

Cursul nr. 6

C6.1 Multiplexorul / Selectorul de date

Multiplexorul (MUX) este un circuit logic combinațional care selectează una din intrările sale pentru a o transmite la ieșirea unică. Schema de principiu este reprezentată în fig. C6.1a.

Selecția uneia dintre intrări se face cu ajutorul comutatorului K. În fig. C6.1b este reprezentată schema bloc a unui multiplexor cu 4 căi (D_0, D_1, D_2, D_3) și 2 intrări de selecție A și B care îndeplinesc rolul comutatorului K din figura C6.1a. Din tabelul de adevăr din fig. C6.1c rezultă expresia care descrie funcționarea multiplexorului cu 4 căi:

$$Y = (\overline{B}\overline{A})D_0 + (\overline{B}A)D_1 + (B\overline{A})D_2 + (BA)D_3$$

Funcția poate fi implementată sub forma circuitului din fig. C6.2a. MUX-ul cu 4 căi poate fi utilizat în scheme arborescente ca cea din fig. C6.2b, pentru extinderea numărului de căi, în cazul prezentat la 16. I_0, \dots, I_7 și I_8, \dots, I_{15} nefigurate, sunt intrările de date, S_0, S_1, S_2 și S_3 intrările de selecție.

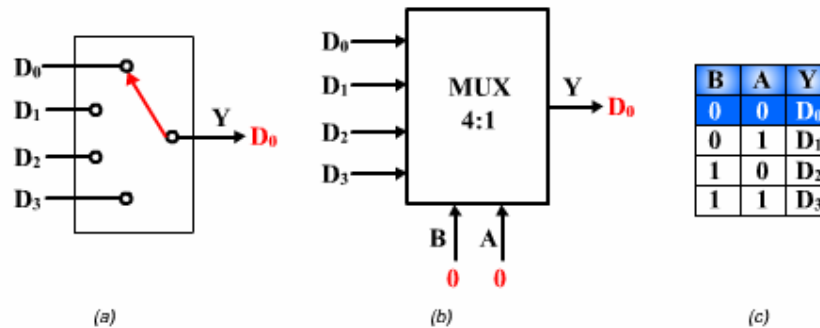


Fig. C6.1 Multiplexorul: (a) Schema de principiu; (b) Schema bloc pentru MUX cu 4 căi; (c) Tabelul de adevăr pentru MUX cu 4 căi

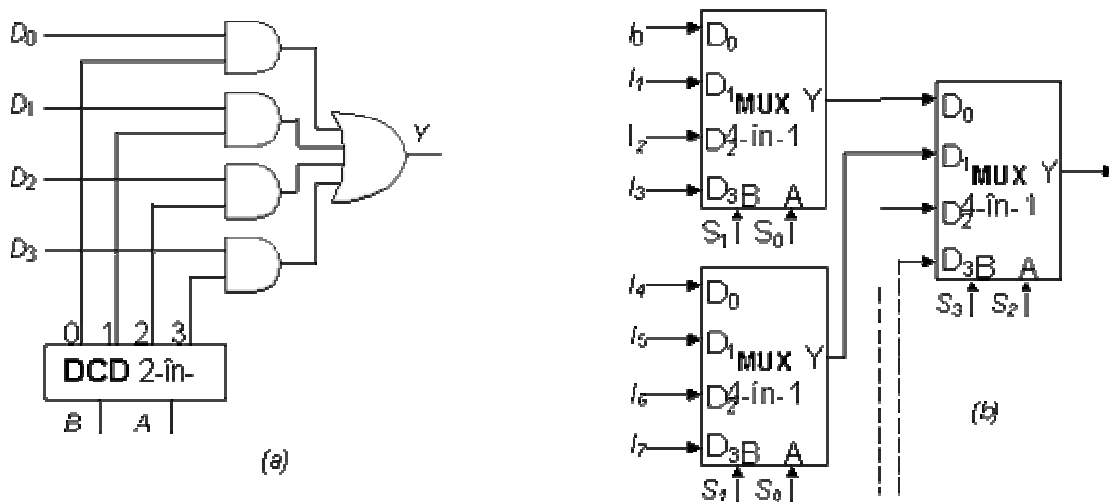


Fig. C6.2 Multiplexorul 4-în-1: (a) Schema circuitului; (b) Extinderea numărului de căi de intrare

Circuite integrate multiplexoare standard:

- **74151A**, este un multiplexor 8-în-1, a cărei configurație este reprezentată în fig. C6.3a. De altfel, funcționarea sa este sintetizată de expresia:

$$Y = \left(\sum_{i=0}^7 m_i D_i \right) (\overline{0})$$

în care \overline{G} este *strob*-ul (semnal de validare activ pe nivel scăzut), expresie ce poate fi transpusă în tabelul de adevăr din fig. C6.3b.

