

---

# Lucrarea 12

## Sinteza Dispozitivelor de Împărțire Binară

Lucrarea prezintă succint principiile fundamentale ale algoritmilor de împărțire binară și modalitățile de implementare a unor structuri hardware corespunzătoare. Problema rezolvată arată maniera în care se poate manipula algoritmul astfel încât el să funcționeze pe operanzi de dimensiune arbitrară. Rezolvarea se constituie într-un exemplu semnificativ relativ la modul de abordare a problemelor de sinteză a dispozitivelor aritmetice: se pleacă de la construcția algoritmului seminumeric, este descrisă platforma hardware pe care se efectuează implementarea, urmând ca mai apoi să se realizeze implementarea propriu-zisă.

### 1 Dispozitivele de Împărțire Binară

Pentru a trata subiectul împărțirii binare vom folosi următoarele notații:

- $D$  – dividend (deîmpărțit)
- $V$  – divisor (împărțitor)
- $Q$  – quotient (cât)
- $R$  – reminder (rest)
- $D = Q \times V + R$
- $D \approx Q \times V$

$R$  este utilizat pentru sporirea capacității de apreciere a câtului. Dacă la înmulțire produsele parțiale se deplasează spre dreapta și avem în mod fix deînmulțitorul, la împartire deplasarea se face spre stânga și divizorul este fix.

Există două modalități de a efectua împărțirea:

- a) ordinary paper pencil division (cu creionul pe hârtie)
- b) computer implementation (algoritm care poate avea implementare hardware)

Dintre metodele de implementare în calculator, cele mai directe metode, cu implementare eficientă pe platformele hardware de tip Hayes sunt împărțirea cu și fără restaurarea restului:

Vom prezenta două exemple edificatoare în acest sens, unde deîmpărțitul este stocat în regiștrii  $A$  și  $Q$ , iar împărțitorul în registrul  $M$ . Câtul va fi conținut de registrul  $Q$ , iar restul se va afla, la terminarea algoritmului, în registrul  $A$ .

a.) restoring division (împărțire cu restaurarea restului)

A	Q	M
00000	1110	0001
00001	1110	
-00011		
<hr/>		
11110	1100	
+00011		
<hr/>		
000011		
<hr/>		
00001	1100	
00011	100	
-00011		
<hr/>		
00000	10001	
00001	001	
<hr/>		
-00011		
<hr/>		
+00011		
<hr/>		

```

00001
00010
-00011
-----
11111
+00011
-----
00010

```

b.) non restoring division (împărțire fără restaurarea restului)

```

      A      Q      M
00000   1110   00011
00001   110
-00011
-----
11110   1100
11101
+00011
-----
00000   1000
+00011
-----
00000   1001
-----
00001   001
-00011
-----
11110   010
+00011
-----
11111
+00011
-----
00010

```

Structura utilizată pentru implementarea acestor dispozitive este prezentată în Figura 12.1 la nivel de principiu (sunt utilizați regiștrii P, A și B a căror corespondență cu regiștrii din structura Hayes este indicată în figură).

Pentru o platformă hardware de tip Hayes, funcționarea este descrisă cu ajutorul următorului pseudocod HDL:

```

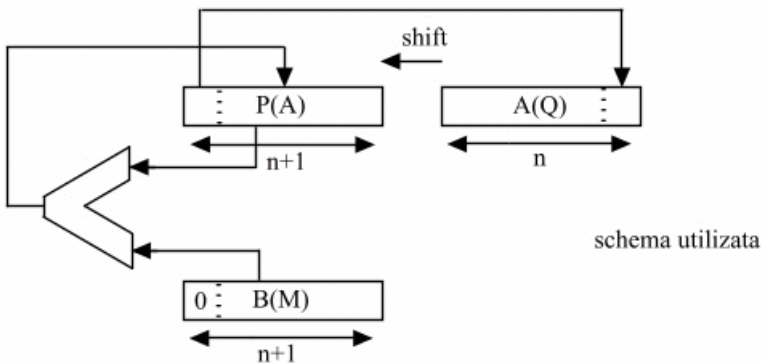
Non restoring division (in:inbus; out:outbus);
register S,A[n-1:0], Q[n-1:0], M[n:0], count[k:0];
bus inbus[n-1:0]; outbus[n-1:0];
Begin
count := 0, S = 0,

