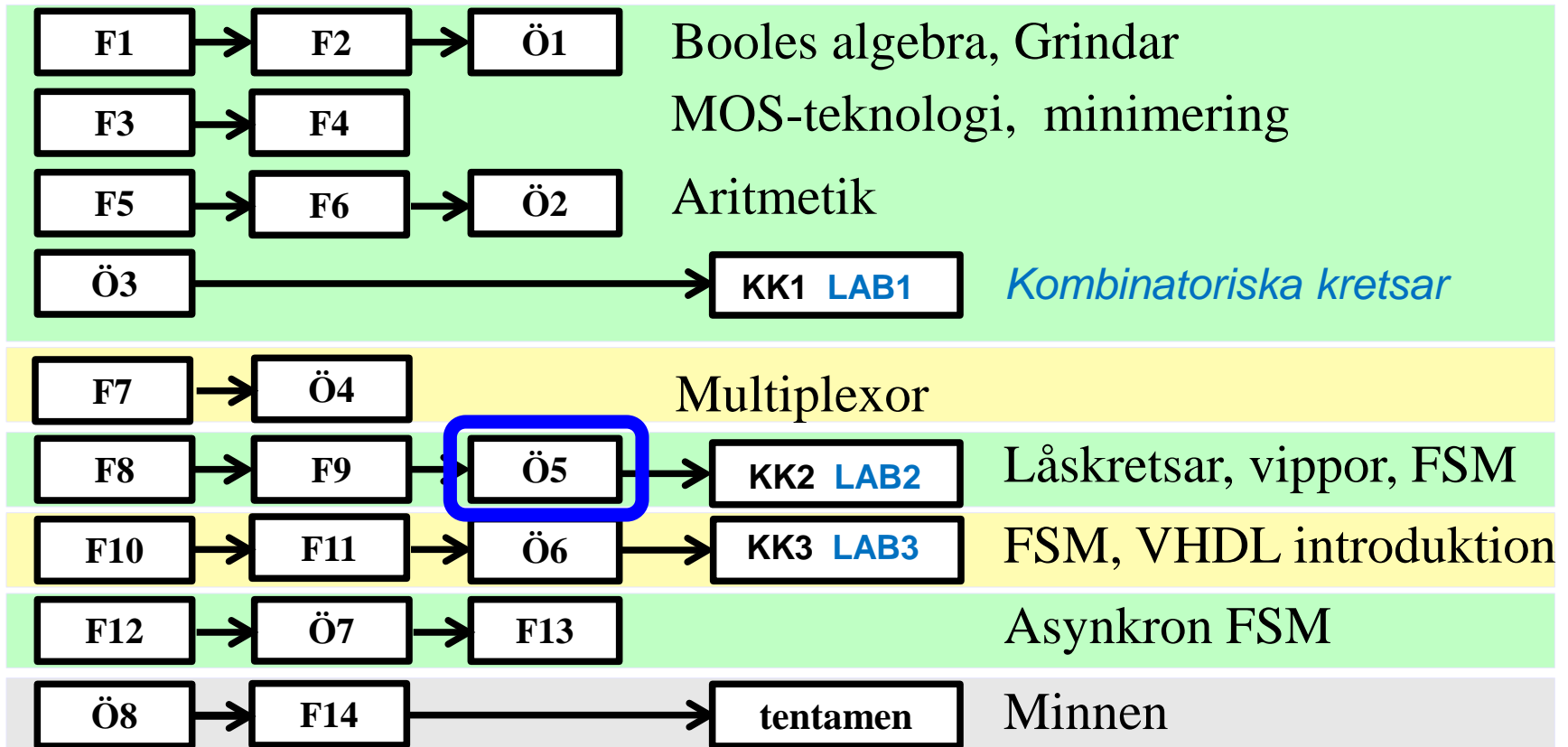


# IE1204 Digital Design

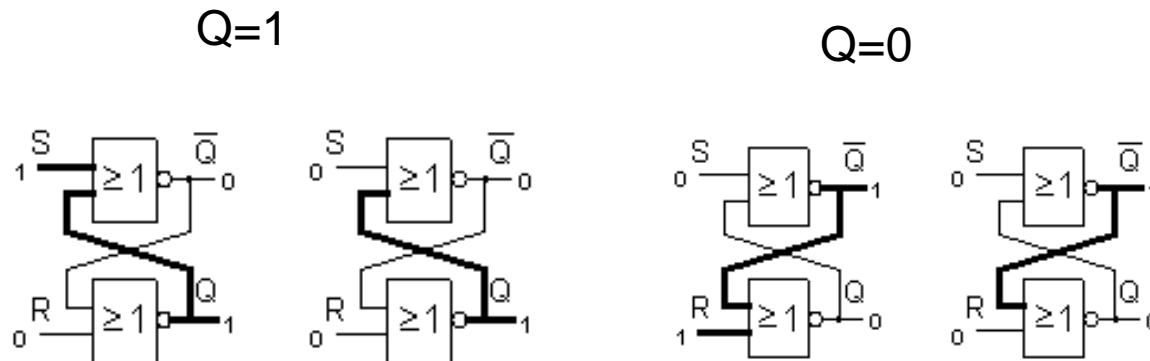


*Föreläsningar och övningar bygger på varandra! Ta alltid igen det Du missat!  
Läs på i förväg – delta i undervisningen – arbeta igenom materialet efteråt!*

# Låskretsar och Vippor

Låskretsar (latch) och vippor (flip-flop) är kretsar med minnesfunktion. De ingår i datorns minnen och i processorns register.

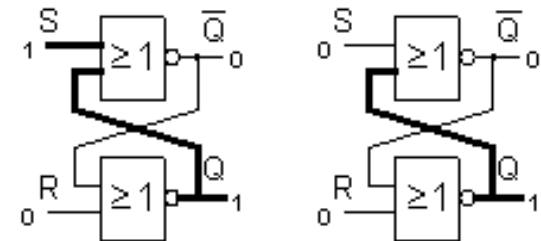
- SR-låskretsen är i princip datorns minnescell



# SR låskrets med två NOR-grindar

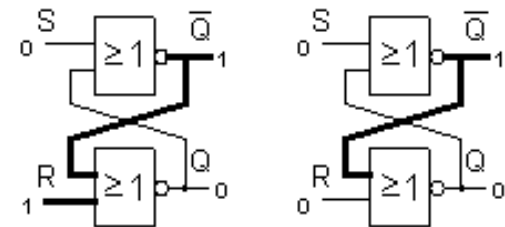
Om  $S=1$  och  $R=0$  låses den övre grinden till "0". Den undre grindens ingångar får då 00 vilket ger utgången  $Q=1$ . Eftersom den övre grinden nu får "1" från två håll, gör det inget om blir  $S=0$  nu.  $Q$  förblir *låst* till att vara "1".

$Q=1$



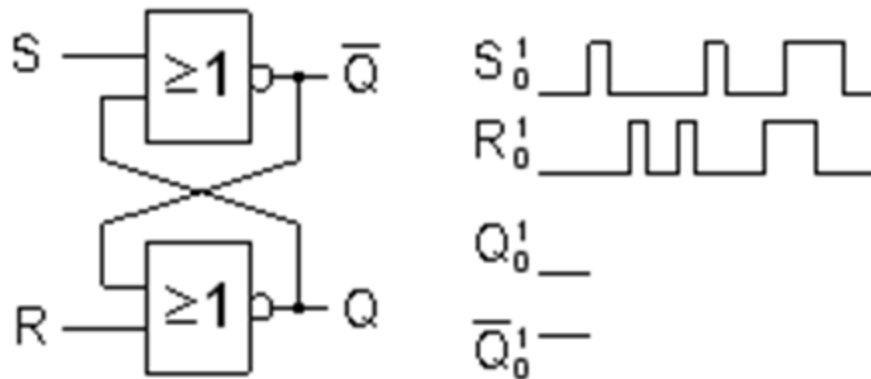
Om  $R=1$  och  $S=0$  låses den nedre grinden till "0". Utgången blir  $Q=0$ . Den övre grindens ingångar får 00 vilket ger "1" på utgången och "1" på ingången till den nedre grinden. Eftersom den nedre grinden nu får "1" från två håll, gör det inget om nu  $R=0$ .  $Q$  fortsätter att vara *upplåst* till "0".

