

## Sumatoare binare

Subsistemul logic combinațional care asigură, direct sau indirect, efectuarea tuturor operațiilor aritmetice într-un sistem de calcul este sumatorul.

Operația de adunare a 2 numere binare  $A$  și  $B$  de  $n$  cifre (biți) constă în efectuarea la fiecare rang  $i$  ( $i = 0, 1, 2, \dots, n-1$ ) a unei sume  $S_i$ , prin adunarea cifrelor binare de același rang  $i$  ale numerelor  $A$  și  $B$  și a cifrei binare de transport  $C_{i-1}$  ce rezultă de la însumarea de pe rangul precedent.

De exemplu, la adunarea numerelor binare  $A=1011$  și  $B=1101$  vom avea:

$$\begin{array}{rcl} 1011 + & i=1 & - \text{ corespunde rangului 1 al biților numerelor binare} \\ \underline{1101} & C_{i-1}=C_0=1 & - \text{ este bitul de transport de la rangul 0 (imediat inferior)} \\ 11000 & S_i=S_1=0 & - \text{ este bitul sumă rezultat al însumării pe rangul 1} \end{array}$$
$$A_1 + B_1 + C_0 = 1 + 0 + 1 = 10$$
$$C_i=C_2=1 \quad - \text{ bitul de transport rezultat, trimis spre rangul 2 (imediat superior)}$$

După adunarea cifrelor binare de rang  $k$  cu bitul de transport  $C_{i-1}$ , rezultă bitul sumă  $S_i$  (pe rangul  $i$ ) și bitul de transport  $C_i$  (rezultat de la rangul  $i$ ). Adică se poate scrie relația:

$$A_i + B_i + C_{i-1} = C_i S_i \quad (1.1)$$

unde,  $C_i S_i$  reprezintă secvența celor doi biți rezultați din însumarea aritmetică.

În principiu, există două posibilități de realizare a unui sumator binar:

- sumatorul numit serie, la care biții de același rang, ai celor două numere care trebuiesc adunate, sunt aplicați la intrările sumatorului în mod succesiv/în serie, rang după rang; pentru realizarea unui astfel de sumator se folosește un singur circuit elementar (sumatorul complet pentru un rang – v. explicațiile de mai jos);
- sumatorul paralel/derivație, format, în cazul general, din  $n$  circuite elementare identice (sumatoare complete pentru un rang) la care cei  $n$  biți, ai celor două numere care trebuiesc adunate, se aplică simultan/în paralel la intrările celor  $n$  circuite elementare.

Prima soluție, deși este mai economică din punct de vedere al componentelor (este necesar un singur circuit elementar în loc de  $n$  circuite elementare pentru numere de  $n$  cifre) este mult prea lentă în raport cu cea de a doua, motiv pentru care se folosește foarte rar.

### 1. Sumatorul binar elementar

Sumatorul binar complet sau sumatorul elementar realizează însumarea a două cifre binare de același rang,  $A_i$  și  $B_i$  cu cifra transportului pentru rangul respectiv  $C_{i-1}$ , generând cifra sumei de rang  $i$ ,  $S_i$  și transportul pentru rangul superior,  $C_i$ . Funcțiile booleene  $S_i(A_i, B_i, C_{i-1})$  și  $C_i(A_i, B_i, C_{i-1})$  sunt definite în tabelul 1, întocmit pe baza regulii de adunare binară și au următoarele forme canonice disjunctive:

$$S_i = \bar{A}_i \bar{B}_i C_{i-1} + \bar{A}_i B_i \bar{C}_{i-1} + A_i B_i C_{i-1} + A_i \bar{B}_i \bar{C}_{i-1} \quad (1.2)$$

$$C_i = \bar{A}_i B_i C_{i-1} + A_i B_i \bar{C}_{i-1} + A_i B_i C_{i-1} + A_i \bar{B}_i C_{i-1} \quad (1.3)$$

