

ANEXA 4. OPERAȚII ARITMETICE – IMPLEMENTĂRI

ADUNAREA ÎN BINAR: $A + B$

Adunarea a două numere de câte N biți va furniza un rezultat pe N+1 biți.

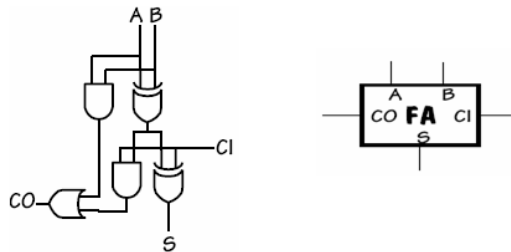


Figura1. Anexa4. Sumator binar complet

Schema bloc a unui sumator pentru cuvinte de 4 biți:

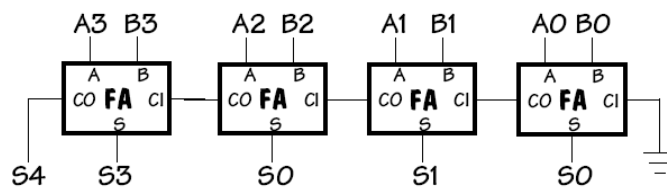


Figura2. Anexa 4. Sumator pentru cuvinte de 4biți.

SCĂDEREA ÎN BINAR: $A - B = A + (-B)$

Folosind complementul față de 2: $-B = \sim B + 1$ (\sim este complementul la nivel de bit).



Schema bloc a unui sumator-scăzător pe 4 biți. Selectarea operației este asigurată de semnalul de comandă *Scădere*.

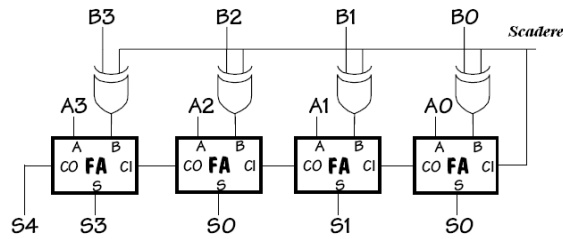


Figura3. Anexa 4. Sumator-scăzător pe 4 biți.

Codificarea Condițiilor:

Este de dorit ca, în afară de sume, să se obțină și alte informații privind rezultatul adunării:

- **Z (zero):** rezultatul este = 0. $Z = \sim(| (S_{N-1}, \dots, S_0))$; Poarta NOR cu N+1 intrări.
- **N (negativ):** rezultatul este < 0, S_{N-1} .
- **C (transport):** transportul la stânga rangului de semn: S_N .
- **D (depașire):** rezultatul depășește capacitatea de reprezentare corectă în gamă:

-2^{N-1} până la $2^{N-1} + 1$ (în codul complementar)

$$D = \text{Cout}_{N-1} \oplus \text{Cin}_{N-1}$$

Timpul de propagare t_{pD} pentru sumatorul cu transport succesiv:

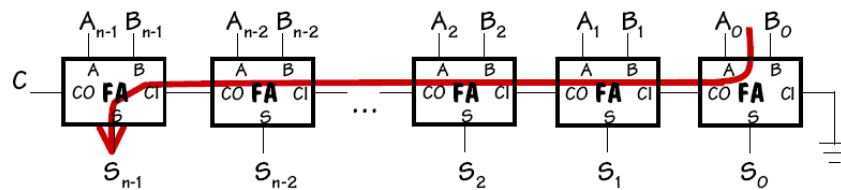
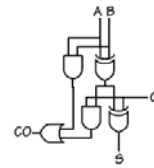


Figura4. Anexa 4. Timpul de propagare t_{pD} pentru sumatorul cu transport succesiv.