$Q=0$



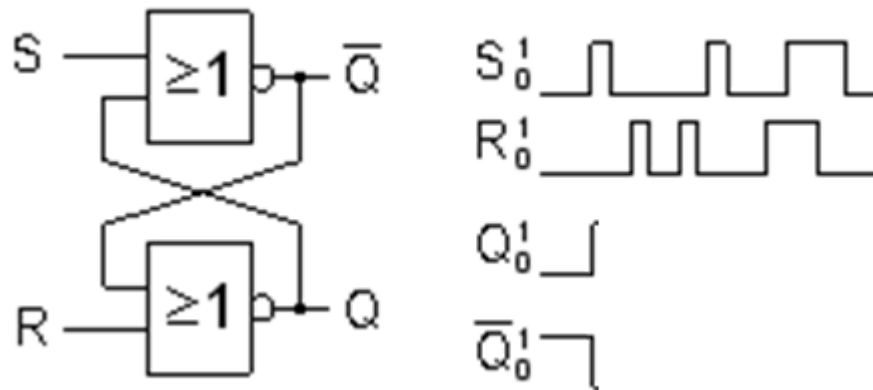
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



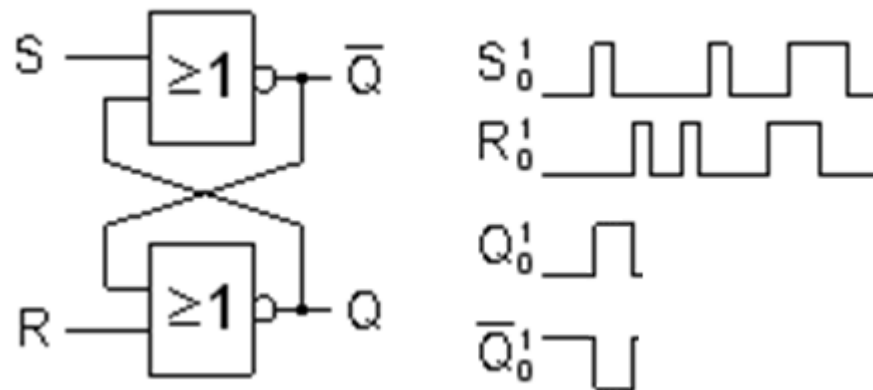
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



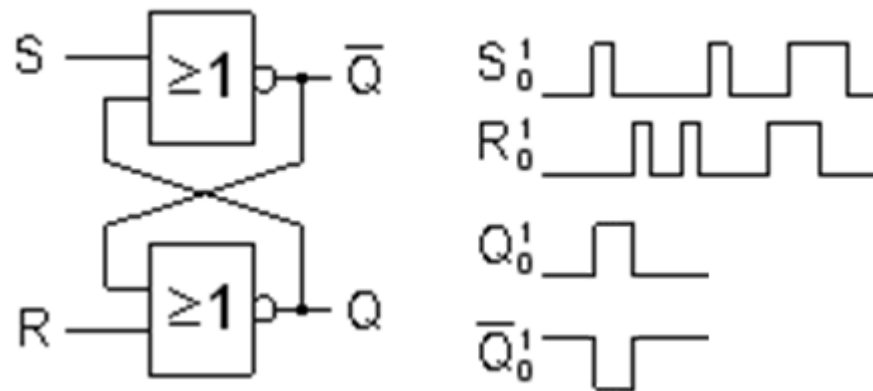
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



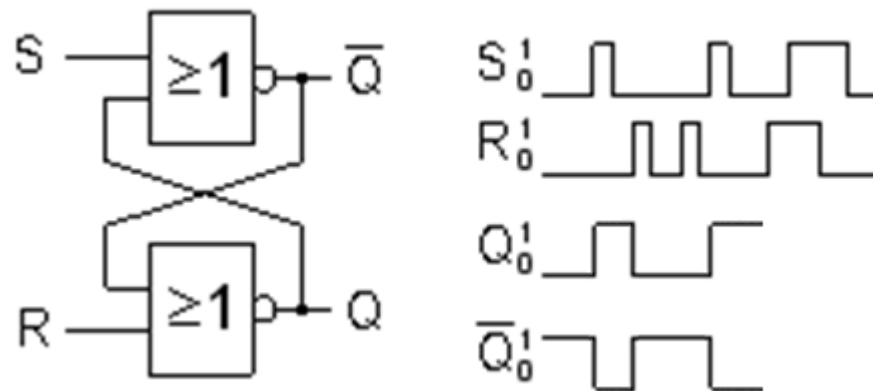
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



# ÖH 9.1

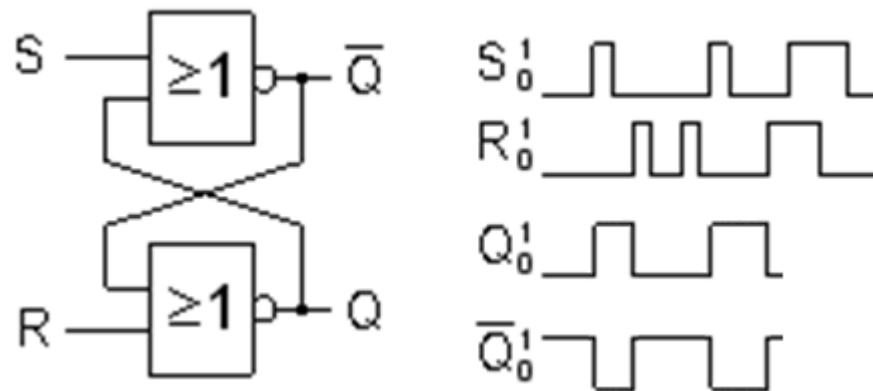
Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)





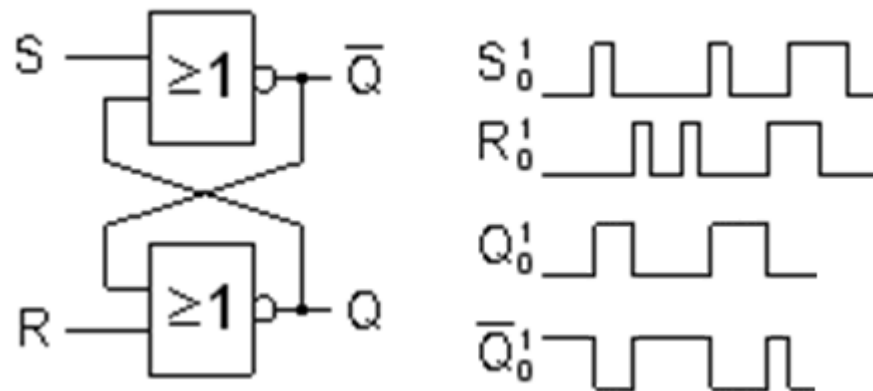
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



