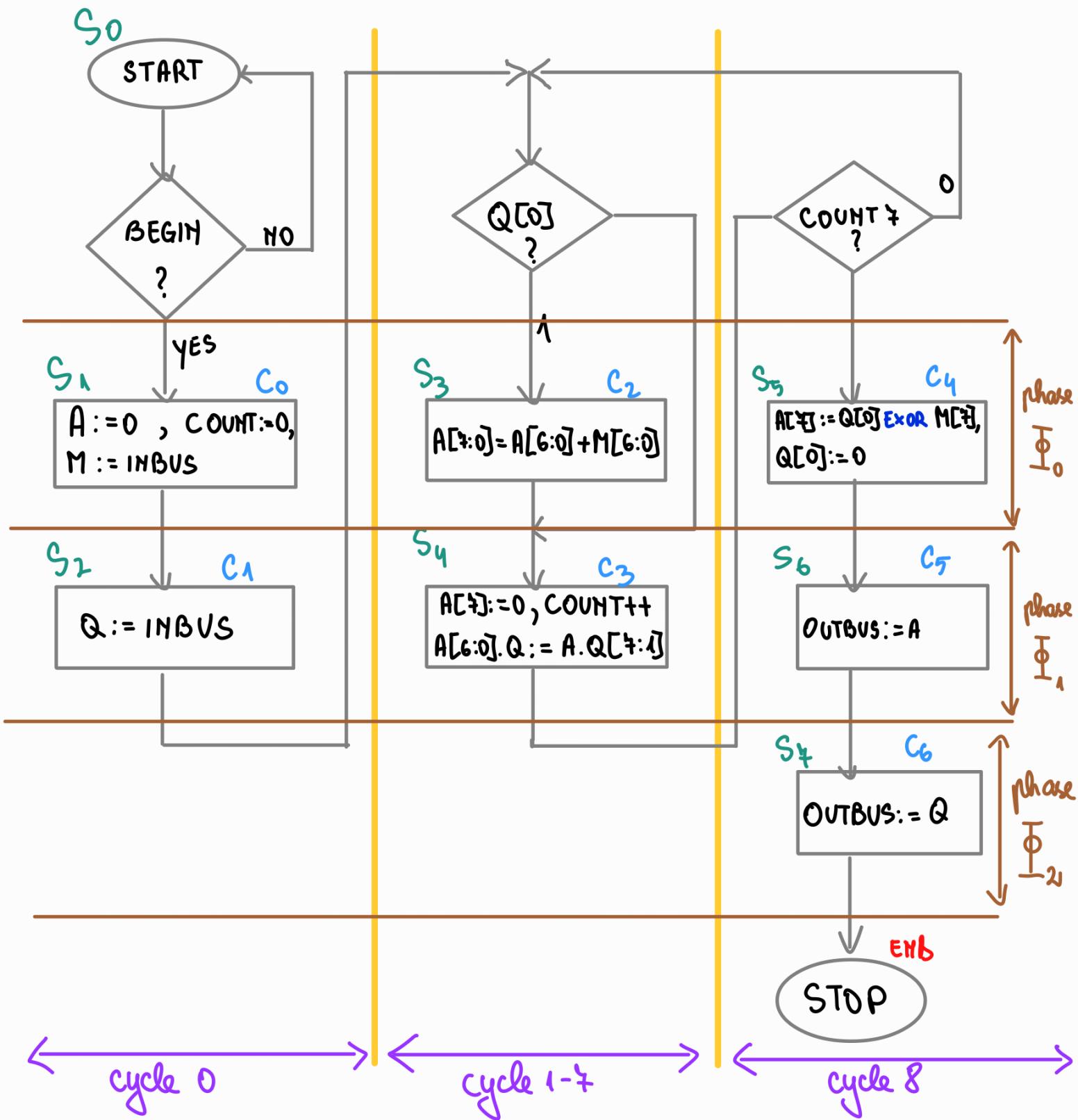


Arhitectura calculatoarelor

(curs 12 - S12)

4.3. Elemente de sinteză a unităților de control



A. Metoda tabelului de stare

→ avem 8 stări \Rightarrow 3 biți

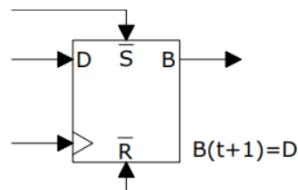
Stare	Codificare		
S_0	0	0	0
S_1	0	0	1
S_2	0	1	0
S_3	0	1	1
S_4	1	0	0
S_5	1	0	1
S_6	1	1	0
S_7	1	1	1

B. Metoda One Hot

→ utilizează un element de stocare pt. fiecare stare

→ doar un el. este activ (numit "hot")

Se vor utiliza bistabile de tip D:



Note: Ieșirea bistabilelor este notată B pentru a evita suprapunerea cu biți ai registrului Q .