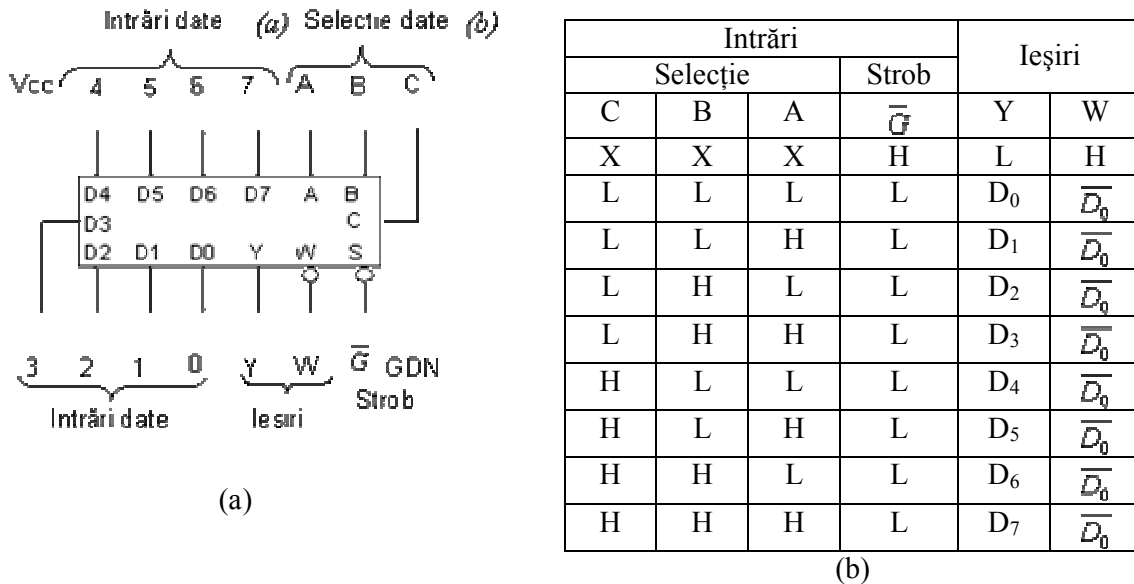


Fig. C6.3 Multiplexorul 8-în-1 tip 74151A: (a) Configurație terminale; (b) Tabel de adevăr

- **74150** este un multiplexor 8-în-1 într-o capsulă cu 24 pini, având 16 intrări de date (E_i), un strob (\overline{G}), 4 intrări de selecție și o ieșire (W). Funcționarea este dată de relația

$$W = \left(\sum_{i=0}^{15} m_i E_i \right) \overline{G}$$

- **74153** este un circuit ce conține 2 multiplexoare 4-în-1 cu intrări de selecție comune, permițând astfel multiplexarea simultană a 2 cuvinte de 4 biți.
- **74157** este un circuit ce conține 4 multiplexoare 2-în-1 cu intrări de selecție comune, permițând astfel multiplexarea simultană a 4 cuvinte de 2 biți.

Multiplexorul poate fi utilizat și la implemetarea funcțiilor logice.

EXEMPLUL C6.1

Să se implementeze funcția $f(a, b, c) = ab + \overline{bc}$ cu ajutorul multiplexorului 4-în-1.

