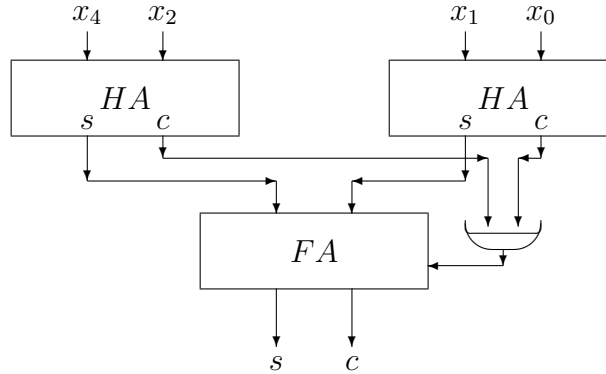


## Seminarul nr. 4

1. Se dă structura din figură (formată din două  $HA$ , un  $FA$  și o poartă  $OR$ ):



Construiți tabela de valori.

2. Construiți un circuit pentru scăderea a trei biți ( $x - y - z$ ), folosind următoarea tabelă:

$x$	0	0	0	0	1	1	1	1
$y$	0	0	1	1	0	0	1	1
$z$	0	1	0	1	0	1	0	1
$d$	0	1	1	0	1	0	0	1
$t$	0	1	1	1	0	0	0	1

Numiți acest circuit  $FS$  (full-subtract). Folosiți circuitul pentru a defini scăderea dintre două numere formate din câte 4 biți.

- Folosind acest  $FS$  construiți un circuit secvențial pentru scăderea a două numere cu un numă arbitrar de biți.
- Construiți un codicator cu prioritate pentru  $m = 7$  biți.
- Construiți un circuit de deplasare dreapta/stânga cu 0,1 poziții pentru un vector de 4 biți.
- Detaliați construirea  $CL$  (Carry lookahead) pentru un sumator pe 4 biți.
- Să se construiască un comparator pe pe 4 biți și apoi – pe baza lui – un comparator pe 16 biți.
- Construiți un comparator pe 4 biți folosind ca bază un singur 4 –  $FA$ .
- Se dă funcția booleană  $f : \{0, 1\}^3 \longrightarrow \{0, 1\}^3$  definită

$$f(x, y, z) = (\bar{x} + yz, x + \bar{y}z, x + y\bar{z})$$

- Construiți un  $PLA$  pentru implementarea ei.
- Construiți un  $PROM$  pentru implementarea ei.

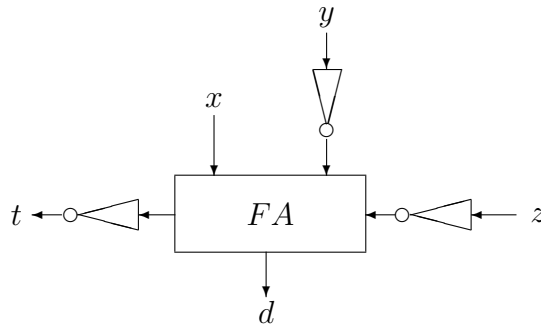
## Soluții:

1. Tabela de valori obținută este (am introdus și rezultatele parțiale):

$x_0$	$x_1$	$x_2$	$x_3$	$s_1$	$s_2$	$c_1$	$c_2$	$s$	$c$
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	1	0
0	0	1	0	0	1	0	0	1	0
0	0	1	1	0	0	0	1	1	0
0	1	0	0	1	0	0	0	1	0
0	1	0	1	1	1	0	0	0	1
0	1	1	0	1	1	0	0	0	1
0	1	1	1	1	0	0	1	0	1
1	0	0	0	1	0	0	0	1	0
1	0	0	1	1	1	0	0	0	1
1	0	1	0	1	1	0	0	0	1
1	0	1	1	1	0	0	1	0	1
1	1	0	0	0	0	1	0	0	1
1	1	0	1	0	1	1	0	0	1
1	1	1	0	0	1	1	0	0	1
1	1	1	1	0	0	1	1	1	0

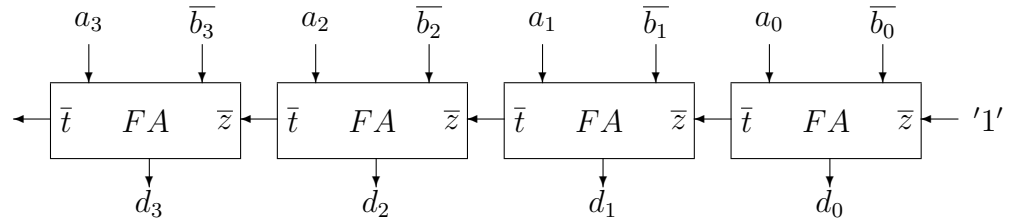
2. Deoarece  $x - y - z = x + (\bar{y} + 1 - 2^1) + (\bar{z} + 1 - 2^1) = x + \bar{y} + \bar{z} + 2^1 - 2^2$ , rezultă că diferența  $d(x, y, z)$  este rezultatul adunării  $x + \bar{y} + \bar{z}$ , iar  $t(x, y, z)$  este valoarea de transport complementată (datorită adunării cu  $2^1$ ) a acestei sume.