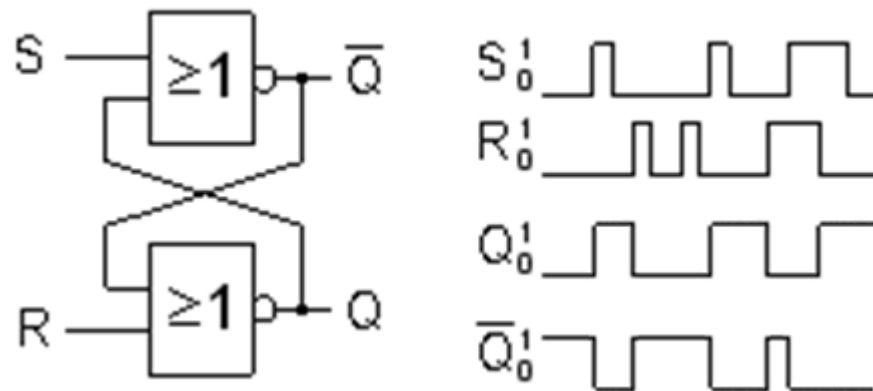
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



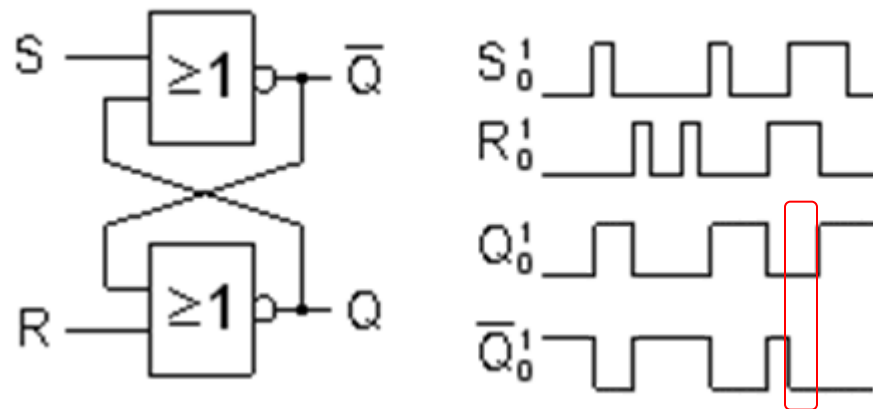
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



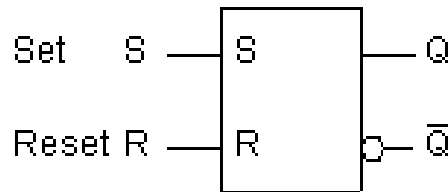
# ÖH 9.1

Komplettera tidsdiagrammet för utsignalerna  $Q$  och  $\bar{Q}$ .  
Avståndet mellan pulserna är mycket längre än grindfördröjningen.  
(Ledning, vad är låsande signal för NOR-grindar)



För  $S=1$  och  $R=1$  fungerar inte låskretsen, utgångarna blir då inte varandras inverser, utan båda 0.

# SR-låskretsens sanningstabell



S	R	Q
0	0	Q, samma
0	1	0
1	0	1
1	1	?, förbjudet

En kort "puls"  $S=1$  "ettställer" låskretsen och en kort "puls"  $R=1$  "återställer" den. Så länge  $S=0$  och  $R=0$  behåller låskretsen sitt värde.

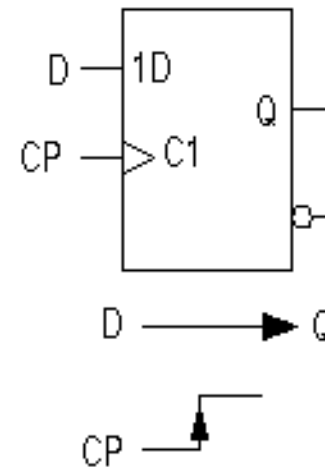
I sanningstabellen anges insignalkombinationen  $S=1$  och  $R=1$  som *förbjuden*!. För den insignalskombinationen skulle båda grindarnas utgångar bli "0" på samma gång. För de övriga insignalkombinationerna gäller att de båda utgångarna är varandras inverser. Om man vill *garantera* att den "andra" utgången alltid är inverterad måste man "förbjuda" en av insignalskombinationerna.

# Klockad vippra

Inom digitaltekniken brukar man skilja på enkla låskretsar ( latch ) och klockade vippra ( flip-flop ).

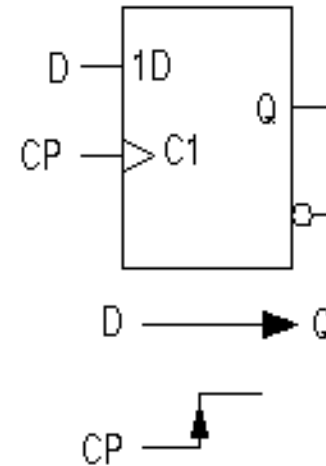
De enkla låskretsarnas begränsning ligger i att man *inte* kan mata in ett nytt värde till ingången *samtidigt* som man läser av utgångsvärdet. De elektroniska kretsarnas snabbhet har gjort det nödvändigt att utveckla mer sofistikerade kretsar.

**Flanktriggad D-vippa.** D-ingången är dataingång, C-ingången är klockpulsingång, därav beteckningen CP. Styringången C har ett flanktriggningstecken, en triangel. När C-ingången nås av en **positiv flank**, det vill säga under den korta tid då C går från "0" till "1", kopieras D-ingångens värde till utgången Q. Utgångsvärdet är sedan låst tills det inkommer en ny flank på klockpulsingången.

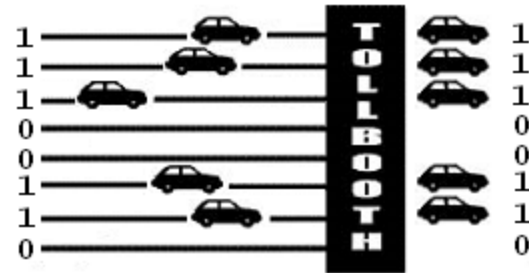


# Synkronisering med D-vippor

- D-vippan används för att *synkronisera* signal-flödet mellan datorns olika delar.

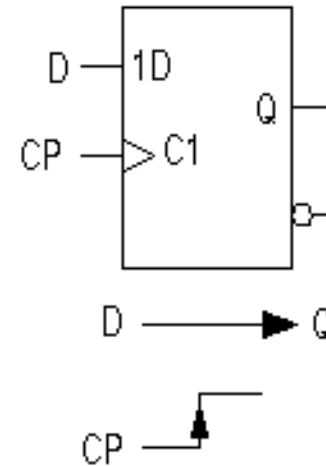


D-vippor används för att bromsa upp *kapplöpningen* mellan signaler tills värdet blivit stabilt.  
(Jämför med tullstationen som stoppar bilarna).

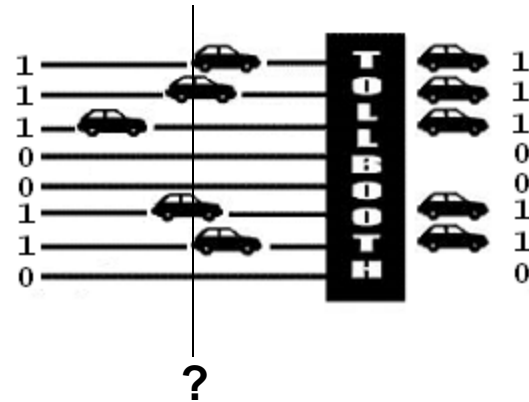


# Synkronisering med D-vippor

- D-vippan används för att *synkronisera* signal-flödet mellan datorns olika delar.