```

```

Input: A := inbus;
      Q := inbus;
      M := inbus, M[n] := 0;
Substract: SA := S.A - M;
Test: if s = 0 then
      Begin Q[0] := 1
            if count[n-1] = 1 then go to connection;
            else
            Begin
            count := count + 1,
            S.Q[n-1:1] := a.q;
            end
            S.A := S>A-M, go to Test;
            end
Output: outbus := Q;
        outbus := A
end non restoring divider.

```



**Figura 12.1** Shema de principiu a platformei hardware utilizată la implementarea operației de împărțire.

Implementarea pe structura hardware de tip Hayes este prezentată la nivel de schemă logică bloc în Figura 12.2.

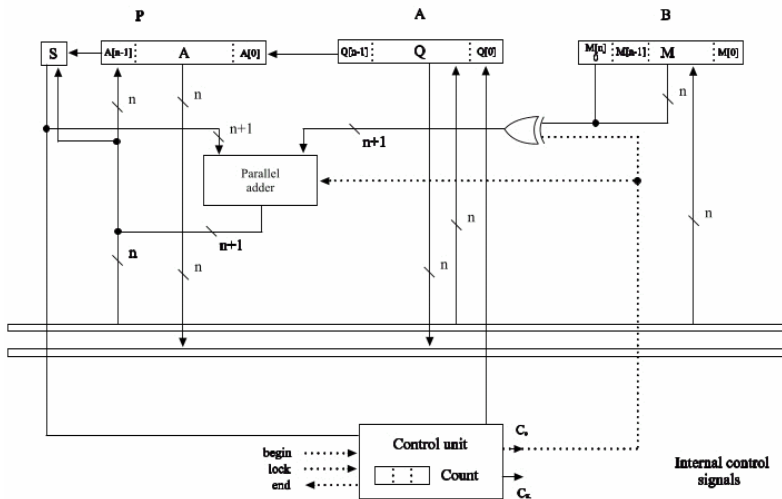


Figura 12.2 Structura platformei hardware de tip Hayes, necesară implementării operației de împărțire binară non-restoring.

## 2 Aplicații

### Problema 12.1 (propusă)

Se consideră procedura iterativă de împărțire fără restaurare a restului (non-restoring) pentru întregi fără semn. Această metodă presupune operarea cu un deîmpartit pe  $2n-1$  biți pentru un împărțitor pe  $n$  biți, datorită rolului special jucat de bitul cel mai puțin semnificativ al registrului Q. Să se modifice în mod corespunzător procedura de împărțire fără restaurare a restului, astfel încât să se poată opera cu un deîmpartit pe 16 biți, pentru un împărțitor pe 8 biți.

Se vor lua în considerație următoarele **indicații**:

1. Regiștrii A și Q se vor încărca exact în maniera clasică de la procedura fără restaurare a restului, bitul cel mai puțin semnificativ al deîmpărțitului încărcându-se într-un flag P (Parity).