Deci un circuit digital – numit *FS* (full subtract) – pentru scădere este:

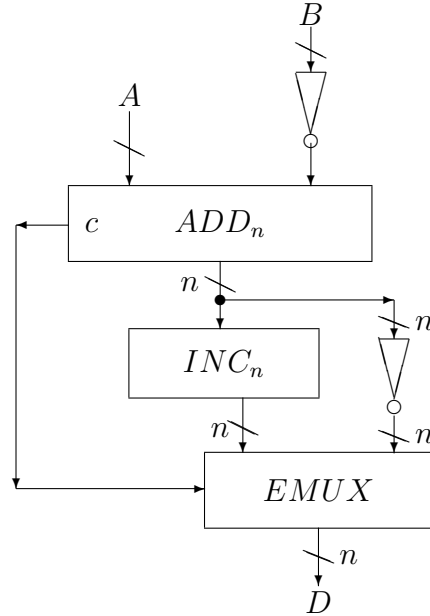


Pentru un circuit care să realizeze scăderea a două numere de câte  $n$  biți se va face o extensie serială formată din  $n$  astfel de circuite, similar construcției de la  $ADD_n$  sau  $INC_n$ . La identificarea transportului  $t$  de la un *FS* cu variabila  $z$  de la *FS*-ul anterior se pot elimina cele două complementări (care se anulează reciproc).

3. Să construim un circuit pentru scăderea a două numere de 4 biți ( $FS_4$ ). Schema sa este (pentru simplificare, porțile *NOT* au fost subînțelese):

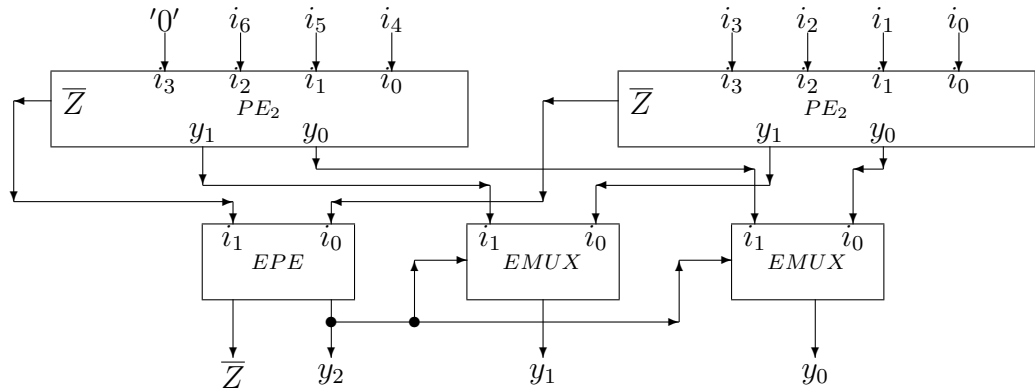


Un circuit mai compact pentru scădere este

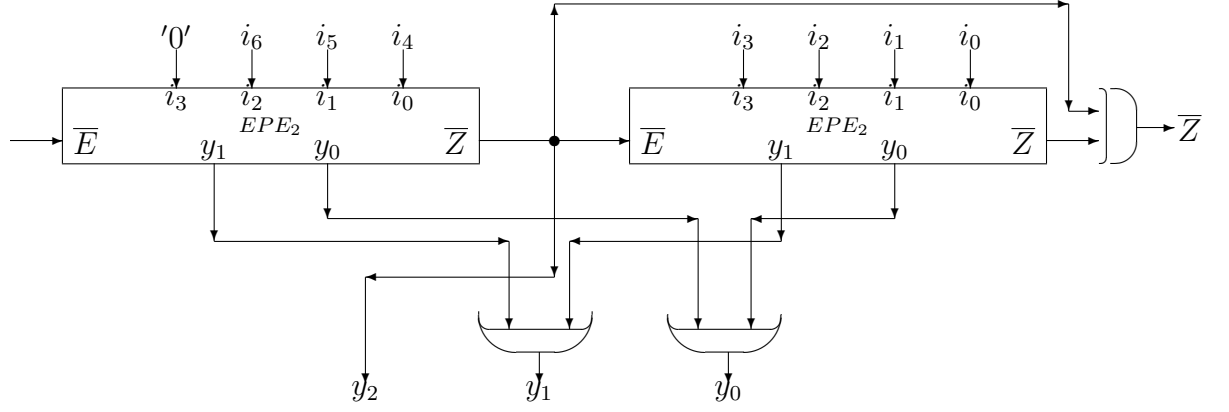


Avantajul acestui circuit este acela că se poate opera și cu rezultate negative. Semnul diferenței este dat de  $\bar{c}$  (deci  $c = 0$  semnifică un rezultat negativ, iar  $c = 1$  – unul pozitiv).