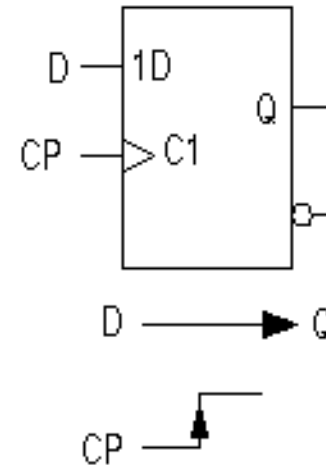
D-vippor används för att bromsa upp *kapplöpningen* mellan signaler tills värdet blivit stabilt.  
(Jämför med tullstationen som stoppar bilarna).



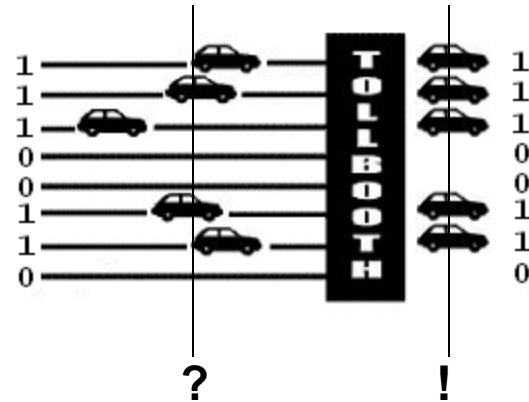


# Synkronisering med D-vippor

- D-vippan används för att *synkronisera* signal-flödet mellan datorns olika delar.

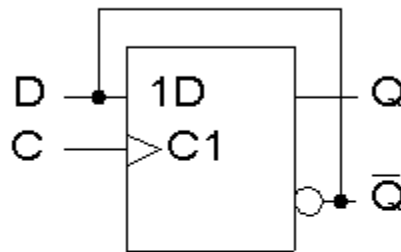


D-vippor används för att bromsa upp *kapplöpningen* mellan signaler tills värdet blivit stabilt.  
(Jämför med tullstationen som stoppar bilarna).



(9.4)

Rita in Q i detta tidsdiagram

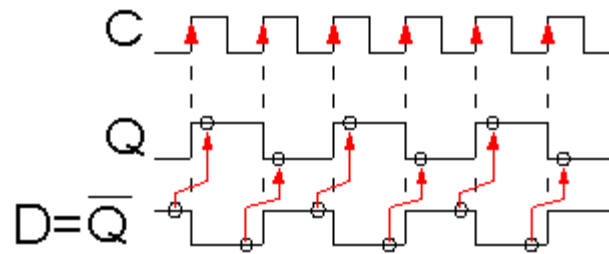
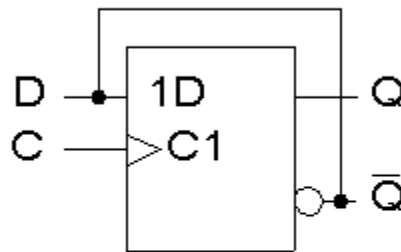


C 

Q 

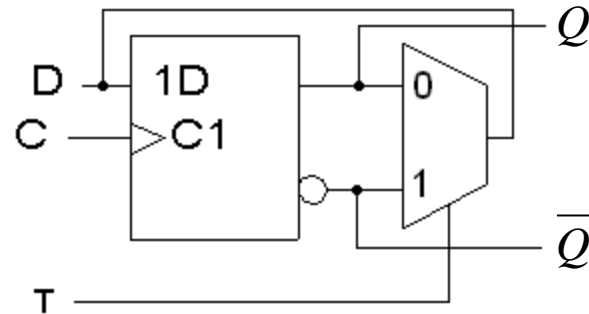
# (9.4)

Rita in Q i detta tidsdiagram

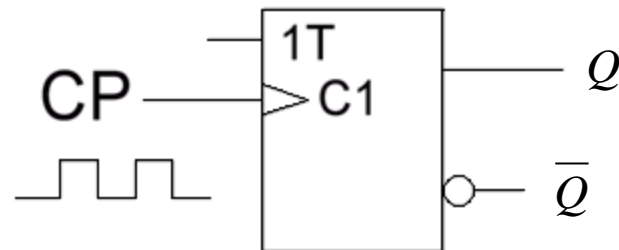


# T-funktionen

T=0 På stället marsch  
T=1 Byt, Toggla

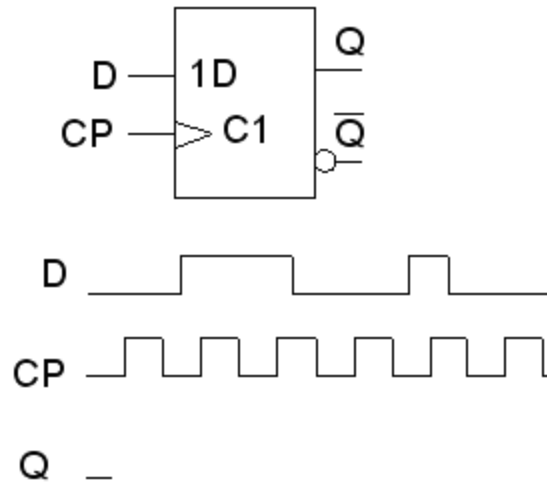
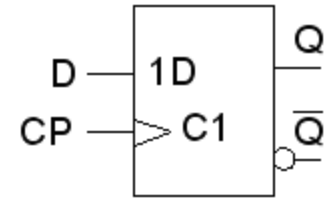


Ibland används denna  
symbol för T-funktionen.  
T-vippa.



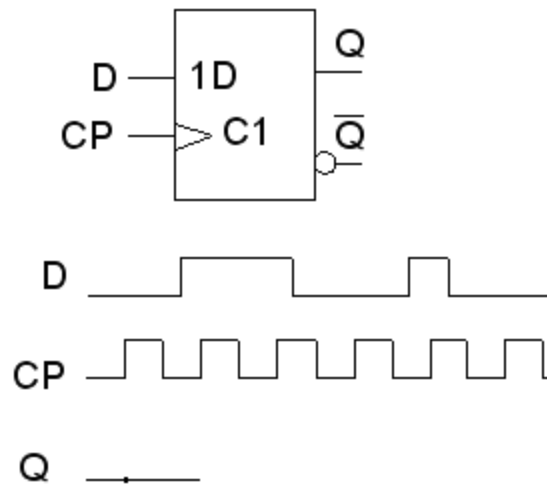
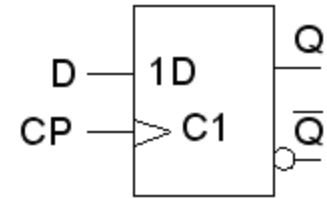
# ÖH 9.3

Rita tidsdiagrammet för utsignalen **Q**, för D-vippan.



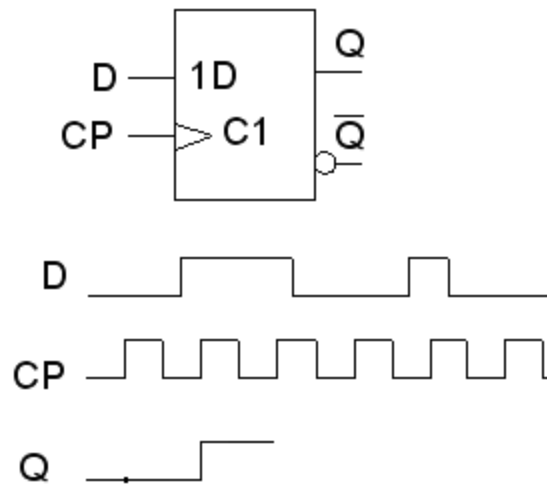
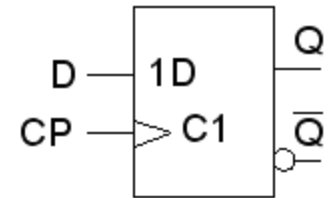
# ÖH 9.3

Rita tidsdiagrammet för utsignalen **Q**, för D-vippan.