Cazul cel mai defavorabil este acela când se adună 11111....11 cu 00000....01, care presupune propagarea transportului pe întreg lanț de sumatoare complete.

$$t_{PD} = (N-1) * \underbrace{(t_{PD,OR} + t_{PD,AND})}_{CI \text{ la } CO} + \underbrace{t_{PD,XOR}}_{CI_{N-1} \text{ la } S_{N-1}} \approx \Theta(N)$$



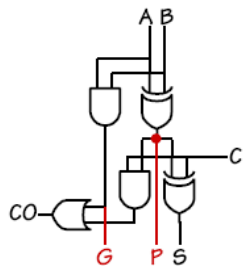
$\Theta(N)$ se citește ca “ordin N” și arată că latenta sumatorului crește direct proporțional cu numărul de biți ai operanzilor

Logica mai rapidă pentru transport:

Se rescriu ecuațiile pentru C_{OUT} :

$$\begin{aligned} C_{OUT} &= AB + AC_{IN} + BC_{IN} \\ &= AB + C_{IN}(A + B) \\ &= G + PC_{IN}, \quad \text{unde: } G = AB \text{ (Generează), iar } P = A + B \text{ (Propagă)} \end{aligned}$$

De regulă se utilizează expresiile: $P = A \oplus B$, iar $S = P \oplus C_{IN}$



Pentru adunarea a două numere de N biți:

$$\begin{aligned} C_N &= G_{N-1} + P_{N-1}C_{N-1} \\ &= G_{N-1} + P_{N-1}G_{N-2} + P_{N-1}P_{N-2}C_{N-2} \\ &= G_{N-1} + P_{N-1}G_{N-2} + P_{N-1}P_{N-2}G_{N-2} + \dots + P_{N-1}\dots P_0C_{IN} \end{aligned}$$

C_N se obține cu o întârziere introdusă de 3 porți:

- generare P/G;
- produs logic AND;
- sumă logică finală OR;

ADUNAREA PE N BIȚI ÎN TIMP CONSTANT

Dacă s-ar utiliza porți cu (N+1) intrări și nu ar interesa aspectele privind încărcarea în ceea ce privește semnalele P, întârzierea propagării transportului într-un sumator, construit pe baza ecuațiilor P/G, pentru a obține C_{IN} , pentru fiecare bit, ar fi egală cu:

4 întârzieri la nivel de poartă $\approx \Theta(1)$, ceea ce nu este practic pentru $N > 4$.

Pentru implementarea adunării pe N biți în timp constant s-au conturat următoarele soluții:

- transport succesiv rapid;
- sumatoare cu transport anticipat ierarhic.

SUMATOARE CU TRANSPORT ANTICIPAT

Se poate construi un lanț ierarhic pentru transportul anticipat, prin generalizarea definiției pentru logica *Generează Transport/Propaga Transport*: (Logica GP).

Sumatorul va fi divizat în două părți: partea superioară H (Higher part) și partea inferioară L (Lower part).

Funcția GP se poate exprima astfel:

$G_{HL} = G_H + P_H G_L$ (Se generează transport dacă partea superioară generează transport sau dacă partea inferioară generează transport și partea superioară propagă transport);

$P_{HL} = P_H P_L$ (Se propagă transport dacă partea inferioară și partea superioară propagă transport).

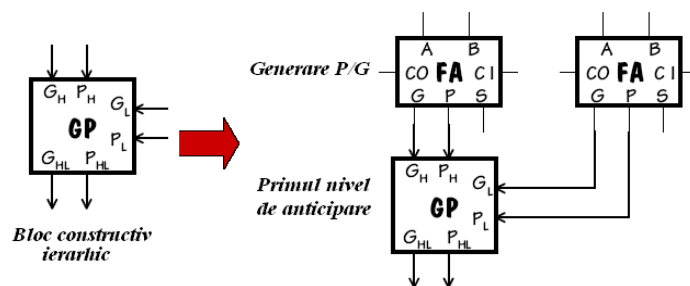


Figura5. Anexa 4. Sumator cu transport anticipat

Transport anticipat pe 8 biți (Generare GP)