Fiecare stare are asociat un bistabil \Rightarrow sunt folosite 8 variabile de stare: $B_0, B_1, B_2, B_3, B_4, B_5, B_6, B_7$; codificarea stării este prezentată în tabelul următor:

State	B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0
S_0	0	0	0	0	0	0	0	1
S_1	0	0	0	0	0	0	1	0
S_2	0	0	0	0	0	1	0	0
S_3	0	0	0	0	1	0	0	0
S_4	0	0	0	1	0	0	0	0
S_5	0	0	1	0	0	0	0	0
S_6	0	1	0	0	0	0	0	0
S_7	1	0	0	0	0	0	0	0

Ecuatii de feedback

$$b_0 = \beta_0 \cdot \overline{\text{BEGIN}} \text{ și } \beta_7 \quad (\text{pentru a mă reîntorc la start})$$

$$b_1 = \beta_0 \cdot \text{BEGIN}$$

$$b_2 = \beta_1$$

$$b_3 = \beta_2 \cdot Q[0] \text{ și } \beta_4 \cdot \overline{\text{COUNT7}} \cdot Q[0]$$

$$b_4 = \beta_2 \cdot \overline{Q[0]} \text{ și } \beta_3 \text{ și } \beta_4 \cdot \overline{\text{COUNT7}} \cdot \overline{Q[0]}$$