Mai întâi se aduce expresia funcției la forma canonică disjunctivă:

$$f(a, b, c) = abc + abc + \overline{abc} + a\overline{bc}$$

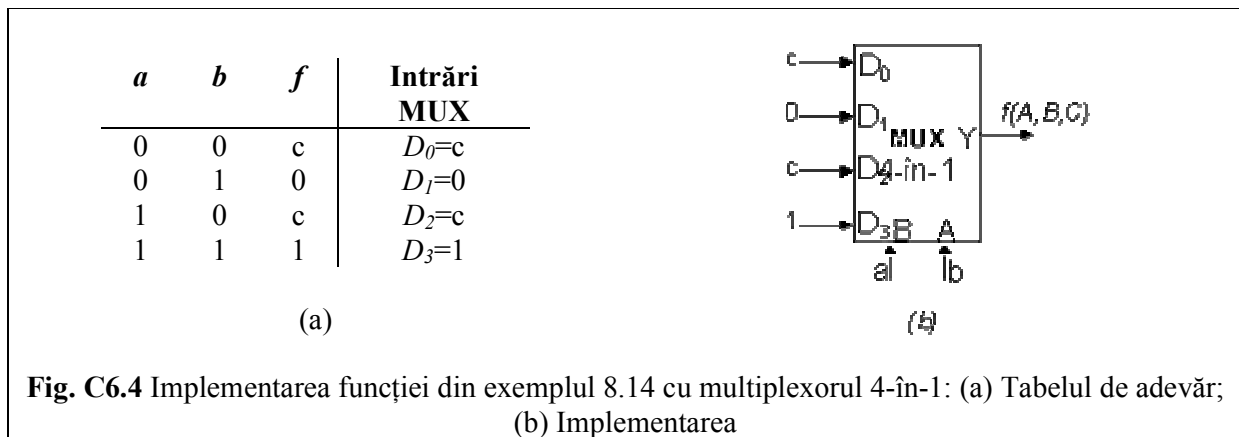
și apoi se alege 2 variabile pentru a le lega la intrările de selecție. Alegem de exemplu variabilele **a** și **b** prin care și factorizăm expresia

$$f(a, b, c) = \overline{a}\overline{b}(c) + a\overline{b}(c) + ab(\overline{c} + c)$$

În fig. C6.4a este reprezentat tabelul de adevăr al funcției împreună cu valorile asociate ale intrărilor de date ale multiplexorului, iar în fig. C6.4b implementarea funcției.

C6.2 Demultiplexorul / Distribuitorul de date

Demultiplexorul este circuitul logic combinațional care realizează operația inversă multiplexării, adică trimite semnalul prezent la intrarea unică la una din ieșirile sale și anume la cea selectată. Relația între **n**, numărul de ieșiri și **s** numărul de linii de selecție este $2^s \geq n$.



Schema de principiu a demultiplexorului (**DMUX**) este cea din fig. C6.5a. Comutatorul K selectează una din cele n ieșiri (Y_1, \dots, Y_n) căreia îi trimite semnalul de la intrarea I . În fig. C6.5b este reprezentată schema bloc a demultiplexorului 1-în- n în care apar și cele s intrări de selecție. În fig. C6.5c este reprezentată schema unui demultiplexor 1-în-4 în care D este semnalul de intrare date, E semnal de validare, A și B semnale de selecție iar $Y_0 \dots Y_3$ ieșirile circuitului. Relația ce descrie funcționarea acestui circuit este $Y_i = (m_i D) E$. Deoarece între această relație și cea care definește funcționarea decodificatorului există similitudine, rolul de DMUX poate fi jucat de un decodificator. Spre exemplu funcționarea circuitului 74138 (prezentat în figura C5.13) este dată de relația $Y_i = m_i (G1 \overline{G2} A \overline{G2} B)$. Făcând $\overline{G2} A = 0, \overline{G2} B = 0$ prin legare la masă și utilizând pe $G1$ pe post de intrare de date acest circuit va emula demultiplexorul 1-în-8.