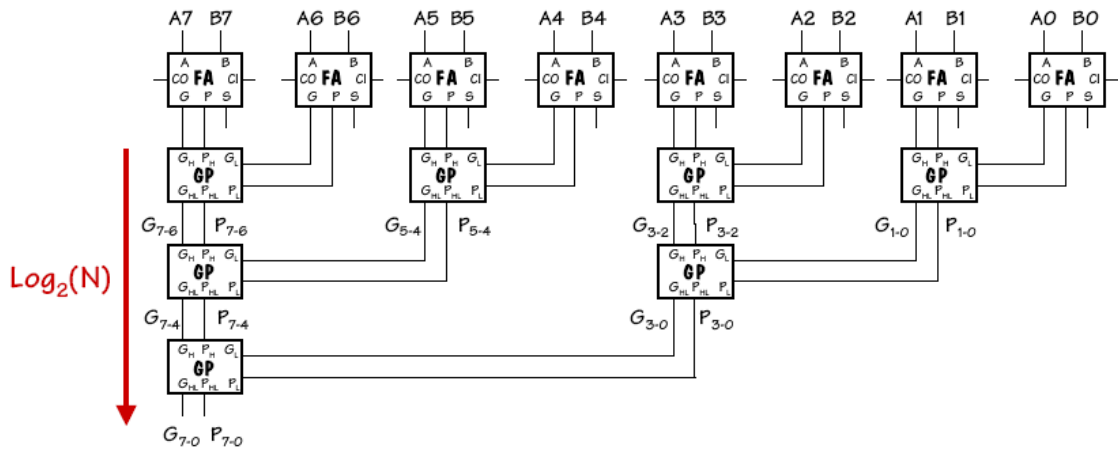


Figura 6. Anexa 4. Evidențierea transportului anticipat.

Se poate construi un arbore de unități GP, pentru a stabili logica de generare și propagare a transportului în cazul unui sumator cu o dimensiune dată. Pentru un sumator pe 2^N biți sunt necesare 2^{N-1} unități GP.

$$C = \underbrace{G_7 + P_7G_6 + P_7P_6G_5 + P_7P_6P_5G_4 + \dots + P_7P_6\dots P_0C_{IN}}_{G_{7-0}} \underbrace{\phantom{G_7 + P_7G_6 + P_7P_6G_5 + P_7P_6P_5G_4 + \dots + P_7P_6\dots P_0C_{IN}}}_{P_{7-0}}$$

Generator de transport anticipat pe 8 biți:

Dându-se valoarea transportului în bitul cel mai puțin semnificativ, se pot genera biții de transport pentru fiecare sumator: $c_j = G_{j-1} + P_{j-1}c_i$

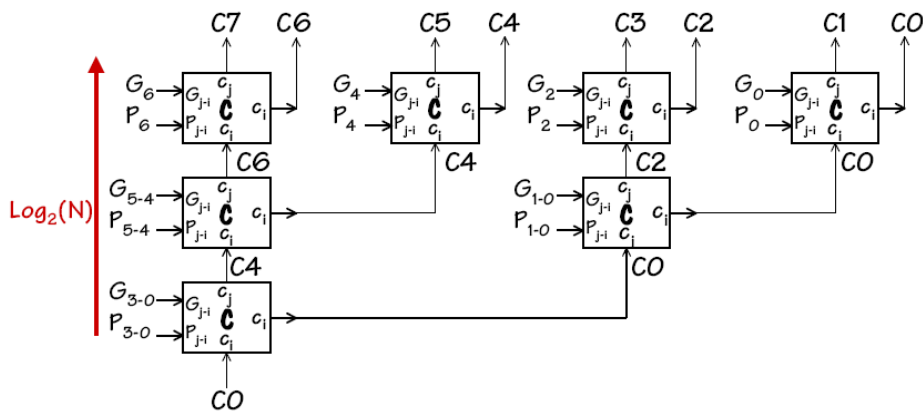


Figura 7. Anexa 4. Generarea transportului anticipat.