$$b_5 = \beta_4 \cdot \text{COUNT7}$$

$$b_6 = \beta_5$$

$$b_7 = \beta_6$$

Ecuatii de ieșire

$$C_0 = \beta_1$$

$$C_4 = \beta_5$$

$$C_1 = \beta_2$$

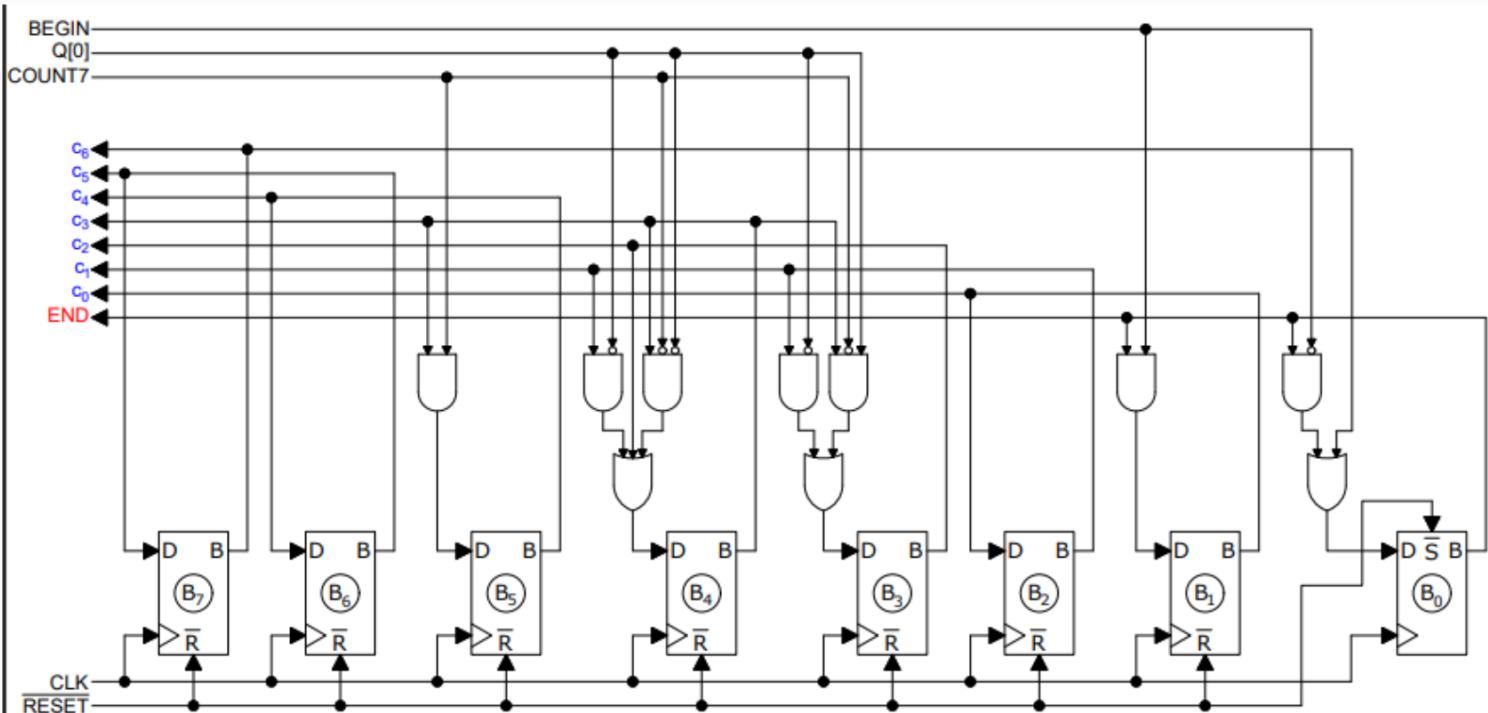
$$C_5 = \beta_6$$

$$C_2 = \beta_3$$

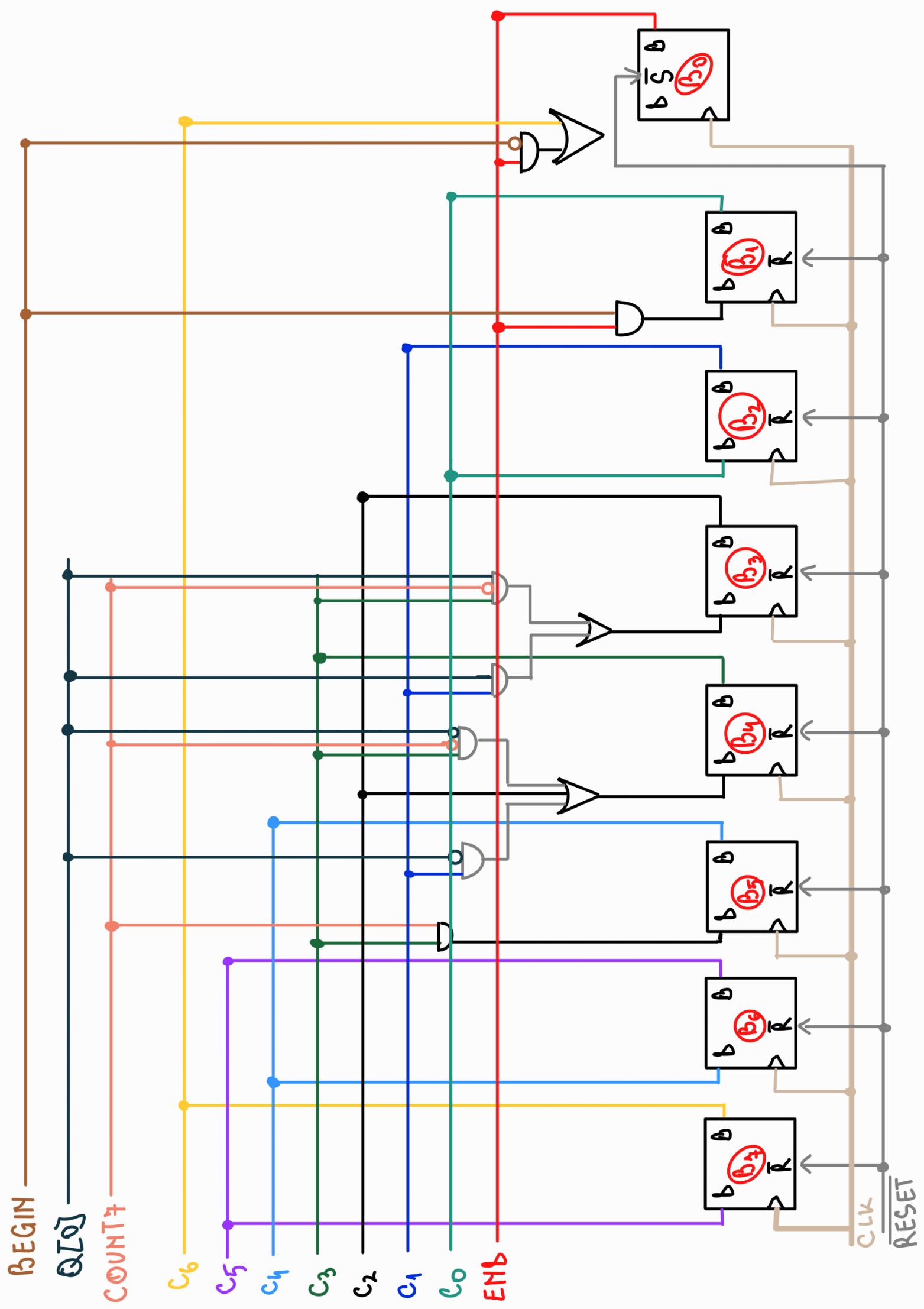
$$C_6 = \beta_7$$

$$C_3 = \beta_4$$

$$ENb = \beta_0$$



Notă: Arhitectura este afectată de **clock skew**

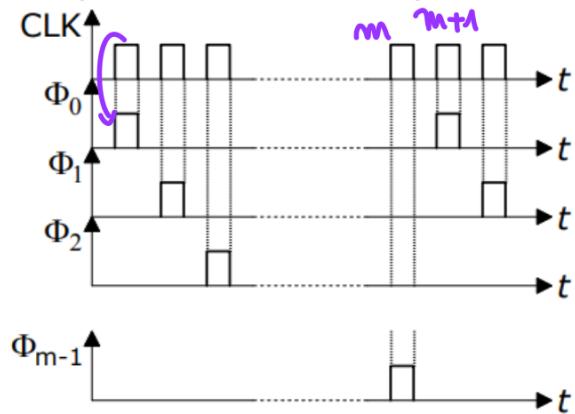


C.) Metoda Sequence Counter

→ extrage anumite impulzuri

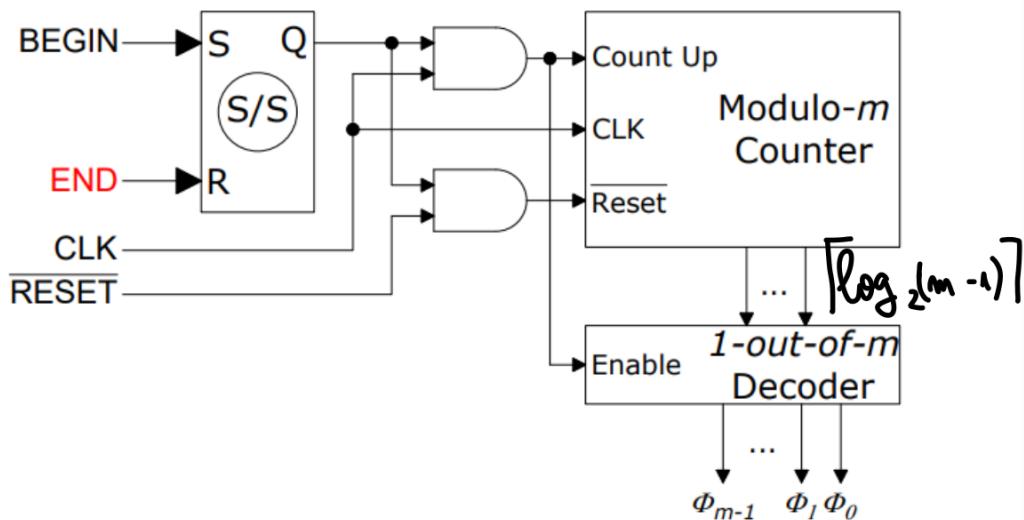
- construită în jurul unui Sequence Counter

- un numărător de secvență generează la ieșiri impulsuri de fază non-suprapuse ($\Phi_0, \Phi_1, \Phi_2, \dots, \Phi_{m-1}$) ca mai jos



- Impulsurile de fază sunt folosite pentru a activa semnale de control
- Datorită naturii repetitive a impulsurilor de fază, metoda Sequence Counter poate executa secvențe iterative

Numărătorul de secvență încorporează un numărător modulo- m , a cărui ieșiri sunt decodificate în m impulsuri de fază