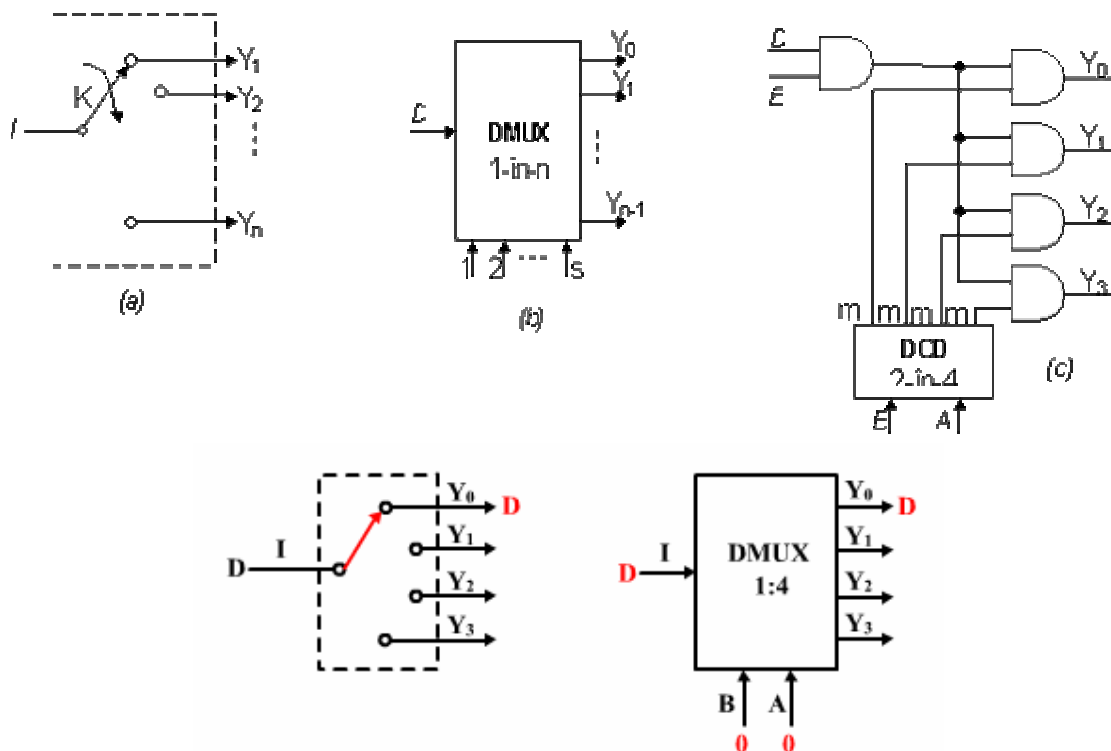


Fig. C6.5 Demultiplexorul: (a) schema de principiu; (b) schema bloc 1-în- n ; (c) schema circuitului 1-în-4; (d) funcționarea DMUX 1-în-4

EXEMPLUL C6.2

Să se realizeze un circuit de multiplexare / demultiplexare a 8 căi cu ajutorul circuitelor 74138 și 74151A.

În fig. C6.6 este ilustrat circuitul propus în care se poate observa că pentru transmisia a 8 semnale la distanță sunt necesare doar 4 linii, una pentru date și 3 pentru semnalele S_0, S_1 și S_2 ce formează adresa de selecție.

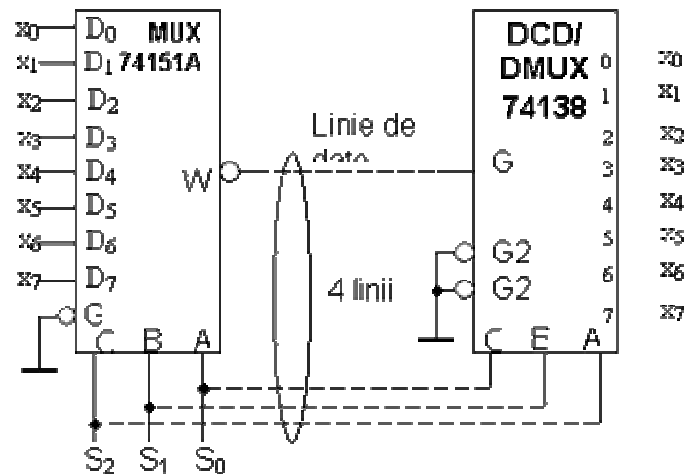


Fig. C6.6 Circuit de multiplexare / demultiplexare a 8 căi.

6.3 Circuite sumatoare

A. Semisumatorul elementar este un circuit logic combinațional (CLC) cu ieșiri multiple care adună 2 biți de date și obține la ieșire un bit sumă și un bit de transport. În fig. C6.7a este reprezentată schema bloc a unui semisumator elementar, iar în fig. C6.7b tabelul de adevăr ce descrie funcționarea acestui circuit.