Schema completă a sumatorului pe 8 biți cu transport anticipat:

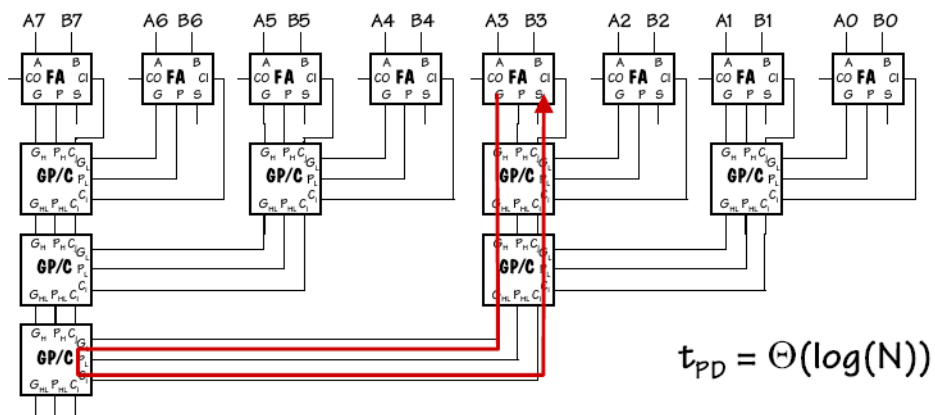


Figura 8. Anexa 4. Schema completă a sumatorului.

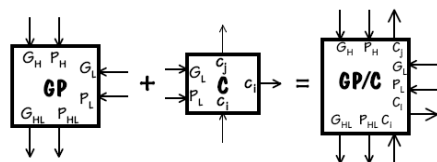
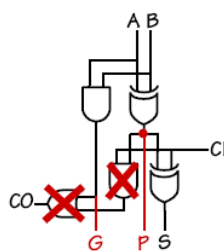


Figura 9. Anexa 4. Termenul de propagare și generare a transportului

Se poate observa că nu mai este necesară ieșirea C_0 de la fiecare sumator:



SUMATOARE CU TRANSPORT OCOLITOR:

Ecuțiile complete P/G sunt destul de complicate, dar P este relativ simplu. În consecință se poate folosi P, pentru a ocoli blocuri de sumatoare cu transport succesiv

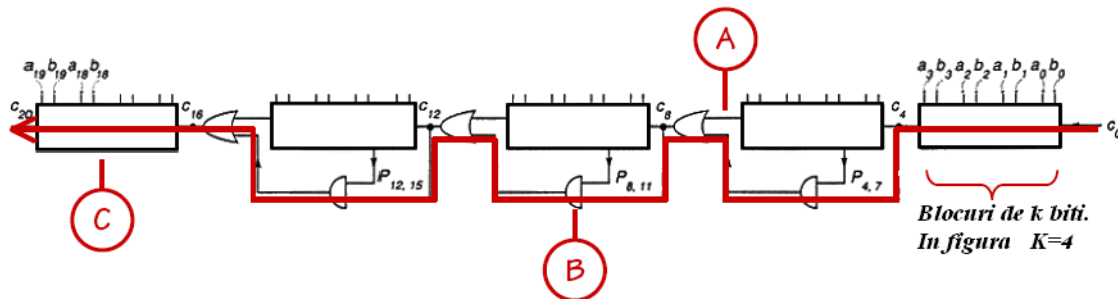


Figura 10. Anexa 4. Sumator cu transport ocolitor.

- (A) Transporturile se propagă succesiv la nivelul fiecarui bloc; dacă blocul generează un transport, acesta apare la ieșirile pentru transporturi ale aceluși bloc (similar cu G). Dacă transportul din afară (C_{IN}) este 0, la începutul operării, nu vor apărea transporturi false în afară.
- (B) Dacă C_{IN} și P_{BLOC} sunt simultan adevărate, transportul ocolește/sare blocul dat, către următorul bloc.
- (C) Transportul se propagă până la blocul final:

$$t_{PD} = 2 * [K + (N/K - 2) + K]$$

În condițiile unor blocuri de dimensiuni variabile:

$$t_{PD} \rightarrow \Theta(\sqrt{N})$$

SUMATOARE CU SELECTARE A TRANSPORTULUI:

Ideea are la bază utilizarea a două sumatoare: pentru unul se consideră că $C_{IN} = 0$, iar pentru celălalt că $C_{IN} = 1$. În continuare se utilizează multiplexoare pentru a selecta răspunsul corect, atunci când C_{IN} este cunoscut.