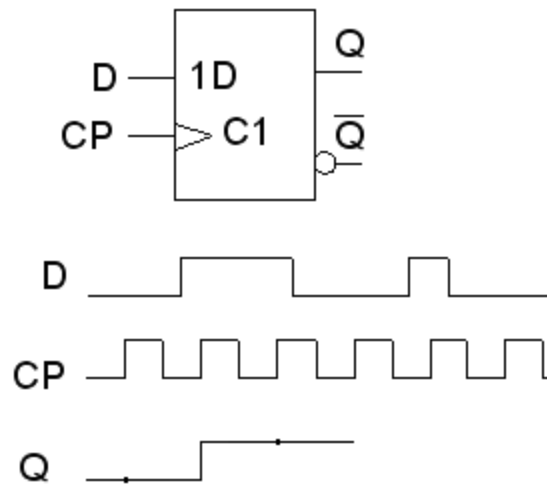
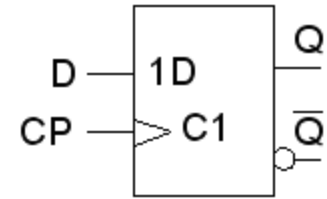
# ÖH 9.3

Rita tidsdiagrammet för utsignalen **Q**, för D-vippan.



# ÖH 9.3

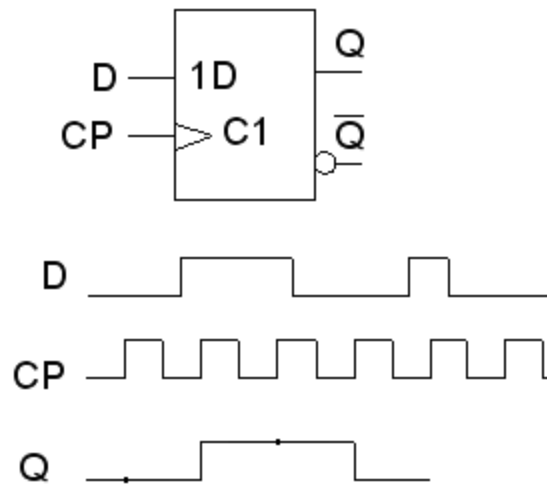
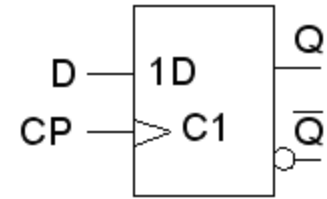
Rita tidsdiagrammet för utsignalen **Q**, för D-vippan.





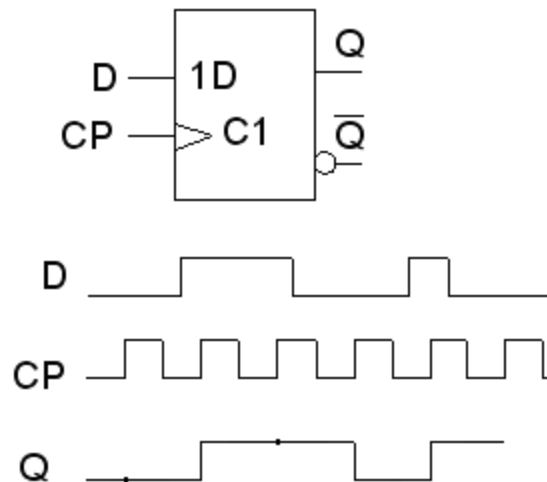
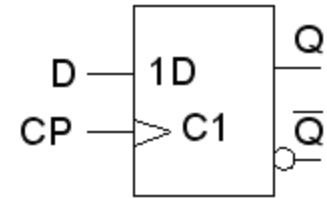
# ÖH 9.3

Rita tidsdiagrammet för utsignalen **Q**, för D-vippan.



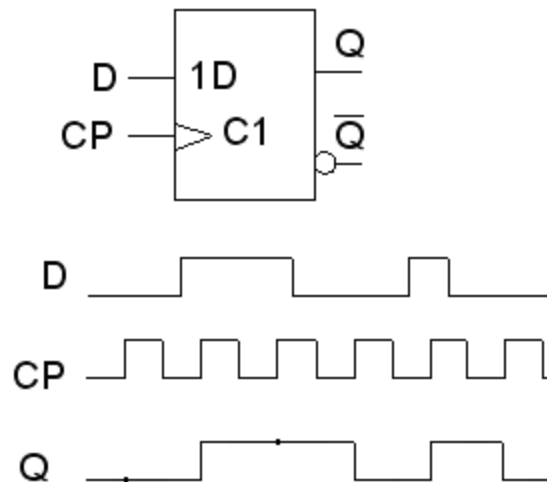
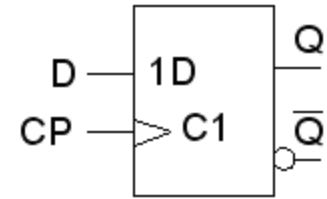
# ÖH 9.3

Rita tidsdiagrammet för utsignalen **Q**, för D-vippan.



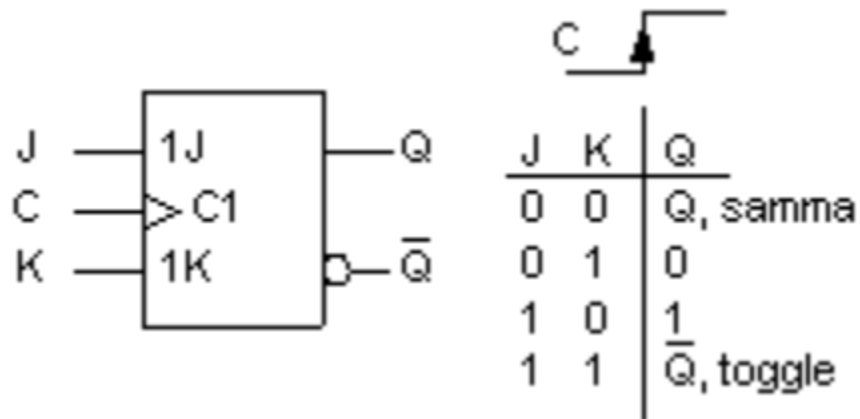
# ÖH 9.3

Rita tidsdiagrammet för utsignalen **Q**, för D-vippan.

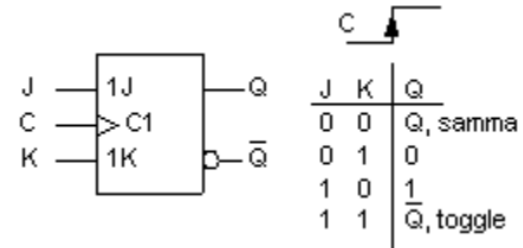


# ÖH 9.5

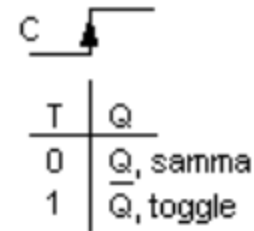
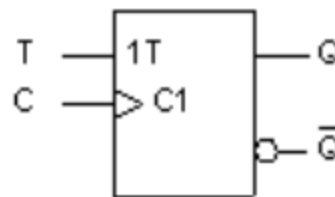
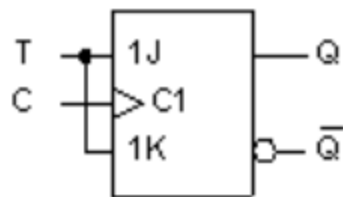
JK-vippan var en äldre typ av "universalvippa". Visa hur den kan användas som T-vippa och som D-vippa.