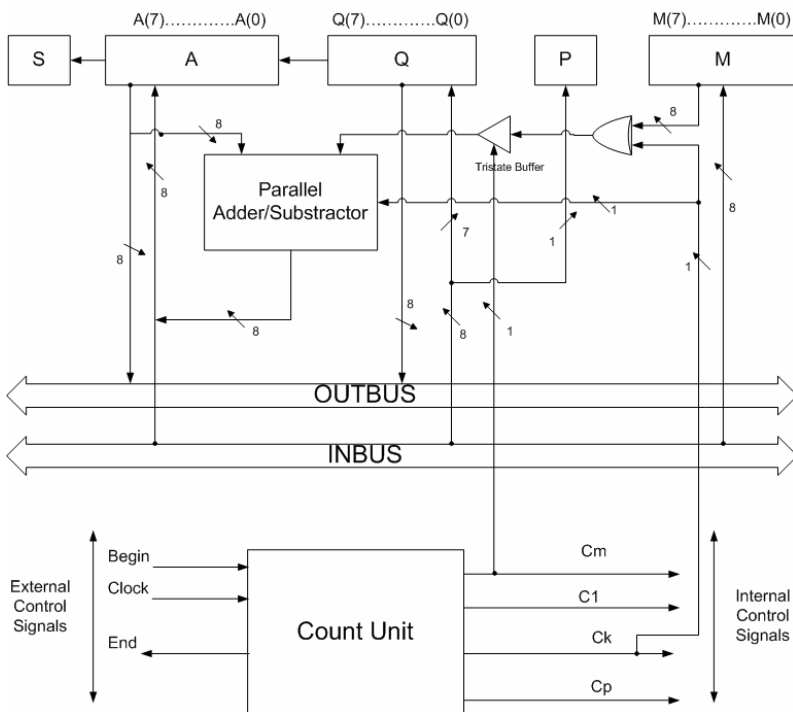
2. Procedura de împărțire va decurge identic cu procedura clasică, completându-se în mod corespunzător doar partea de corecție.
3. În cazul în care flag-ul P este 0, soluția este banală. Pentru  $P=1$  se vor folosi următoarele formule pentru corecție:

$$\text{rest\_corectat} = (2 \times \text{rest\_divizor}) + 1$$

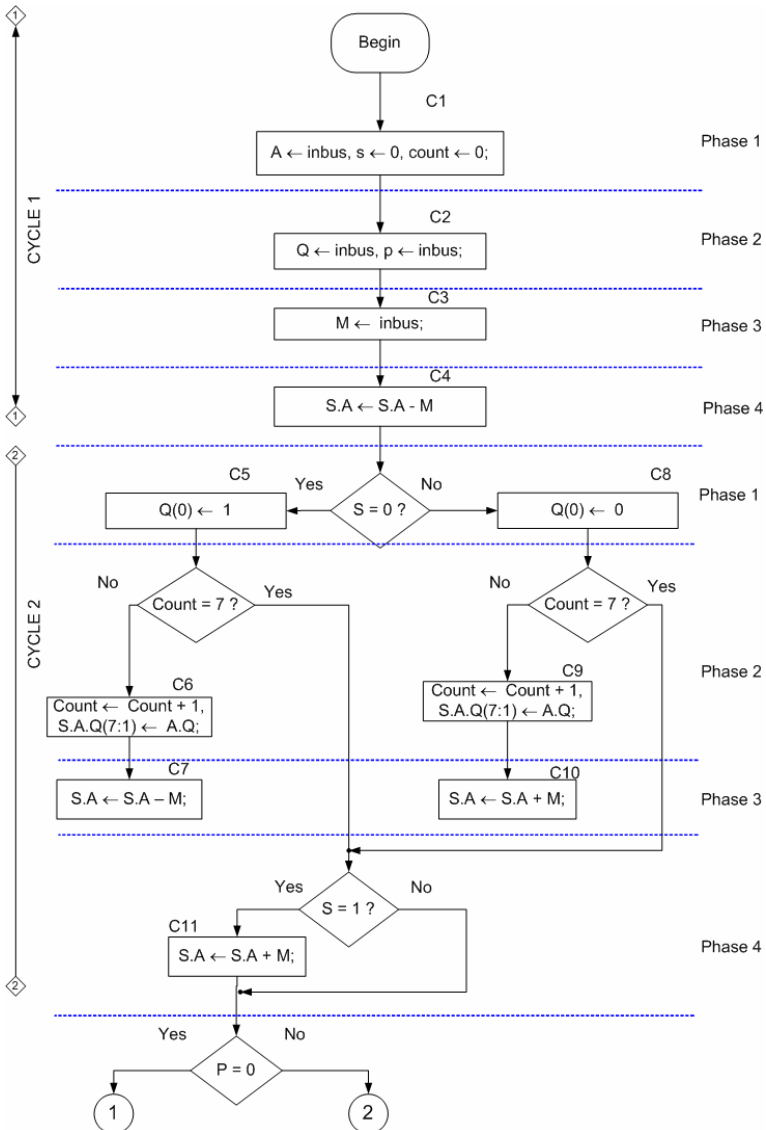
$$\text{cat\_corectat} = (2 \times \text{cât} + 1)$$