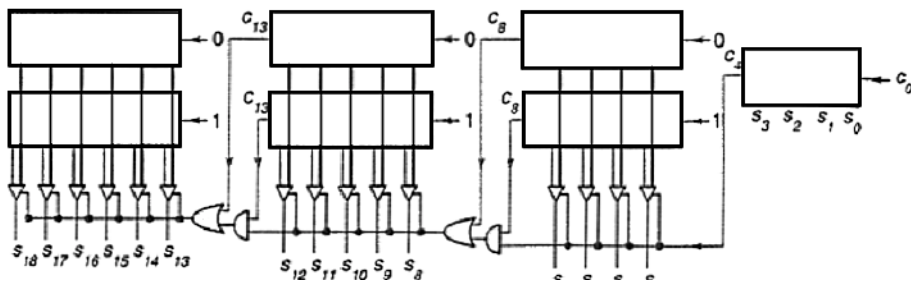


Figura 11. Anexa 4. Folosirea multiplexoarelor pentru selecție

Blocurile din stânga pot fi de dimensiuni mai mari deoarece au mai mult timp la dispoziție în cadrul propagării transportului.

Cu un singur etaj costul crește cu 50%, dar viteza se dublează față de cazul transportului succesiv. În cazul unor blocuri multiple, cu dimensiuni diferite:

$$t_{PD} \rightarrow \Theta(\text{sqrt}(N))$$

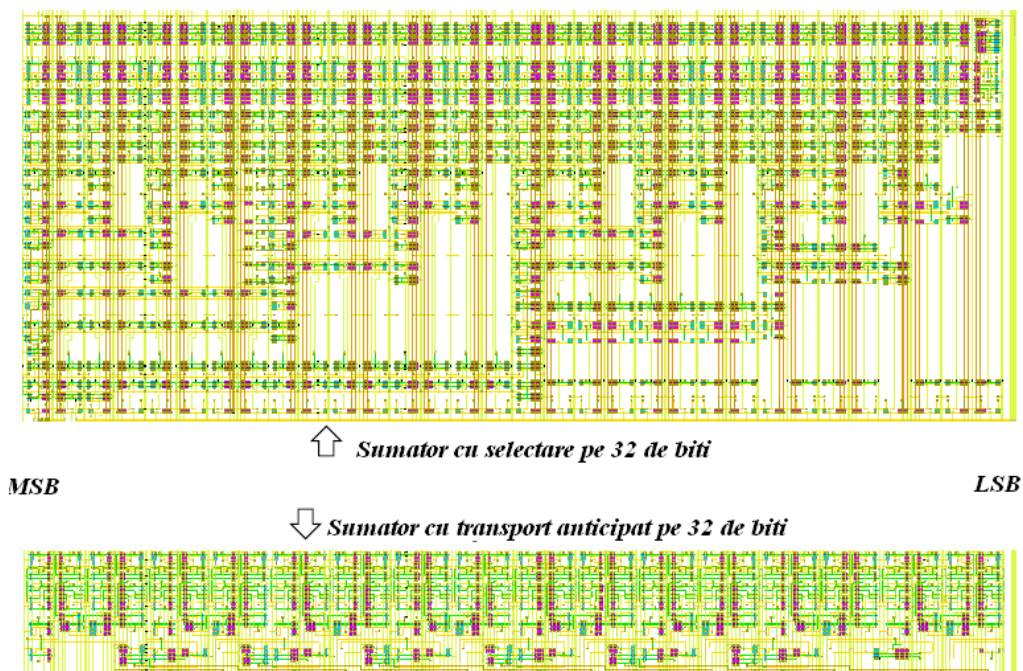


Figura 12. Anexa 4. Măști pentru sumatoare

ÎNMULȚIREA

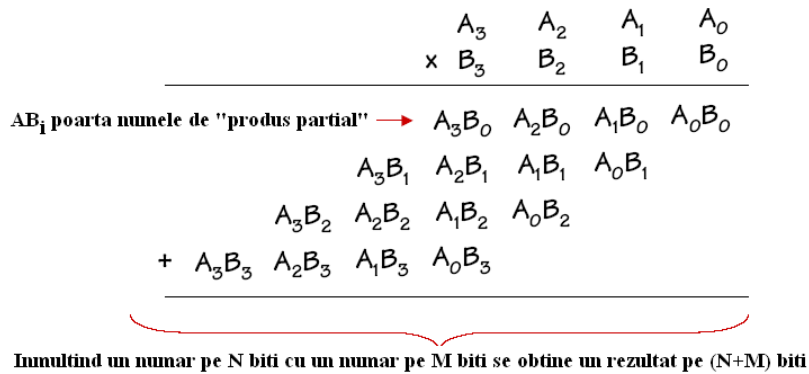


Figura 13. Anexa 4. Înmulțirea

Cel mai simplu se obțin produsele parțiale folosind porți AND, întrucât B_i este 0 sau 1.

Mai complicată este adunarea celor M produse parțiale de câte N biți

Multiplicatorul secvențial.

Se presupune ca Deînmulțitul (A) are N biți, iar Înmulțitorul (B) are M biți. Dacă se va utiliza un singur sumator pe N biți, atunci va trebui să se realizeze un circuit secvențial, care va prelucra, la un moment dat, un singur produs parțial, ciclând de M ori pentru a termina înmulțirea.

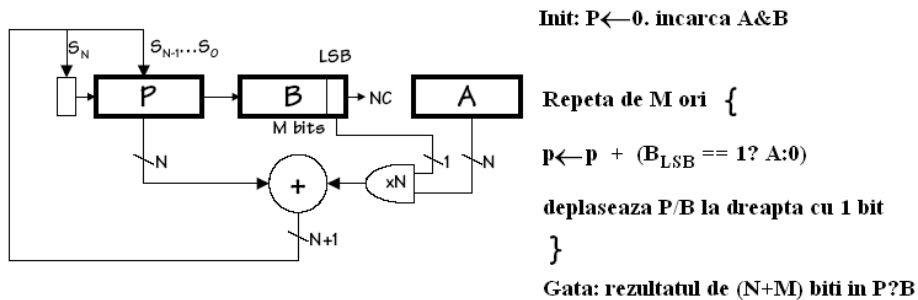
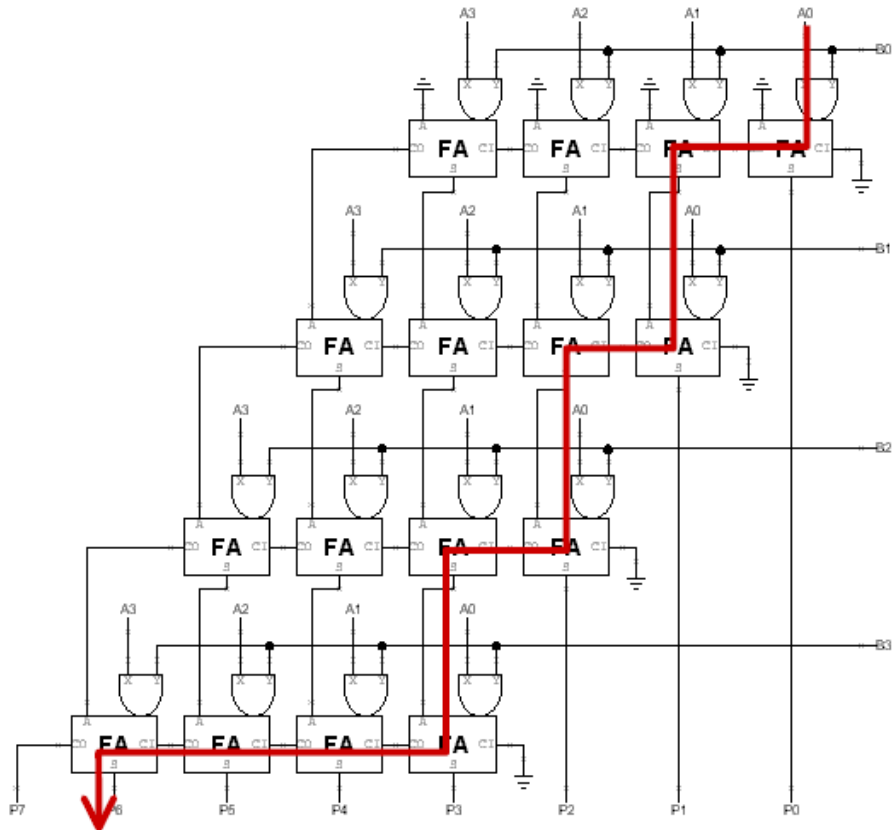