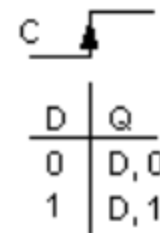
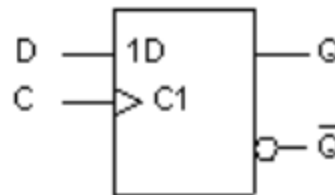
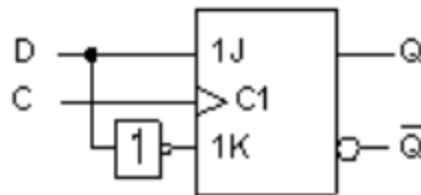
# ÖH 9.5



$T = J = K$



$D = J = \bar{K}$



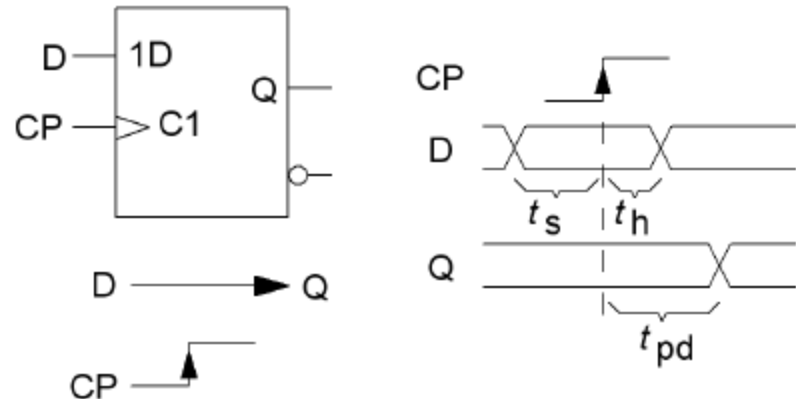
JK-vippan kan användas som T-vippa eller som D-vippa. (När vippor kopplas ihop med varandra finns oftast det inverterade utgångar att tillgå, då behövs inte inverteraren för att göra JK-vippan till D-vippa.)

# ÖH 9.6

## Flip-Flop Timing Parameters.

Vippan laddas med data vid klockpulsens positiva flank, men datat måste vara stabilt tiden  $t_s$  före klockpulsens flank och även tiden  $t_h$  efter. Datat återfinns på utgången efter tiden  $t_{pd}$ . ( $t_{pd}$  kan vara olika för  $0 \rightarrow 1$  respektive  $1 \rightarrow 0$ ).

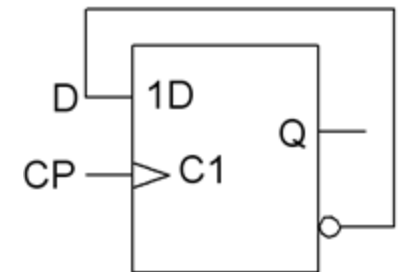
Om dessa tider inte respekteras blir vippans funktion osäker.



Vilken är den högsta klockfrekvens som man kan använda till kretsen i figuren utan att riskera felfunktion?

Antag att

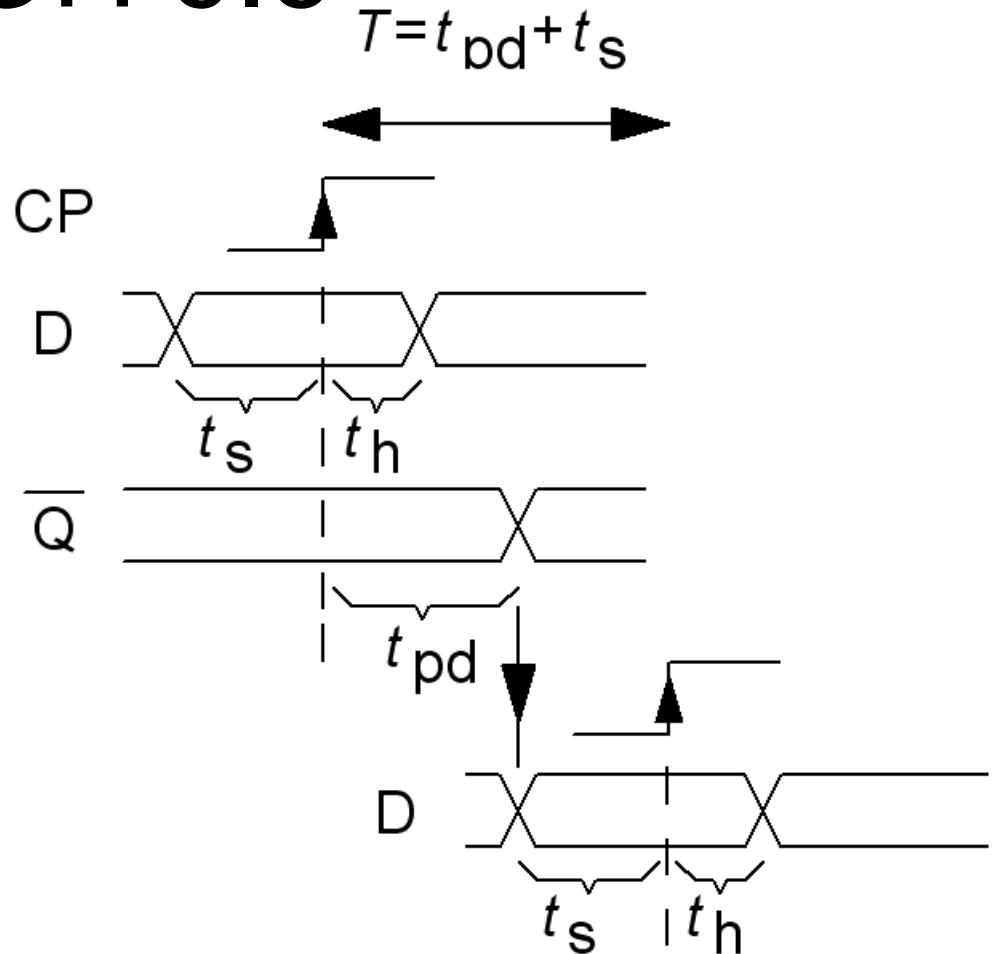
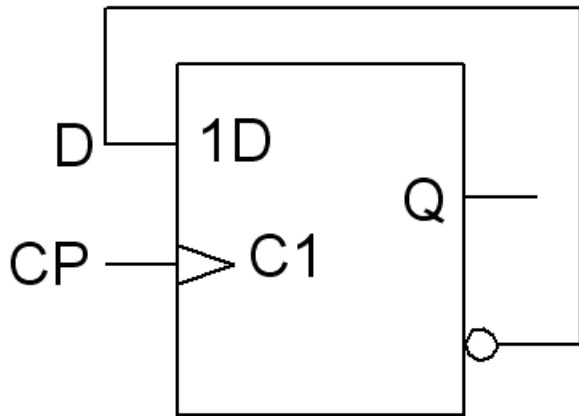
$$t_s = 20 \text{ ns} \quad t_h = 5 \text{ ns} \quad t_{pd} = 30 \text{ ns}$$



- Om utgången kopplas **direkt** till ingången måste  $t_h < t_{pd}$   
Annars - "sågar man av den gren man sitter på"