Expresiile funcțiilor  $S_i$  și  $C_i$  se mai pot scrie:

$$\begin{aligned} S_i &= (\bar{A}_i B_i + A_i \bar{B}_i) \bar{C}_{i-1} + (A_i B_i + \bar{A}_i \bar{B}_i) C_{i-1} = (A_i \oplus B_i) \bar{C}_{i-1} + \overline{(A_i \oplus B_i)} C_{i-1} = \\ &= A_i \oplus B_i \oplus C_{i-1}, \end{aligned} \quad (1.4)$$

respectiv,

$$C_i = A_i B_i (C_{i-1} + \bar{C}_{i-1}) + (\bar{A}_i B_i + A_i \bar{B}_i) C_{i-1} = A_i B_i + (A_i \oplus B_i) C_{i-1}. \quad (1.5)$$

Tab. 1 Tabelul de adevăr al sumatorului complet.

| Intrări |       |           | Ieșiri |       |
|---------|-------|-----------|--------|-------|
| $A_i$   | $B_i$ | $C_{i-1}$ | $S_i$  | $C_i$ |
| 0       | 0     | 0         | 0      | 0     |
| 0       | 0     | 1         | 1      | 0     |
| 0       | 1     | 0         | 1      | 0     |
| 0       | 1     | 1         | 0      | 1     |
| 1       | 0     | 0         | 1      | 0     |
| 1       | 0     | 1         | 0      | 1     |
| 1       | 1     | 0         | 0      | 1     |
| 1       | 1     | 1         | 1      | 1     |

| $A_i B_i$ |    |    |    |    |
|-----------|----|----|----|----|
| $C_{i-1}$ | 00 | 01 | 11 | 10 |
| 0         | 0  | 1  | 0  | 1  |
| 1         | 1  | 0  | 1  | 0  |

(a)

| $A_i B_i$ |    |    |    |    |
|-----------|----|----|----|----|
| $C_{i-1}$ | 00 | 01 | 11 | 10 |
| 0         | 0  | 0  | 1  | 0  |
| 1         | 0  | 1  | 1  | 1  |

(b)

Fig. 1 Diagramele Karnaugh pentru funcțiile  $S_i$  și  $C_i$  ale sumatorului complet.

Având în vedere diagrama Karnaugh din figura 1-b se poate scrie, pentru funcția  $C_i$ , următoarea formă minimă:

$$C_i = A_i B_i + A_i C_{i-1} + B_i C_{i-1} = A_i B_i + (A_i + B_i) C_{i-1} \quad (1.6)$$

În figura 2 este reprezentată schema logică a unui sumator complet (s-au implementat relațiile (1,4) și (1,5)).

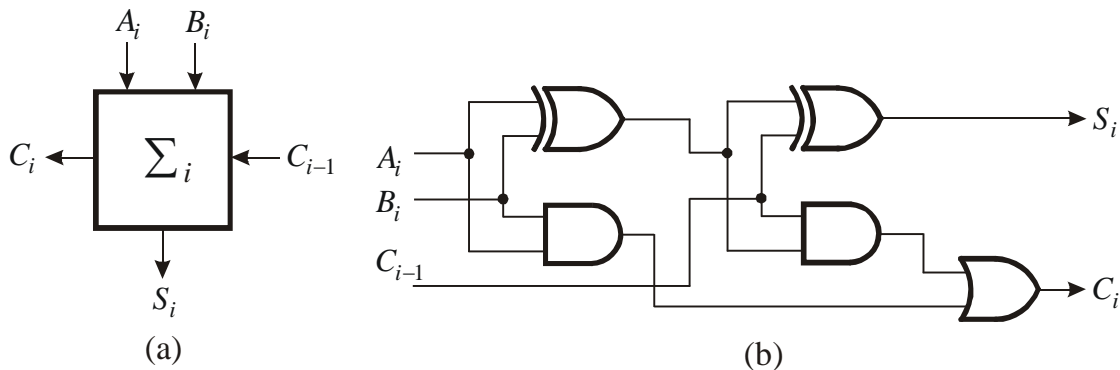


Fig.2 Sumatorul complet: (a) reprezentarea simbolică; (b) schema logică.

## 2. Sumatorul binar serie

Cu un singur sumator elementar se poate efectua adunarea a 2 numere binare  $A$  și  $B$  prin folosirea succesivă a sumatorului, rang după rang. Acest tip de sumator este denumit sumator serie. Cele două numere binare de adunat se găsesc, fiecare, depozitate într-un registru  $R_A$  și respectiv  $R_B$  iar bitul sumă rezultat la fiecare rang este înregistrat imediat într-un registru  $R_S$  (v. fig. 3).

