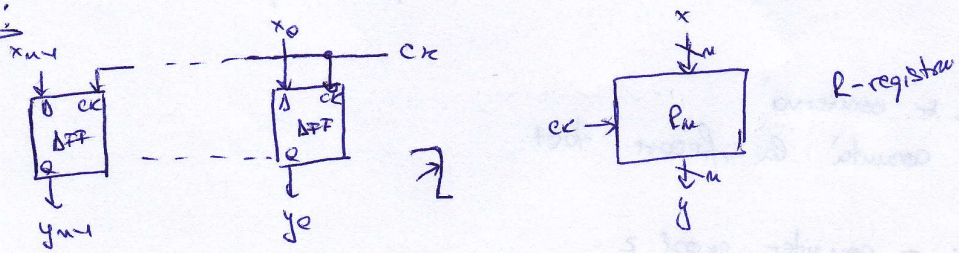
$c_0 = a_0 b_0$



$(q_i) = \begin{cases} 1, & a_i = 1 \\ \text{lipsă fir}, & a_i = 0 \end{cases}$



Curiozitate:



Automate și 2-AS

Def: Automat - șir de forma $A = (Q, X, Y, \delta, \lambda)$ unde

Q = mult. nevidă ale cărui elem. s.n stări

X = mult. finită $\neq \emptyset$ — val. de intrare

Y = — ieșire

δ = ~~funcție de tranziție~~ funcție de tranziție $\delta: Q \times X \to Q$

λ: $Q \times X \to Y$ (aut. Mealy) λ = funcție de ieșire

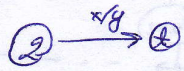
λ: $Q \to Y$ (aut. Moore)

Def. Automat finit { Q finit; δ acceptă o def. recursivă

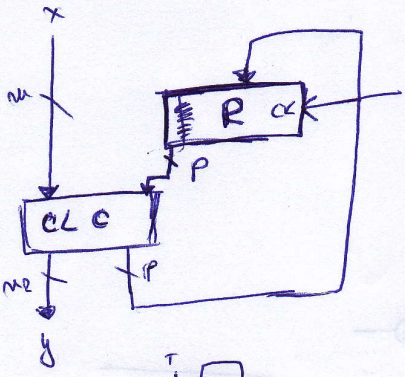
- Se dem. că aut M/M sunt echiv. față de abstracție de prima val de ieșire

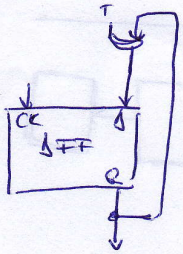
Aut Mealy finite

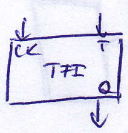
Reprezentare: - Q, X, Y prin enumerare, δ, λ - tabel
- graf { nodurile sunt stări; arcele



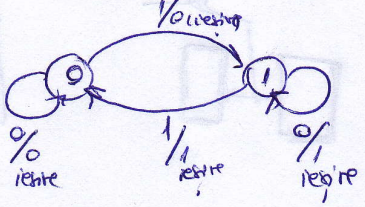
$\delta(q, x) = t$
$\lambda(q, x) = y$





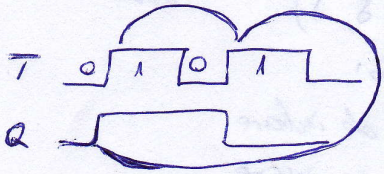


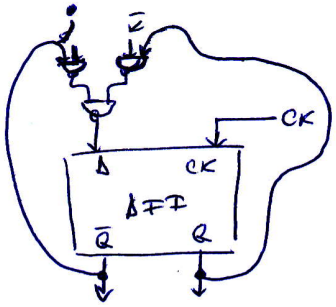
Graful de funcționare:



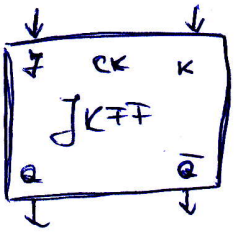
T=0 starea se conservă
T=1 starea comută la fiecare tact