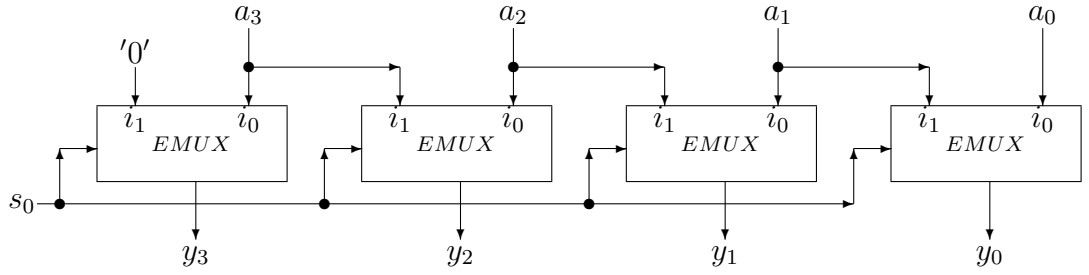
4. Un circuit bazat pe codificatori simpli (fără acces):



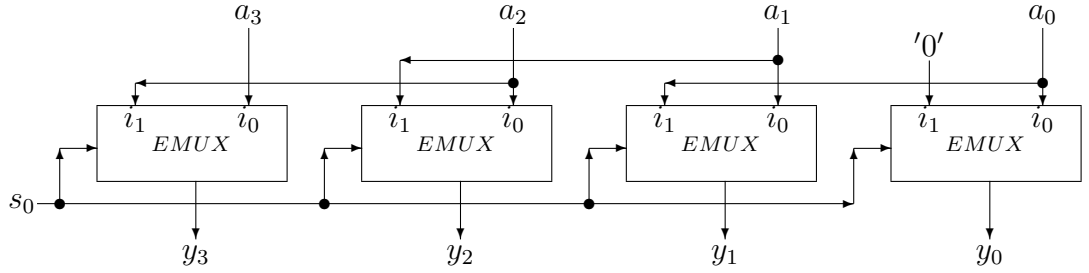
Dacă folosim construcția recursivă bazată pe codificatori cu acces, avem:



5. Circuitul de deplasare spre dreapta cu 0 sau 1 poziții este:



Similar se construiește și circuitul de deplasare spre stânga cu 0, 1 poziții:



6. Detaliem ecuația de transport  $c_{i+1} = G_i \oplus P_i c_i$  pentru  $i = 0, 1, 2, 3$ , eliminând și recursivitatea:

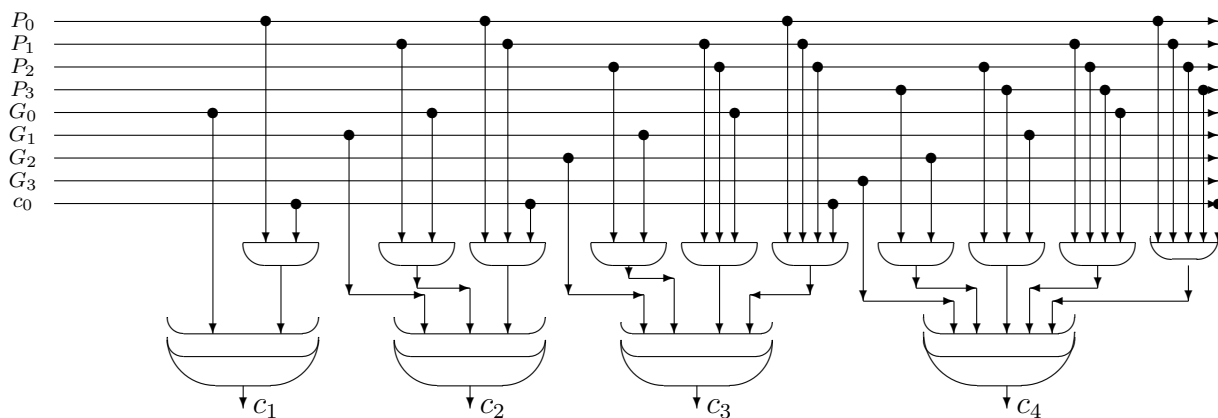
$$c_1 = G_0 \oplus P_0 c_0$$

$$c_2 = G_1 \oplus P_1 c_1 = G_1 \oplus P_1 G_0 \oplus P_1 P_0 c_0$$

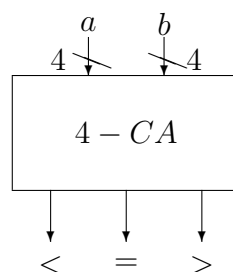
$$c_3 = G_2 \oplus P_2 c_0 = G_2 \oplus P_2 G_1 \oplus P_2 P_1 G_0 \oplus P_2 P_1 P_0 c_0$$