La fiecare un ciclu, marcat de un impuls de clock (de tact), se efectuează o adunare pe un rang. Când se adună cifrele, de exemplu pe rangul  $k$ , la intrările sumatorului elementar se găsesc biții  $A_i$ ,  $B_i$  și  $C_{i-1}$ . Transportul de rang  $C_i$ , rezultat din însumare, este aplicat intrării unui bistabil de tip D. La bascularea bistabilului, ca urmare a aplicării aceluiași impuls de clock, bitul  $C_i$  este transferat de la intrarea D a bistabilului la ieșirea sa Q, de unde este transmis ca bit  $C_{i-1}$  intrării cu același nume sumatorului.

Deplasarea bit cu bit a informației din regiștrii  $R_A$  și  $R_B$  la intrările sumatorului precum și bascularea bistabilului se produc simultan, la comanda aceluiași semnal de clock.

Acest sumator serie se utilizează rar, datorită duratei mari a operației de adunare a celor  $n$  biți ai numerelor. Durata operației de adunare este dată de produsul  $n \cdot \pi$ , unde  $\pi$  reprezintă timpul de rezoluție (întârzierea) introdusă de bistabilul tip D. Întârzierea totală, în cazul unui sumator de 4 biți, poate ajunge  $4 \cdot 22 = 88$  nsec. În cazul adunării unor numere cu multe cifre binare (adică cu un  $n$  de valoare mare) întârzierea devine stânjenitoare.

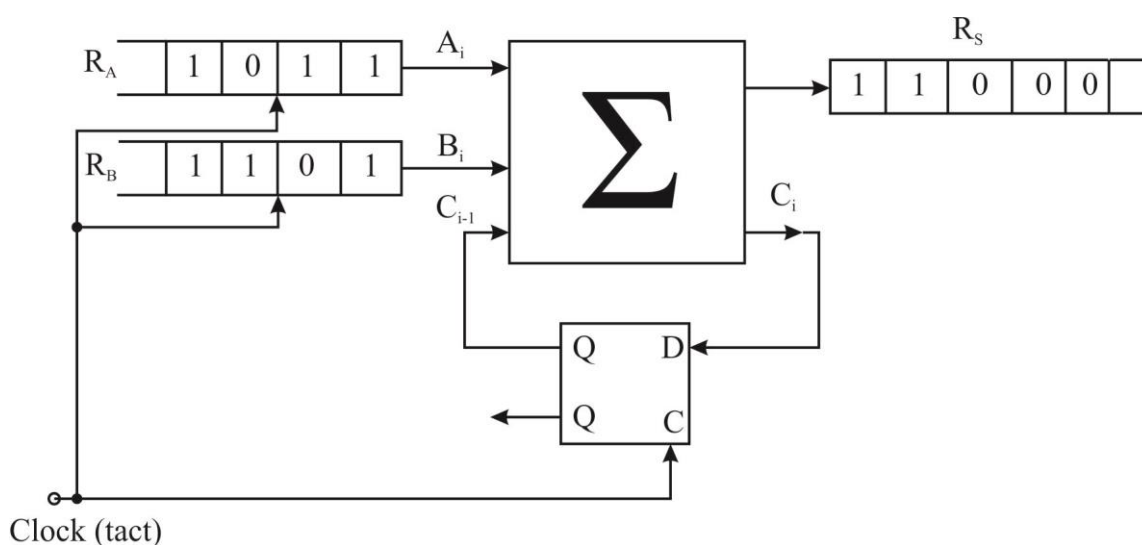


Fig. 3 Schema bloc a sumatorului binar serie

### 3. Sumatorul binar paralel (derivație)

Mai rapid este sumatorul paralel alcătuit din mai multe sumatoare elementare, destinate fiecare adunării pe un rang. Biții celor doi operanzi sunt aplicați simultan, pe toate rangurile, sumatoarelor de rang. Însumarea biților pe un rang trebuie însă să aștepte sosirea bitului de transport de la ieșirea sumatorului precedent.