Figura 14. Anexa 4. Multiplicatorul secvențial

Multiplicator combinațional.

Acest dispozitiv asigură înmulțirea numerelor nonnegative.



$$t_{PD} = 10 * t_{PD,FA}$$

Figura 15. Anexa 4. Multiplicatorul combinațional

Multiplicator în Bandă de Asamblare.

Configurația de mai jos operează cu întregi nonnegativi.

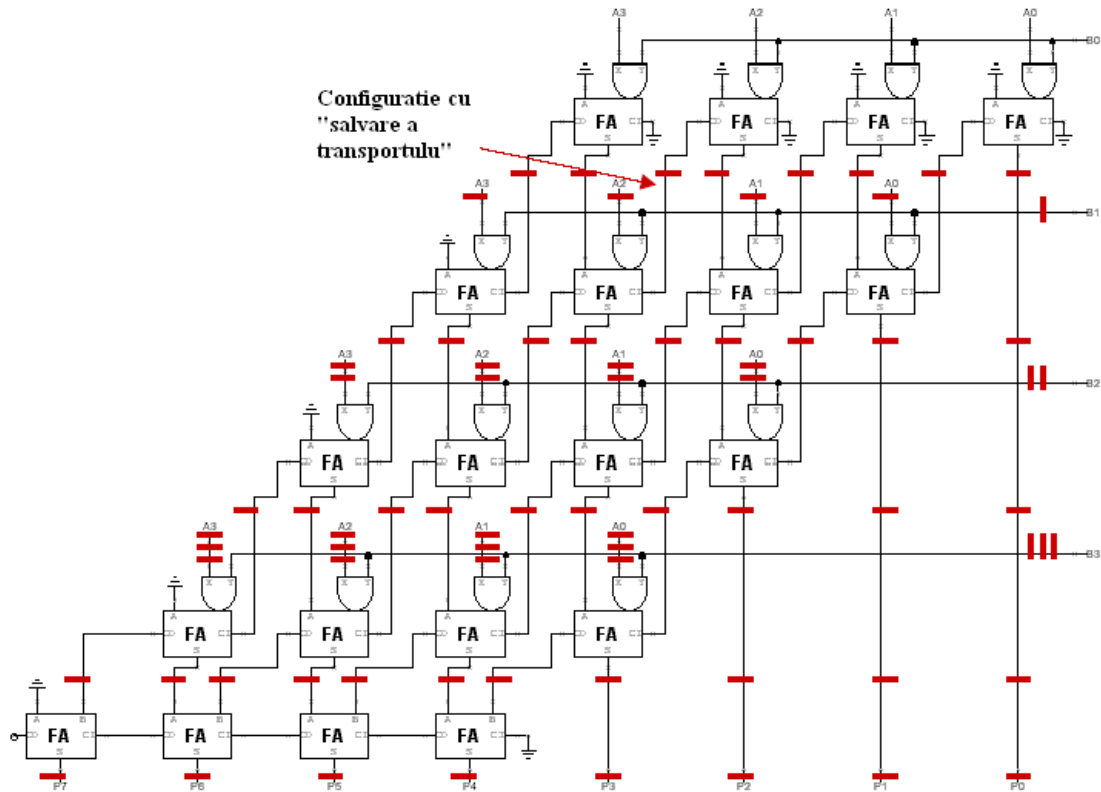


Figura 16. Anexa 4. Multiplicator în bandă de asamblare

În cazul unui etaj final cu transport anticipat:

$$t_{PD} = \Theta(\log(N))$$

ÎMBUNĂȚIREA LATENȚEI

O reprezentare abstractă a sumatorului cu salvare a transportului (CSA- Carry Save Adder) este dată mai jos:

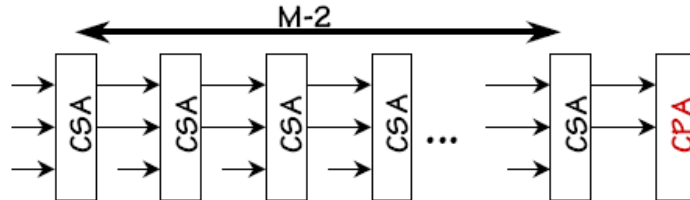


Figura 17. Anexa 4. Reprezentarea abstract a sumatorului CSA

Se efectuează o recablare astfel încât două sumatoare să lucreze în paralel. Rezultatele se vor forța în sumatoarele trei și patru, care operează, de asemenea, în paralel etc.

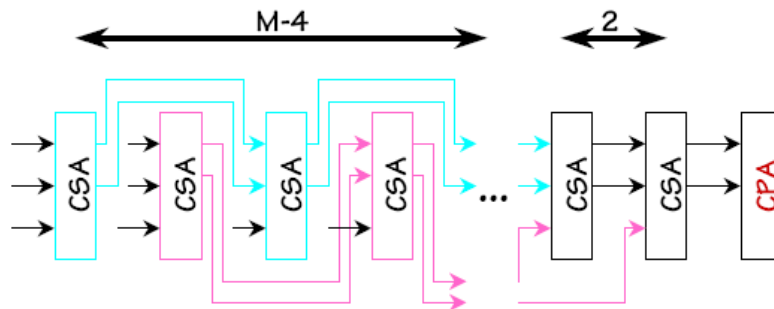


Figura 18. Anexa 4. Recablarea sumatoarelor

Fluxurile par și impar parcurg jumătate din sumatoare astfel, proiectele par/impar operează la o viteză de aproape două ori mai mare decât viteza unei implementări simple CSA.

O ALTĂ ÎMBUNĂȚĂȚIRE A LATENȚEI: ARBORELE WALLACE

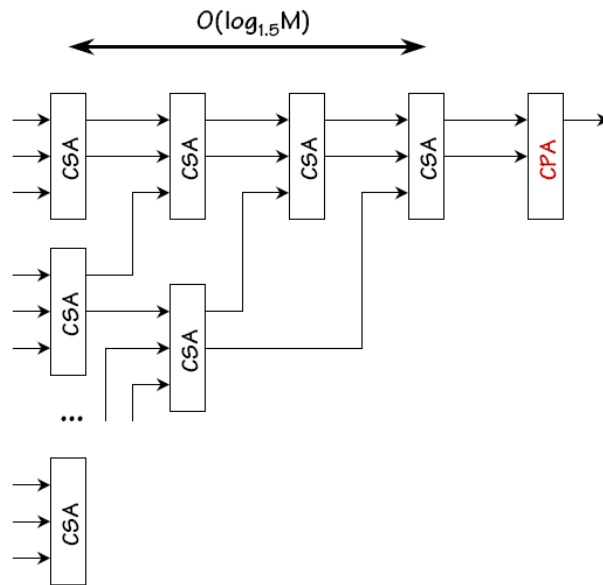


Figura 19. Anexa 4. Arborele Wallace

Până acum, în toate structurile de sumatoare s-au utilizat sumatoare complete cu 3 intrări și 2 ieșiri. Se pot utiliza sumatoare cu un “fan-in” mai mare pentru a reduce întârzierile, în cazul unui M mare. În literatura de specialitate s-au propus sumatoare cu 5 intrări și 3 ieșiri.