TFF - utilizări: - numărător modul 2
- divizor de frecvență



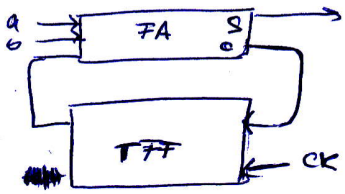


i	K	
0	0	→ ieșirea nemodificată (ca la ~~TFF~~ — T=0)
0	1	→ ieșirea 0 (reset)
1	0	→ ieșirea 1
1	1	→ ieșirea = starea veche



3. Să dă un automat. [illegible] Să se reprezinte matematic

~~Fie aut~~



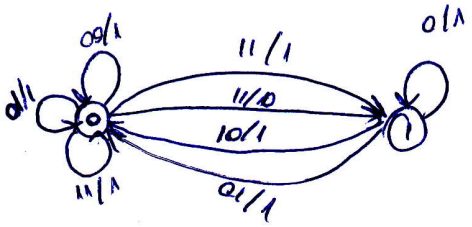
$Q = \{0,1\}$

$X = \{0,1\}^2$

$Y = \{0,1\}$

δ	00	01	10	11
0	0	0	0	1
1	1	0	0	0

λ	00	01	10	11
0	0	1	1	0
1	1	0	0	1

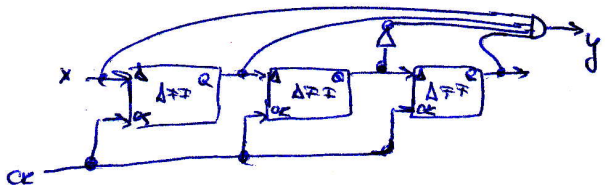


Automate cu stări finite

Enunț: Construiți un automat care recunoaște apariția subșirului 1011 în cadrul unui șir de biți oarecare

Ex: IN: 0 1 0 1 0 1 1 0 1 1 1 0 0 1 1

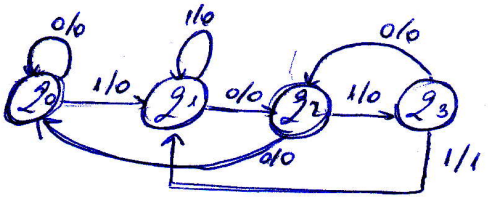
OUT: 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0



Reprezentare cu 1-bit

x - ce intră
y - ce iese

Graf:



$Q = \{0,1\}^2$, $q_0 = 00$, $q_1 = 01$, $q_2 = 10$, $q_3 = 11$, $X = \{0,1\}$, $Y = \{0,1\}$

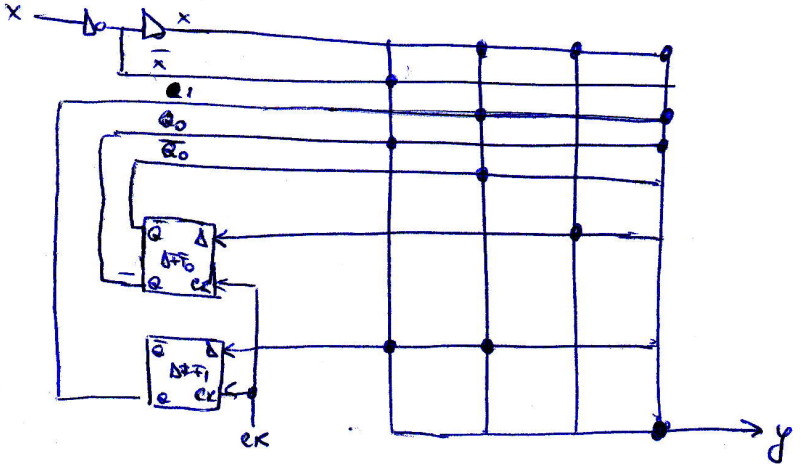
δ	x=0	x=1	Q_1^+	Q_0^+	T_1	T_0	J_1	K_1	J_0	K_0	J_1	K_1	J_0	K_0
00	00	01	0	x	0	x	0	–	x	–	0	x	x	x
01	10	01	x	x	x	x	x	–	–	x	x	x	x	x
10	00	11	x	x	x	x	–	x	x	–	x	x	x	x
11	10	01	x	x	x	x	–	x	–	x	x	x	x	x