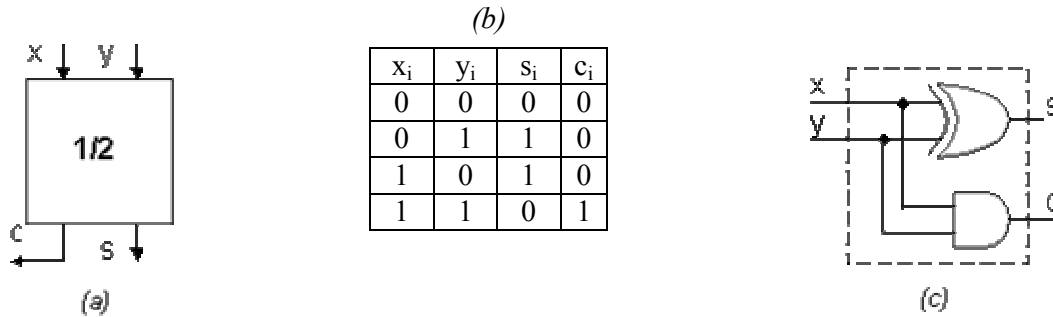


Fig. C6.7 Semisumatorul binar elementar: (a) Schema bloc; (b) Tabelul de adevăr; (c) Schema circuitului

Expresiile funcțiilor extrase din tabel sunt $s_i = x_i \oplus y_i$, respectiv $c_i = x_i y_i$ ce pot fi implementate într-un circuit ca cel din fig. C6.7c.

B. Sumatorul elementar complet este un CLC cu ieșiri multiple care adună 3 biți, 2 biți de date și unul de transport intrare și obține un bit sumă și unul de transport ieșire. În fig. C6.8a este reprezentată schema bloc a circuitului iar în fig. C6.8b tabelul de adevăr din care se extrag expresiile sumei $s_i = x_i \oplus y_i \oplus c_{i-1}$ și transportului de ieșire $c_i = x_i y_i + x_i c_{i-1} + y_i c_{i-1}$, ce pot fi implementate într-o rețea ca cea din fig. C6.8c.

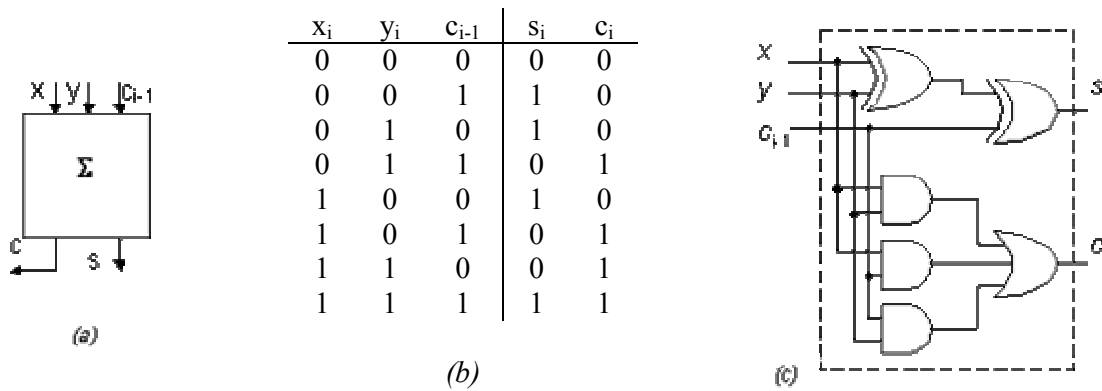


Fig. C6.8 Sumatorul elementar complet: (a) Schema bloc; (b) Tabelul de adevăr; (c) Schema circuitului.

C. Sumatorul pseudoparalel este un CLC care utilizează $n-1$ sumatoare elementare complete și un semisumator elementar, așa cum este ilustrat în fig. C6.9a, pentru a aduna 2 operanzi cu dimensiunea de n biți, în același timp.