Schema bloc a unui astfel de sumator paralel (derivație) pentru 4 biți este prezentată în figura 4. Un exemplu de sumator paralel pentru 4 biți, cu transport succesiv, se poate realiza într-o schemă cu c.i. din clasa SSI (cu inversoare 74LS04, porți AND 74LS08 și porți OR 74LS32) cu ajutorul cărora se alcătuiesc două scheme identice de semi-sumator ce se leagă între ele.

Un astfel de bloc de două semisumatoare constituie un sumator de rang. Din 4 astfel de blocuri se alcătuiește sumatorul derivație de 4 biți.

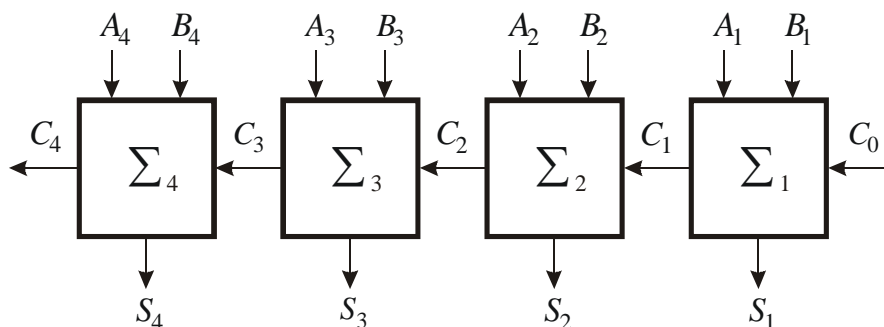


Fig. 4 Schema bloc a sumatorului derivație de 4 biți.

La acest tip de sumator, adunarea pe fiecare rang se face numai după sosirea bitului de transport de la rangul imediat inferior.

Din acest motiv, transportul este denumit transport succesiv (ripple carry – transport în cascadă). După cum se vede, el introduce o întârziere (un timp de delay/rezoluție) ce poate avea ca valoare maximă valoarea dată de relația  $t_r = n \cdot m \cdot t_p$  [nsec], unde:

- $n$  este numărul sumatoarelor de rang;
- $m$  este numărul nivelelor logice parcurse de bitul de transport într-un sumator de rang;
- $t_p$  este timpul de propagare al bitului de transport printr-un nivel logic.

#### 4. Sumatorul cu transport anticipat

Pentru reducerea întârzierii datorate propagării transportului în serie se folosește sumatorul cu transport anticipat.

Analizând relația bitului de transport (v. ec. 1.5) se observă că transportul  $C_i$  devine:

$$C_i = \begin{cases} 1 & \text{și este independent de } C_{i-1}, \text{ când } A_i = B_i = 1, \\ 0 & \text{și este independent de } C_{i-1}, \text{ când } A_i = B_i = 0, \\ C_{i-1} & \text{când } A_i \neq B_i. \end{cases} \quad (1.7)$$

Această constatare a condus la ideea calculării anticipate a cifrei binare de transport în funcție de datele de intrare, simultan pentru toate rangurile. În felul acesta, nu se mai așteaptă efectuarea calculelor pe rangul mai puțin semnificativ în vederea generării cifrei de transport ce se adună la cifrele rangului următor. Aceasta înseamnă, în cazul sumatorului pentru patru ranguri, că trebuie realizate funcțiile:

$$C_1 = f_1(A_1, B_1, C_0) \quad (1.8)$$

$$C_2 = f_2(A_2, A_1, B_2, B_1, C_0) \quad (1.9)$$

$$C_3 = f_3(A_3, A_2, A_1, B_3, B_2, B_1, C_0) \quad (1.10)$$

$$C_4 = f_4(A_4, A_3, A_2, A_1, B_4, B_3, B_2, B_1, C_0) \quad (1.11)$$

Pentru obținerea unei forme cât mai simple a funcțiilor (1.8-1.11) se pleacă de la expresia transportului pentru un rang (v. ec. 1.6), dată mai jos:

$$C_i = A_i B_i + A_i C_{i-1} + B_i C_{i-1} = A_i B_i + (A_i + B_i) C_{i-1} \quad (1.12)$$

în care s-au introdus funcțiile auxiliare  $G_i$ , transportul generat în rangul  $i$  (dacă  $A_i B_i = 1$ , rezultă  $C_i = 1$  indiferent de valoarea lui  $C_{i-1}$ ) și  $P_i$ , condiția de propagare a transportului generat în rangurile inferioare lui  $i$ , peste rangul  $i$  ( $P_i = 1$  va determina „traversarea“ celulei  $\Sigma_i$  de către valoarea bitului de transport  $C_{i-1}$ ).