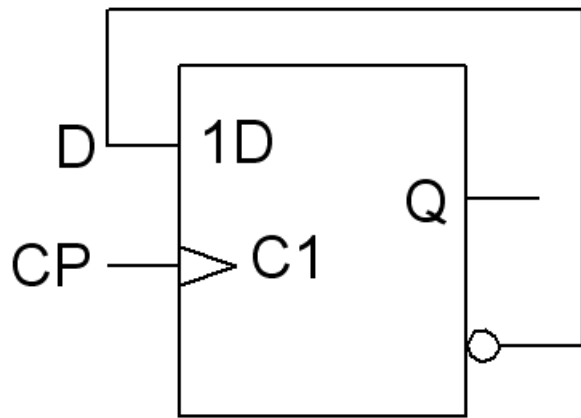
# ÖH 9.6

$t_s = 20 \text{ ns}$     $t_h = 5 \text{ ns}$     $t_{pd} = 30 \text{ ns}$



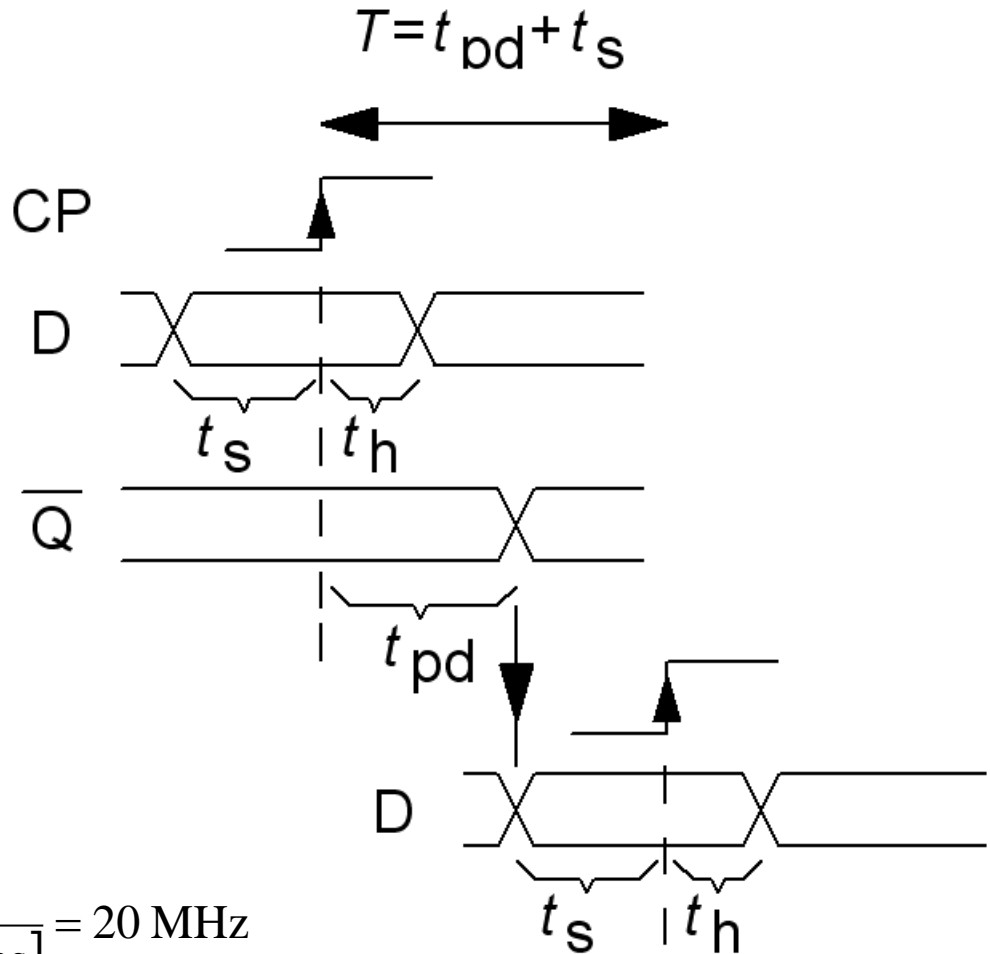
# ÖH 9.6

$$t_s = 20 \text{ ns} \quad t_h = 5 \text{ ns} \quad t_{pd} = 30 \text{ ns}$$



$$T = t_{pd} + t_s$$

$$f = \frac{1}{T} = \frac{1}{t_{pd} + t_s} = \frac{1}{(20 + 30) [\text{ns}]} = 20 \text{ MHz}$$

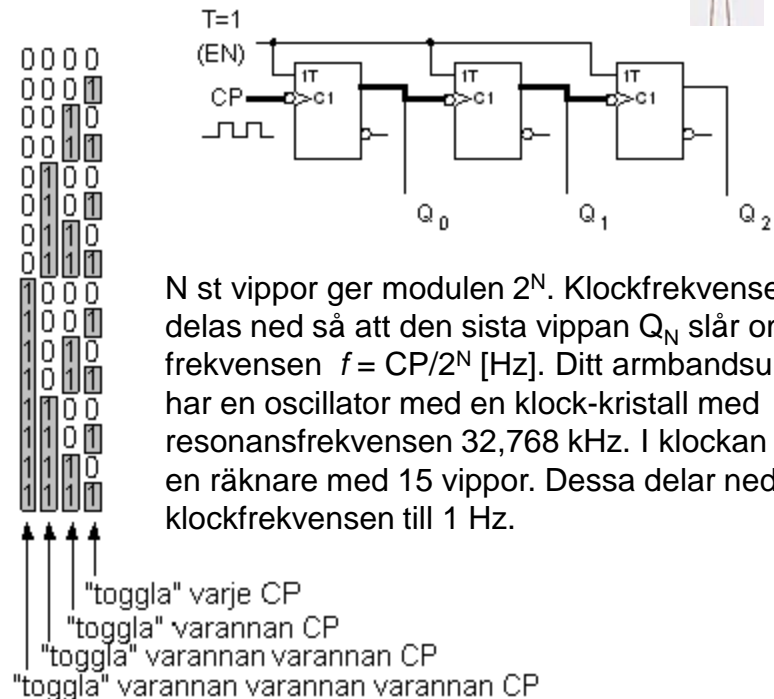




# Asynkron binär-räknare



Figuren ovan visar en binärräknare med tre vippor, räknecykeln har åtta tillstånd så det är en Modulo-8 räknare. Räknaren är uppbyggd av T-vippor, de har alla  $T=1$  och "togglar" därför vid varje klockpuls. Den första vippan  $Q_0$  "togglar" för varje klockpuls. Vippan därefter  $Q_1$  klockas av den första vippan. Den kommer därför bara att "toggla" för varannan klockpuls. Den tredje vippan  $Q_2$  kommer "toggla" för varannan varannan klockpuls.



Enligt binärtabellen kommer räknaren därför att räkna i binärkod.  
(  $Q_2Q_1Q_0$ : 000 001 010 011 100 101 110 111 000 ... ).

# Asynkronräknarens svaghet

Asynkronräknaren har den enklast tänkbara uppbyggnaden. Eftersom klockpulserna tar vägen *genom* vipporna så kan dessa inte slå om exakt samtidigt. Om man läser av den binära koden på vippornas utgångar för att invänta ett visst räknevärde kan man bli "lurad".

Vipporna slår om en efter en och man säger att klockpulsen "ripplar" genom vipporna ( asynkronräknare kallas därför ibland för rippelräknare. Ripple = vågskvalp ). Medan detta pågår kan det kortvarigt förekomma felaktiga räknevärden ( så kallade "räknespikar" ).