S/S: Latch-ul Start/Stop controlează avansarea în secvență de numărare

$m - m$. de faze

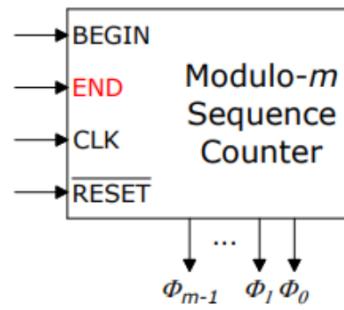
$R = 1 \rightarrow$ optim sistemul

→ mai avantajos numărator sincron

COUNT UP doar dacă $(Q \text{ și } CLK) = 1$

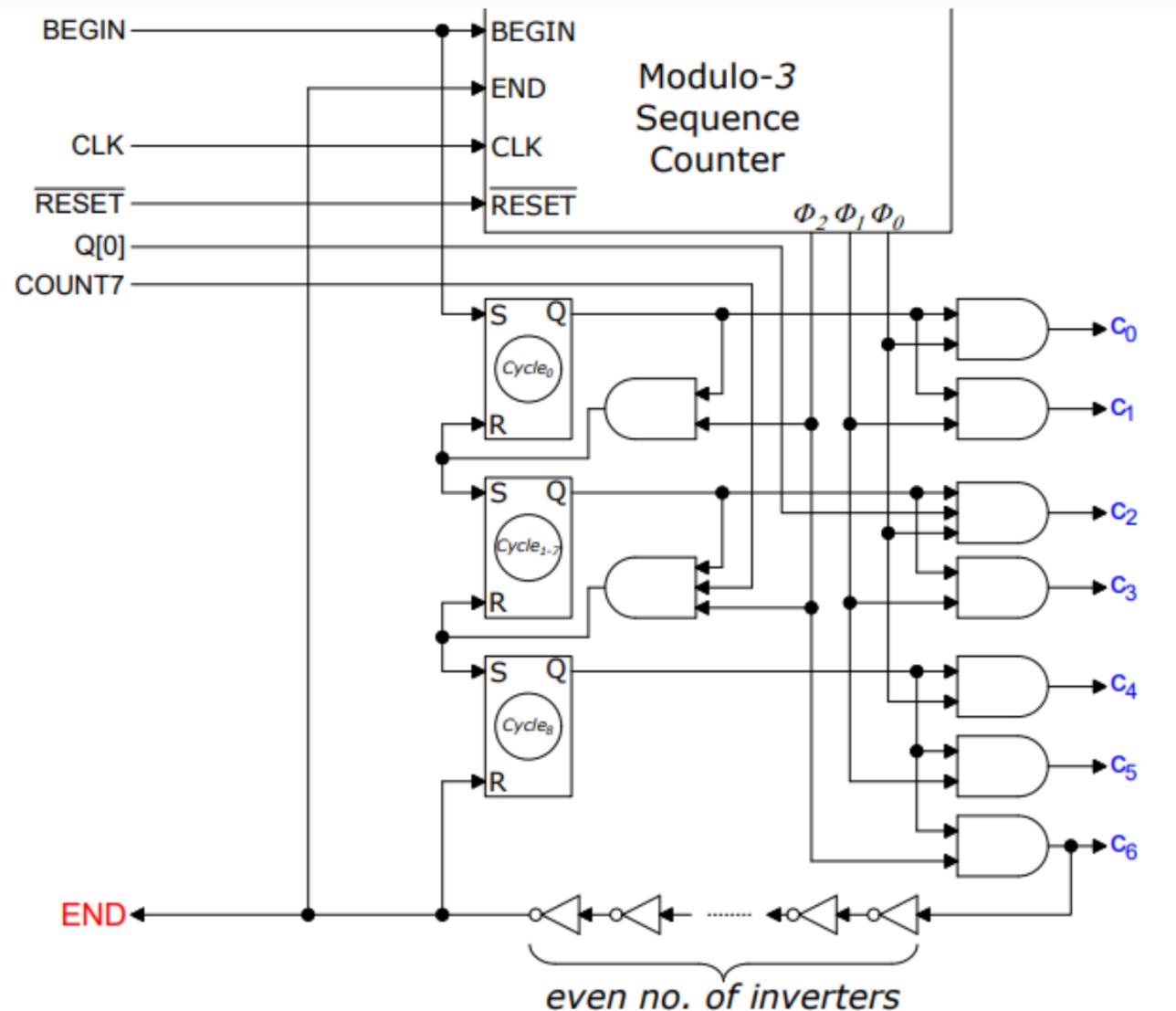
$Q \neq 1 \Rightarrow$ reset contor

Simbol numărător de secvență



Determinarea lui m (valoarea modulo a numărătorului):

- ▶ m este dat de cel mai lung ciclu
- ▶ pornind de la ordinogramă, se obține $m = 3$: numărătorul de secvență va avea 3 faze



- ▶ Utilizează cîte un latch *S/R* pentru fiecare ciclu al algoritmului
 - ▶ Latch-ul este activat când execuția algoritmului ajunge în ciclul asociat
- ▶ Semnalul *END* este obținut prin întârzierea semnalului de control *c₆* printr-un număr par de inversoare
 - ▶ activarea lui *END* determină dezactivarea lui *c₆*: dacă *c₆* nu rămîne activ suficient de mult timp pentru terminarea operației asociate (*OUTPUT* := *Q*), se va adăuga un nou ciclu aloritmului