Se obțin expresiile funcțiilor transporturilor  $C_1, C_2, C_3$  și  $C_4$ , pentru sumatorul de patru ranguri:

$$C_1 = G_1 + P_1 C_0 \quad (1.13)$$

$$C_2 = G_2 + P_2 G_1 + P_2 P_1 C_0 \quad (1.14)$$

$$C_3 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 C_0 \quad (1.15)$$

$$C_4 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1 + P_4 P_3 P_2 P_1 C_0 \quad (1.16)$$

Pe baza relațiilor (1.13) ÷ (1.16) și (1.4) se obține schema logică a sumatorului de 4 biți cu transport anticipat din figura 5.

Deoarece se urmărește realizarea în timp cât mai scurt a funcțiilor (1.13) ÷ (1.16), cele mai indicate circuite sunt porțile logice ȘI-SAU-NU. Pentru sinteza cu acest tip de porți sunt necesare formele normale disjunctive ale negatelor funcțiilor. De aceea, în practică, se realizează funcțiile auxiliare  $\overline{G}_i$  și  $\overline{P}_i$ , respectiv funcțiile:

$$\begin{aligned} \overline{C}_i &= \overline{A_i B_i + (A_i + B_i) C_{i-1}} = \overline{A_i B_i} [(\overline{A_i + B_i}) + \overline{C_{i-1}}] = \\ &= \overline{P}_i + \overline{C_{i-1}} \overline{G}_i. \end{aligned} \quad (1.17)$$

Pe baza relației (1.17) se obțin, pentru patru ranguri:

$$C_1 = \overline{\overline{P}_1 + \overline{G}_1 \overline{C}_0} \quad (1.18)$$

$$C_2 = \overline{\overline{P}_2 + \overline{G}_2 \overline{P}_1 + \overline{G}_2 \overline{G}_1 \overline{C}_0} \quad (1.19)$$

$$C_3 = \overline{\overline{P}_3 + \overline{G}_3 \overline{P}_2 + \overline{G}_3 \overline{G}_2 \overline{P}_1 + \overline{G}_3 \overline{G}_2 \overline{G}_1 \overline{C}_0} \quad (1.20)$$

$$C_4 = \overline{\overline{P}_4 + \overline{G}_4 \overline{P}_3 + \overline{G}_4 \overline{G}_3 \overline{P}_2 + \overline{G}_4 \overline{G}_3 \overline{G}_2 \overline{P}_1 + \overline{G}_4 \overline{G}_3 \overline{G}_2 \overline{G}_1 \overline{C}_0} \quad (1.21)$$

Dezavantajul sumatoarelor cu transport anticipat constă în faptul că odată cu creșterea numărului de ranguri crește considerabil complexitatea circuitelor pentru obținerea transportului. Din acest motiv, pentru adunarea unor operanzi cu număr mare de cifre, se folosesc mai multe sumatoare cu propagarea transportului în paralel pentru un număr limitat de ranguri, de obicei patru. Transportul între aceste sumatoare se propagă fie în serie, fie în paralel.