$$c_4 = G_3 \oplus P_3 c_3 = G_3 \oplus P_3 G_2 \oplus P_3 P_2 G_1 \oplus P_3 P_2 P_1 G_0 \oplus P_3 P_2 P_1 P_0 c_0$$

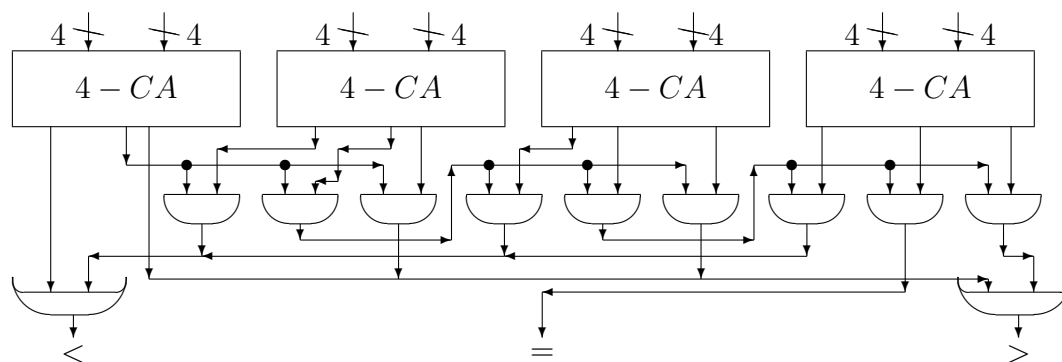
Un circuit posibil se construiește imediat (nu este singura variantă).



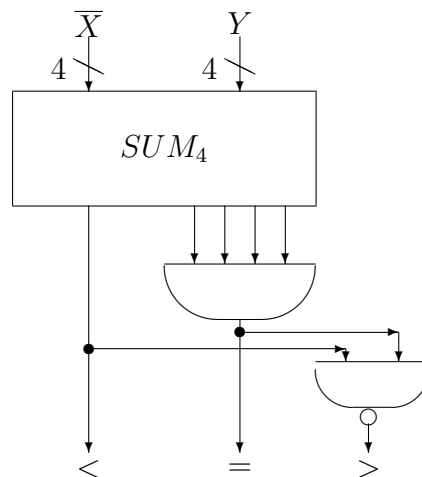
7. Construcția unui comparator pe 4 biți se realizează imediat particularizând structura de la curs. Dacă notăm



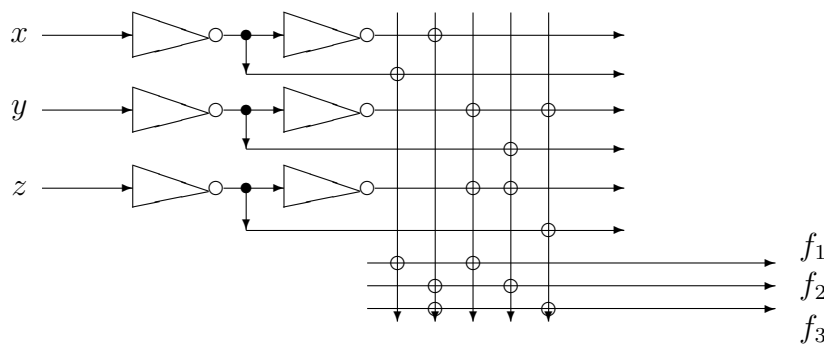
un circuit de comparare pe 4 biți, el poate fi folosit ca bază pentru construirea unui circuit pe 16 biți astfel:



8. Circuitul:



9. Un *PLA* pentru funcție:



Pentru circuitul *PROM* este nevoie de tabela de valori (sau – echivalent – de forma normal disjunctivă):

$x$	0	0	0	0	1	1	1	1
$y$	0	0	1	1	0	0	1	1
$z$	0	1	0	1	0	1	0	1
$f_1$	1	1	1	1	0	0	0	1
$f_2$	0	1	0	0	1	1	1	1
$f_3$	0	0	1	0	1	1	1	1