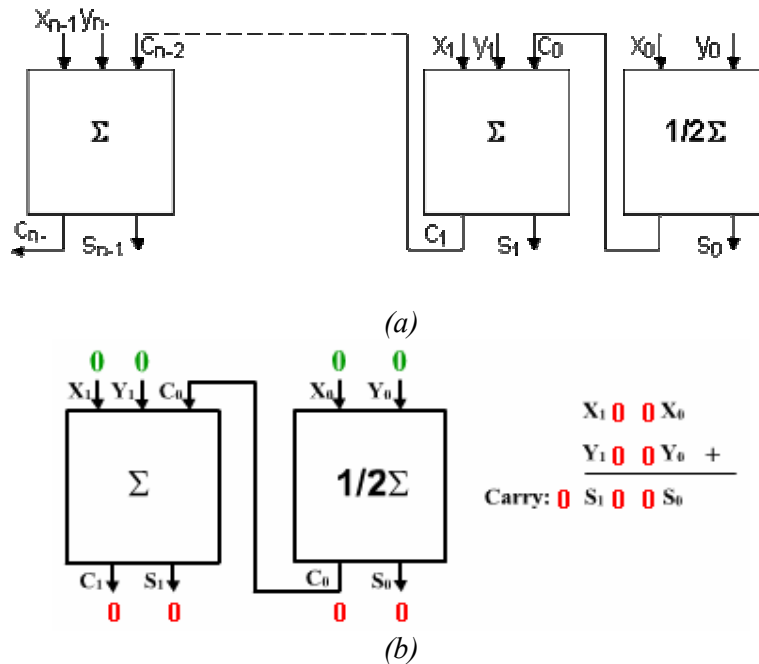


Fig. C6.9 (a) Sumatorul pseudoparalel; (b) Sumatorul de 2 biți

Circuitul se numește **pseudoparalel** deoarece, deși cei doi biți de date se adună simultan, transportul trebuie să se propage de la rangul cel mai mic spre cel mai mare. Se mai numește și **sumator cu transport propagat** și este un sumator lent.

Acest tip de sumator este implementat în circuitele integrate **7482**, sumator pseudoparalel pe 2 biți și **7483**, pe 4 biți, cu care se pot realiza sumatoare de cuvinte cu lungime mai mare (8, 16, etc. biți). În fig. C6.10 este reprezentată configurația pinilor pentru 7483. C_0 este bitul de transport intrare (la rangul cel mai mic), C_4 – bitul de transport de ieșire (de la rangul cel mai mare), $\Sigma_1, \Sigma_2, \Sigma_3$ și Σ_4 sunt biții sumă. Biții de transport C_4 C_4 C_4 sunt interni circuitului, nu au conexiuni în exterior.

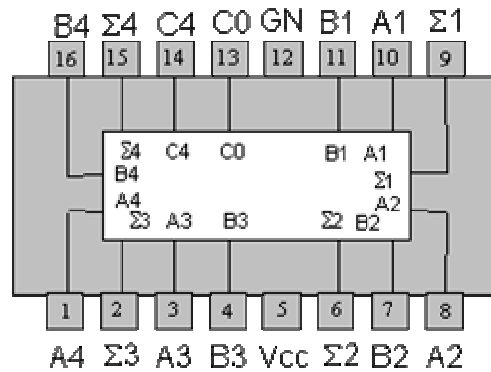


Fig. C6.10 Configurația pinilor sumatorului pseudoparalel pe 4 biți 7483.

Sumatoare de mare viteză

S-a impus realizarea unor astfel de sumatoare pentru a crește viteza de procesare a sistemelor. La aceste sumatoare, pe lângă utilizarea porților cu timpi de propagare mari, s-au utilizat și structuri adecvate acestui scop.

D. Sumatorul paralel complet este proiectat să adune strict în paralel și este cel mai rapid sumator. La intrări se aduce simultan transportul și biții operanzilor. După cum rezultă din expresiile funcțiilor transportului pentru primele 3 ranguri:

$$C_0 = x_0 y_0$$

$$C_1 = x_1 y_1 + (x_1 \oplus y_1)(x_0 y_0)$$

$$C_2 = x_2 y_2 + (x_2 \oplus y_2)(x_1 y_1) + (x_2 \oplus y_2)(x_1 \oplus y_1)(x_0 y_0)$$

implementarea presupune utilizarea unui număr foarte mare de porți logice, din ce în ce mai multe pentru rangurile mai mari, lucru care se dovedește a fi nepractic.

