

CURS 0x04

CIRCUITE SECVENȚIALE

Circuitele combinaționale sunt eficiente (în general) dar au o problema majoră: sunt one-shot

- **nu putem itera**
- **nu permit niciun fel de “logică internă” sau “memorie internă”**
- **nu există o stare internă a circuitului**
- **sunt prea simple, asociază o funcție logică a intrărilor cu o ieșire (după un anumit timp)**
- **unele lucruri nu pot fi implementate folosind doar logică combinațională**

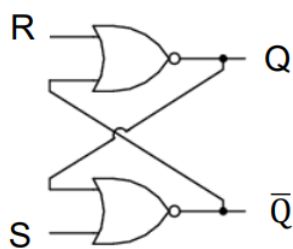
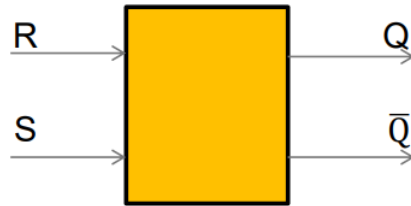
Circuitele secvențiale adresează unele dintre limitările circuitelor combinaționale

Avantajele circuitelor secvențiale:

- **avem stare internă**
- **exemplu: avem un registru în care memorăm count-ul curent**
- **avem variabila de timp**
- **intrările/ieșirile nu sunt fixe**
- **număr variabil de pași în rezolvare**

SR Latch - Set-Reset Latch

memorează un bit de informație



S	R	Q	\bar{Q}
0	0	latch	latch
0	1	0	1
1	0	1	0
1	1	0	0

nu se schimbă nimic

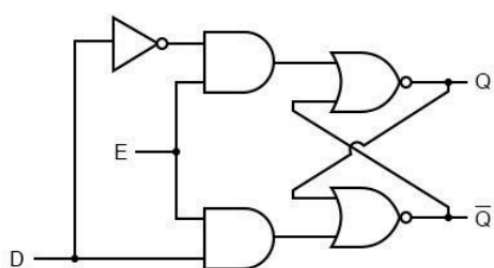
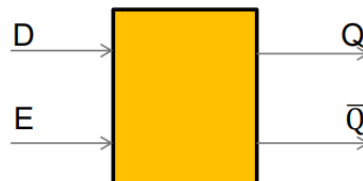
aici punem "0" în memorie

aici punem "1" în memorie

stare invalidă

D Latch

memorează un bit de informație cu un singur input D



E	D	Q	\bar{Q}
0	0	latch	latch
0	1	latch	latch
1	0	0	1
1	1	1	0

nu se schimbă nimic

nu se schimbă nimic

aici punem "0" în memorie

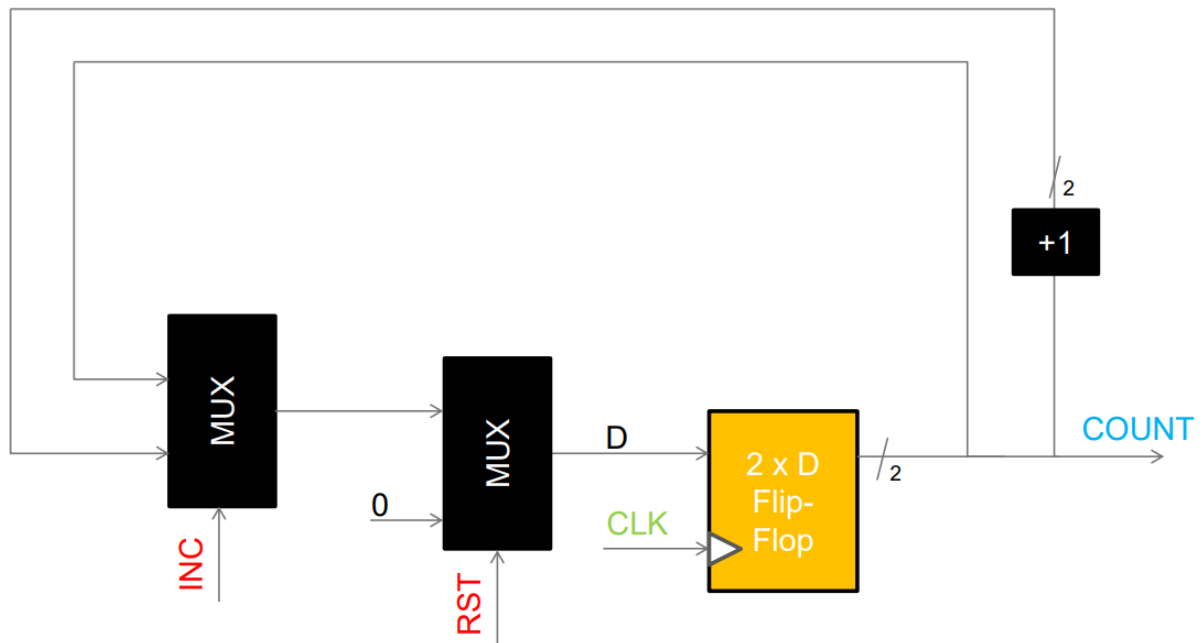
aici punem "1" în memorie

E de la Enable, adică activare
dacă E = 0 nu se întâmplă nimic

D Flip-Flop

pentru sincronizare

exemplu:



este un counter pe 2 biți