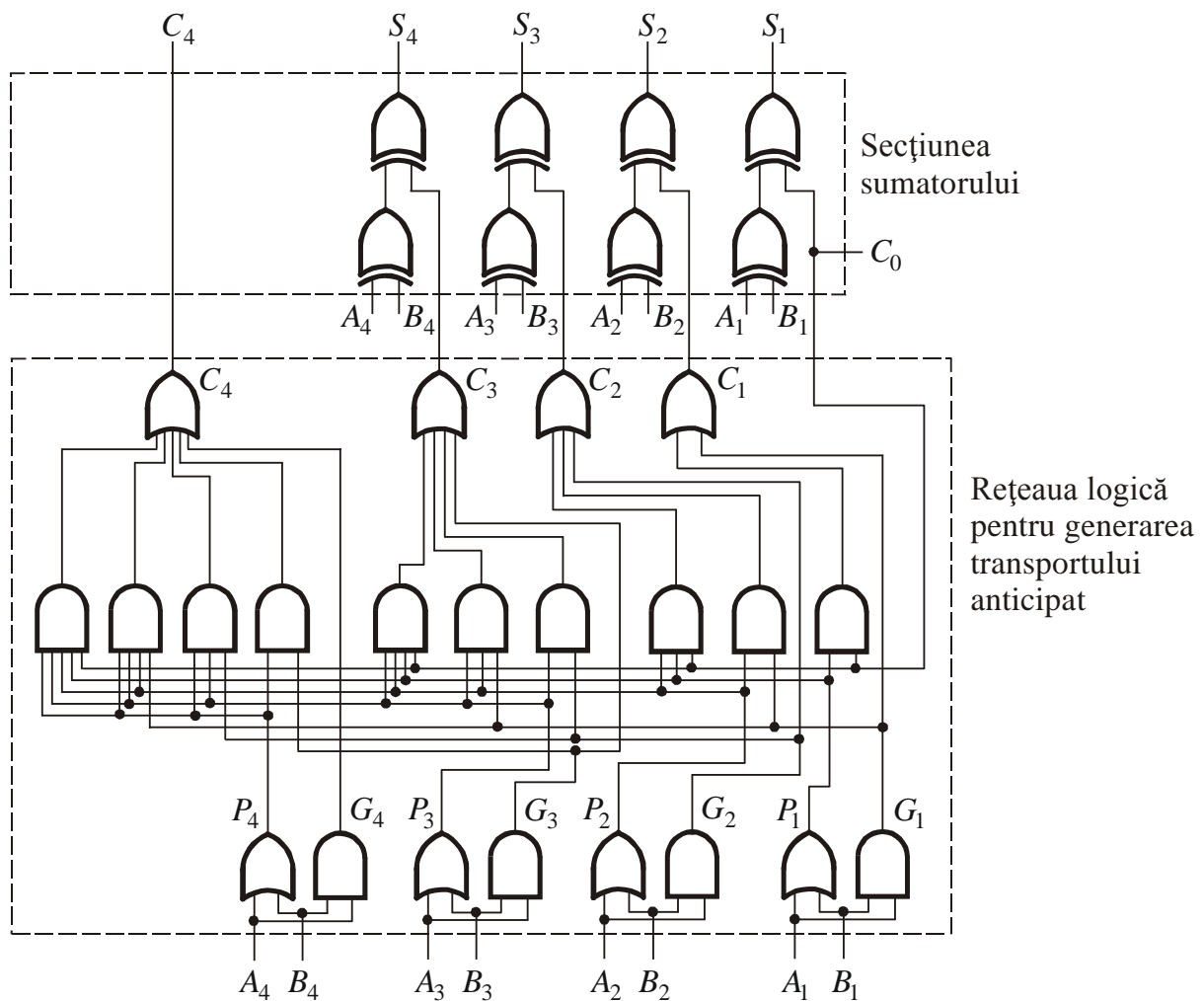


Fig. 4 Schema logică a unui sumator binar de 4 biți cu transport anticipat.

Unele circuite sumatoare MSI, pentru patru biți, de exemplu circuitul 7483, a cărui schemă logică este prezentată în figura 5, realizează anticipat numai transportul final, notat cu  $C_4$ , transporturile intermediare  $\bar{C}_1, \bar{C}_2$ , și  $\bar{C}_3$ , fiind realizate succesiv, pe baza relației (1,17). Anticiparea ultimului transport permite circuitului sumator pentru rangurile superioare să înceapă lucrul mai repede, reducându-se astfel substanțial timpul de adunare al numerelor cu multe cifre binare.

Așa cum s-a mai precizat, sumatoarele pentru mai multe ranguri se pot realiza prin legarea în cascadă a mai multor sumatoare complete pentru 4 biți. Astfel, cu circuitul integrat 7483 se poate realiza un sumator de numere întregi de  $n$  biți, conectând în serie un număr  $k = n/4$  de circuite integrate, unde  $n$  este întotdeauna o putere a lui 2. În figura 6 este dată schema logică a unui sumator pentru 16 ranguri binare.