E. Sumatorul paralel cu transport anticipat, realizează un compromis între sumatorul pseudoparalel și sumatorul paralel complet. El împarte sumatorul în module, calculează transportul pentru fiecare rang și îl aduce apoi la intrarea corespunzătoare a sumatoarelor elementare în același timp. Dacă se notează $G_i = x_i y_i$ (transport generat) și $P_i = x_i \oplus y_i$ (transport propagat), relațiile de mai sus se pot rescrie astfel

$$C_0 = G_0, \quad C_1 = G_1 + P_1 G_0, \quad C_2 = G_2 + P_2 G_1 + P_2 P_1 G_0$$

Se identifică două circuite specializate: *sumatorul* și *generatorul de transport anticipat*, reprezentate în fig. C6.11a respectiv b, care implementează relațiile de mai sus și $S_i = x_i \oplus y_i \oplus C_{i-1} = P_i \oplus C_{i-1}$.

În fig. C6.11c este reprezentată schema bloc a unui modul de adunare cu transport anticipat. Acest tip de sumator este implementat în circuitul integrat logic **74182**.

F. Sumatorul cu salvarea transportului este utilizat în cazul adunării unor șiruri de numere. La acest sumator se salvează transportul rezultat din însumarea a 2 operanzi într-un bistabil (celulă elementară de memorare) care se pune apoi la dispoziția următoarei însumări, realizându-se astfel o adunare în manieră pseudoparalelă.

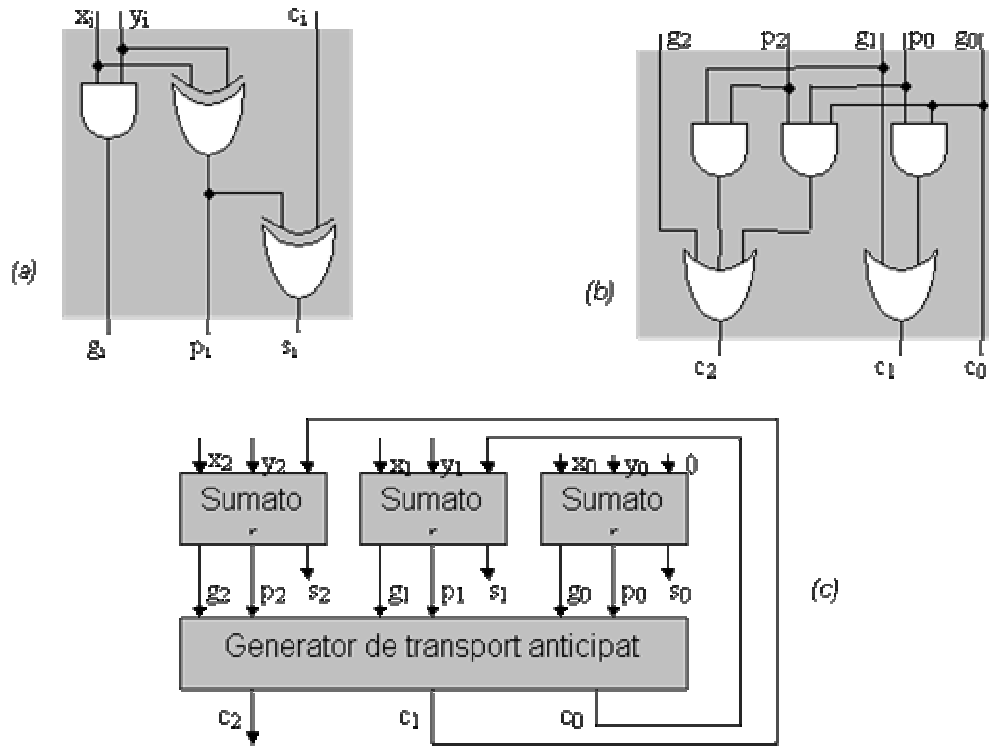


Fig. C6.11 Sumatorul paralel cu transport anticipat: (a) Modulul sumator; (b) Modulul generator de transport; (c) Schema bloc.