În acest context, se cer următoarele:

- a) explicarea procedurii, modificată corespunzător, precizând totodată și diagrama logică a structurii hardware folosită la implementare.
- b) ordinograma algoritmului modificat, corespunzător structurii hardware proiectate.
- c) implementarea unității de control prin metodele sequence counter și delay element.



**Figura 12.3** Schema bloc hardware, pentru Problema 12.1



**Figura 12.4 Ordinograma pentru Problema 12.1 (partea I).**

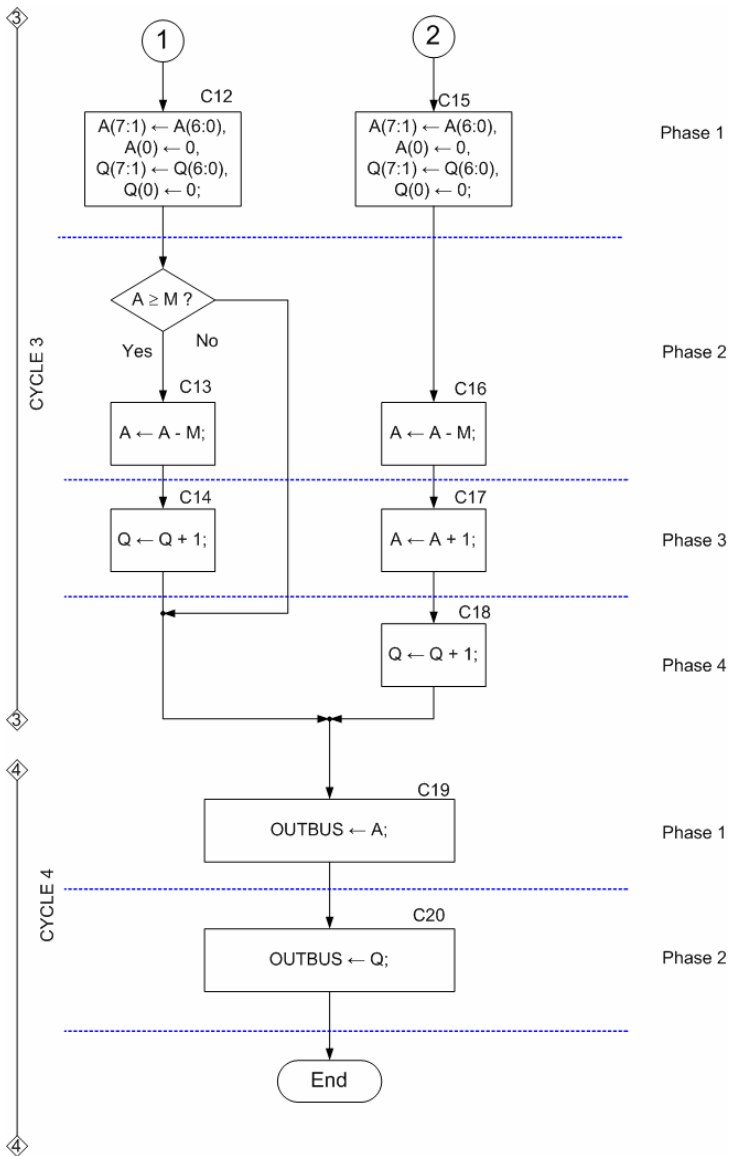
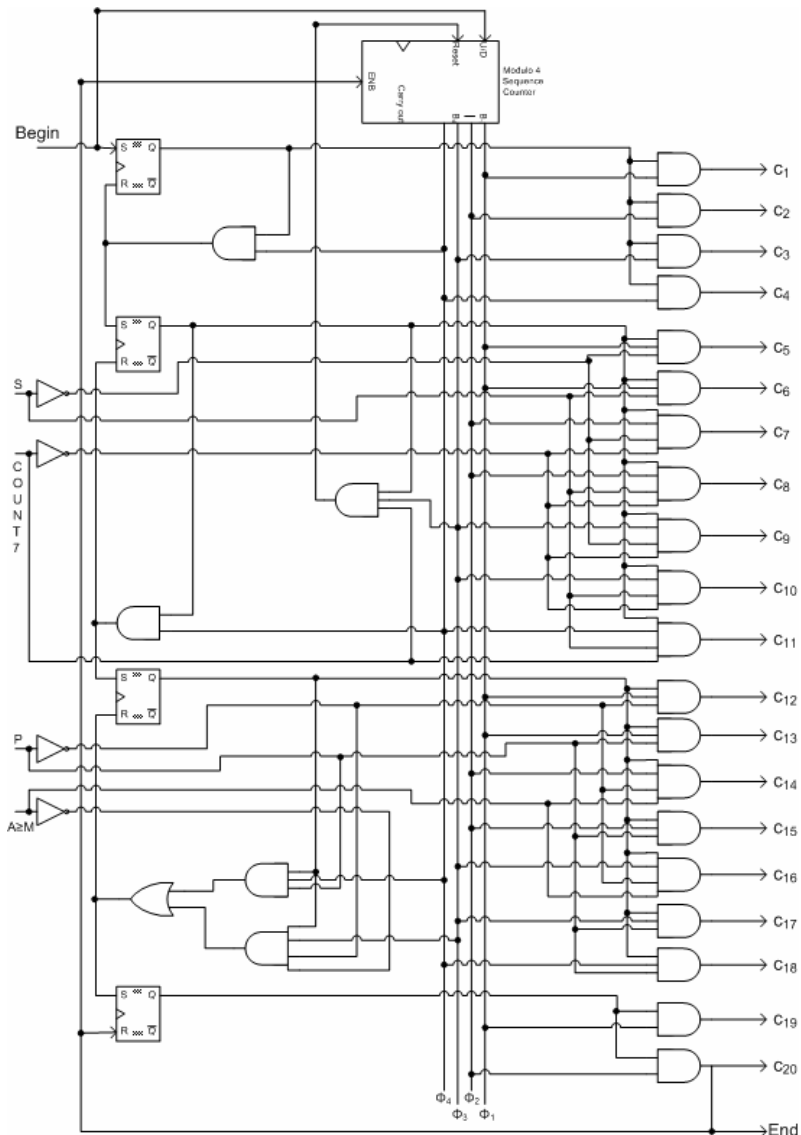


Figura 12.5 Ordinograma pentru Problema 12.1 (partea II).





**Figura 12.6** Implementarea unității de control prin metoda sequence counter.

Figura 12.3 prezintă structura hardware bazată pe platforma Hayes, revendicată de rezolvarea problemei 12.1. Ordinograma ce corespunde implementării algoritmului pe platforma hardware dată apare în Figurile 12.4 și 12.5, în timp ce Figura 12.6 prezintă implementarea unității de control din Figura 12.3 prin metoda sequence counter, folosind bistabile RS.

### **Problema 12.2 (propusă)**

Se dă operația aritmetică  $(X \times Y) \bmod Z$  cu operanzi pe 8 biți. Să se realizeze ordinograma aferentă acestei operații, prin compunerea înmulțirii Booth simplu cu împărțirea fără restaurare a restului. În aceeași ordine de idei, să se proiecteze structura hardware asociată acestei ordinograme (având ca bază platforma hardware Hayes) și să se indice semnalele de control atât pe structura hardware, cât și pe ordinogramă. Apoi, se cere sinteza unității de control pentru acest dispozitiv, prin metodele one-hot și sequence counter.