$Q_1 = \bar{Q}_1 \bar{Q}_0$, $Q_0 = Q_1 \bar{Q}_0$

λ	x=0	x=1	y
00	0	0	0
01	0	0	0
10	0	0	0
11	0	1	x

$T = \{ x, 0 \to x ;\ \bar{x}, 1 \to x \}$

$JK = \{ x -, 0 \to x ;\ -\bar{x}, 1 \to x \}$



Q_1	Q_0	x	Q_1^+	Q_0^+
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	0
1	0	1	0	1
1	1	0	1	0
1	1	1	0	1

Q_1	Q_0	x	y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

FND: $Q_1^+ : \bar{Q}_1 Q_0 \bar{x} + Q_1 \bar{Q}_0 x + Q_1 Q_0 \bar{x} =$

$= Q_0 \bar{x} (\bar{Q}_1 + Q_1) + Q_1 \bar{Q}_0 x = Q_0 \bar{x} + Q_1 \bar{Q}_0 x$

$Q_0^+ = \bar{Q}_1 \bar{Q}_0 x + \bar{Q}_1 Q_0 x + Q_1 \bar{Q}_0 x + Q_1 Q_0 x =$

~~$= \bar{Q}_0 x (\bar{Q}_1 + Q_1) + Q_0 x$~~

$= x (\bar{Q}_1 \bar{Q}_0 + \bar{Q}_1 Q_0 + Q_1 \bar{Q}_0 + Q_1 Q_0) = x$

$Y : Q_1 Q_0 x$

FND alta forma tabel

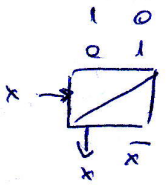
$Q_1^+ : \bar{Q}_1 Q_0 \bar{x} + Q_1 \bar{Q}_0 x + Q_1 Q_0 \bar{x}$

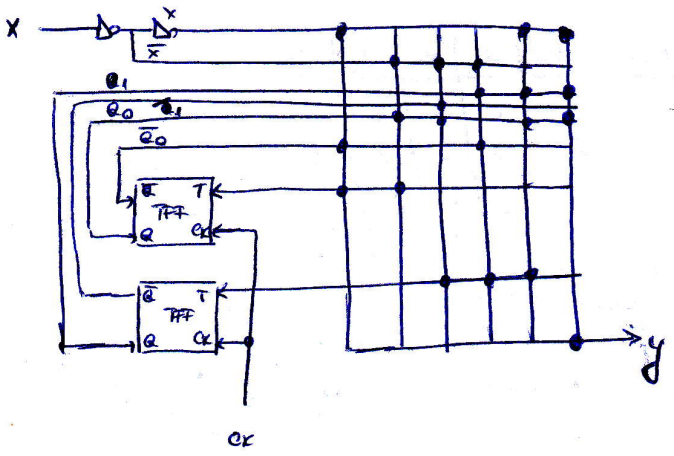
$Q_0^+ : - \quad - - - - \quad = x$

Forma simplificată:

$Q_1^+ = Q_0 \bar{x} + Q_1 \bar{Q}_0 x$

$Q_0^+ = x$





Q_1	Q_0	x	Q_1^+	Q_0^+	T_1	T_0
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	0	0
1	0	0	0	0	1	0
1	0	1	1	1	0	1
1	1	0	1	0	0	1
1	1	1	0	1	1	0

FND:

$T_1 : \bar{Q}_1 Q_0 \bar{x} + Q_1 \bar{Q}_0 \bar{x} + Q_1 Q_0 x$

$T_0 : \bar{Q}_0 x + Q_0 \bar{x}$