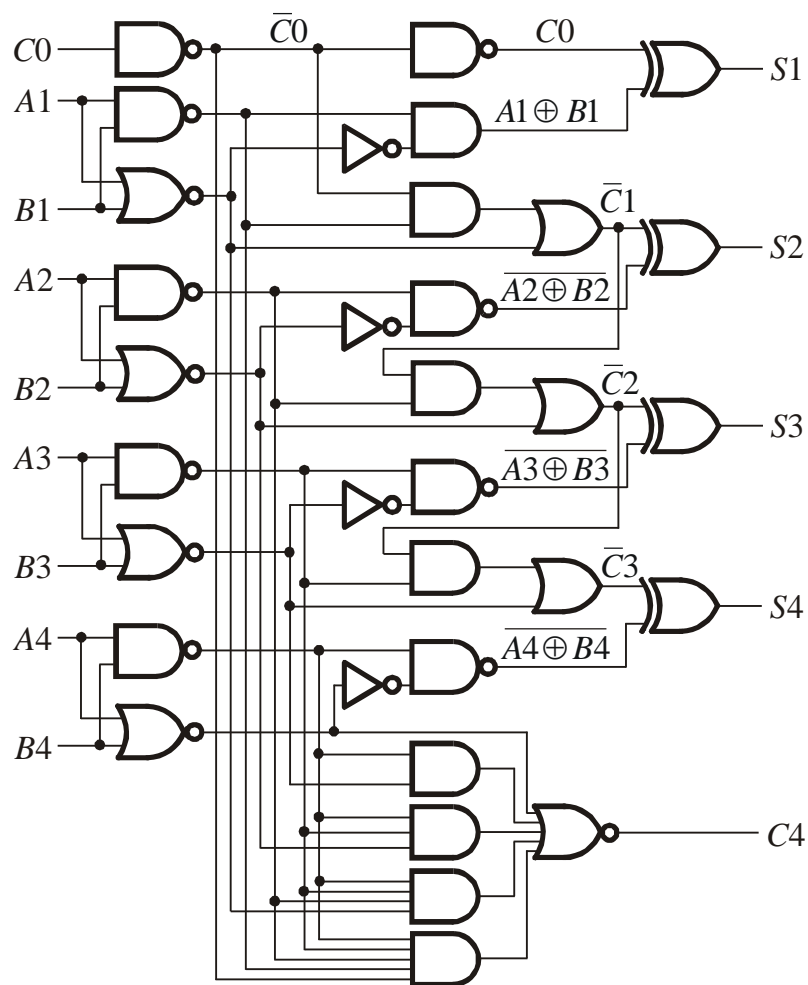


Fig. 5 Schema logică a circuitului sumator 7483.

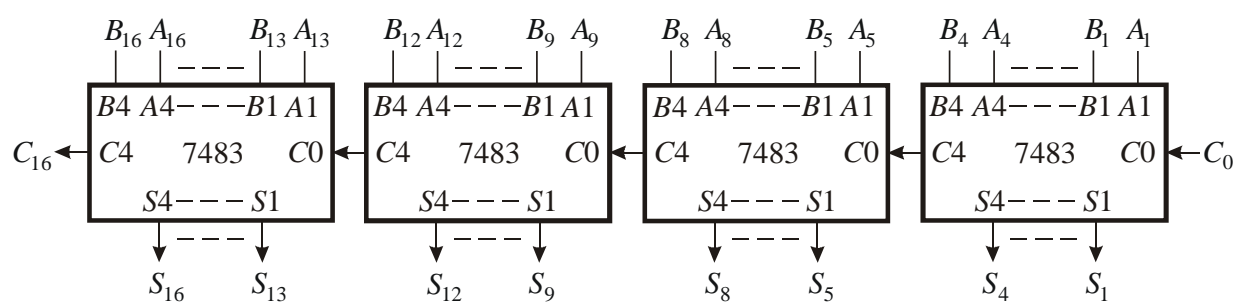


Fig. 6 Schema logică a unui sumator pentru 16 ranguri binare, realizat cu circuite 7483.