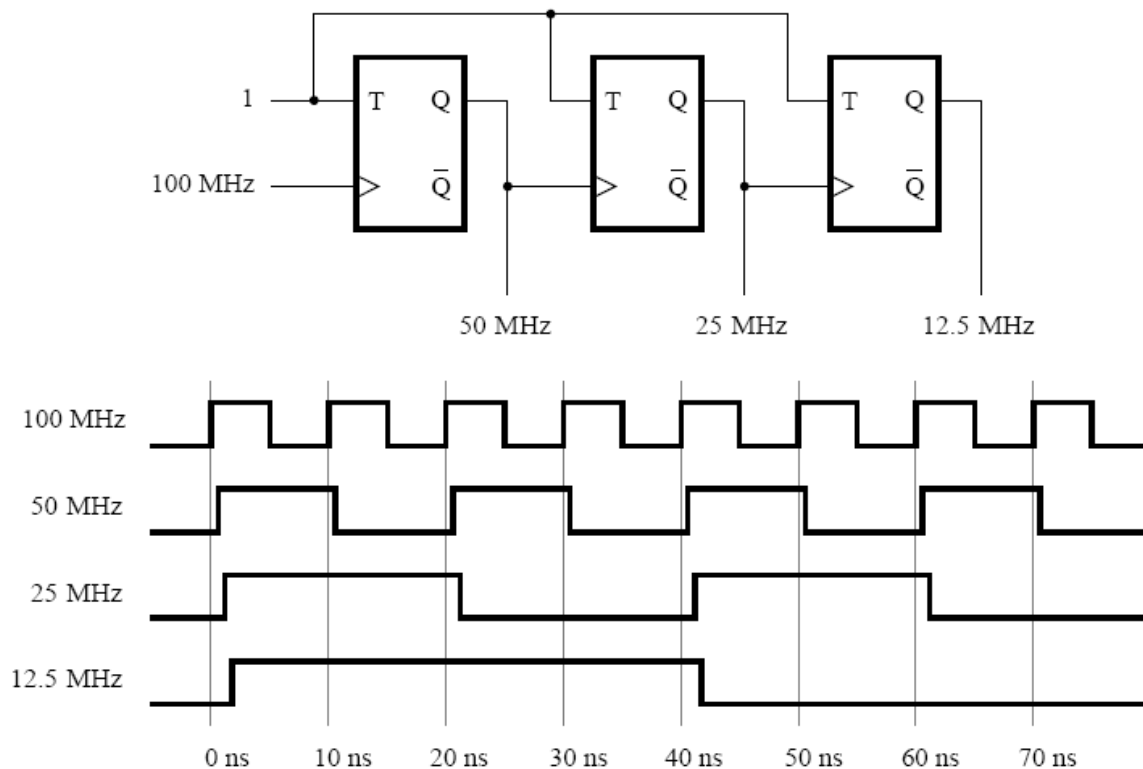
Detta problem har man löst med de **synkrona** räknarna.

# BV 7.5

Given a 100-MHz clock signal, derive a circuit using T flip-flops to generate 50-MHz and 25-MHz clock signals. Draw a timing diagram for all three clock signals, assuming reasonable delays.

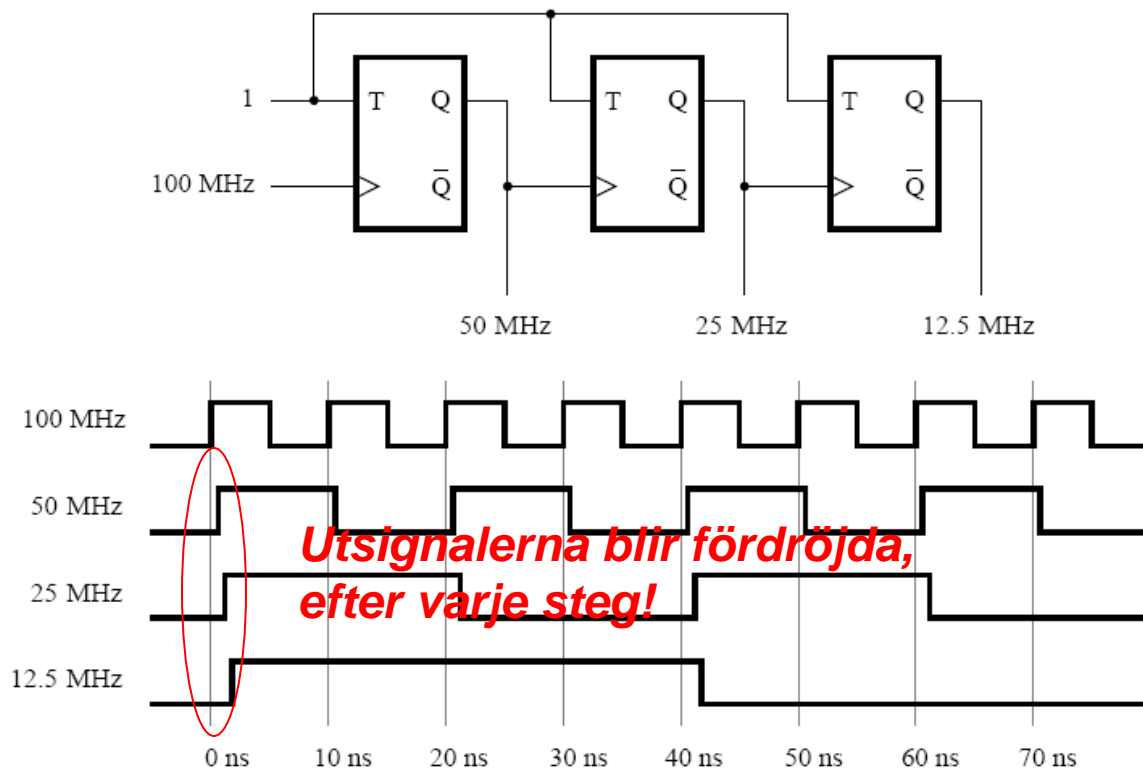
# BV 7.5

Given a 100-MHz clock signal, derive a circuit using T flip-flops to generate 50-MHz and 25-MHz clock signals. Draw a timing diagram for all three clock signals, assuming resonable delays.



# BV 7.5

Given a 100-MHz clock signal, derive a circuit using T flip-flops to generate 50-MHz and 25-MHz clock signals. Draw a timing diagram for all three clock signals, assuming resonable delays.

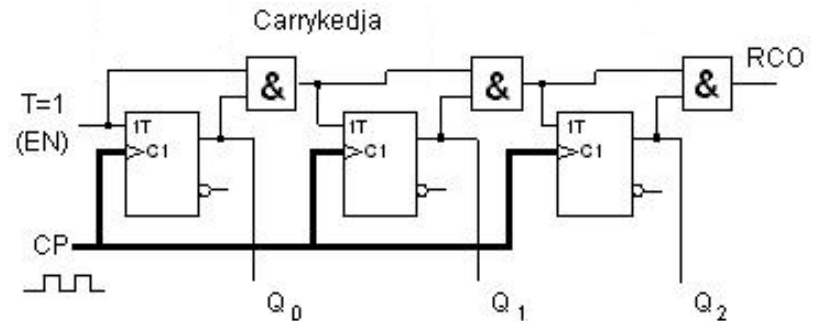


# Synkron binär-räknare

Klockpulserna går direkt till alla vippor och därför slår de om samtidigt. Vilka vippor som ska slå om eller ej styrs med T-ingångarna. Den första vippan har  $T=1$  och den slår om för varje klockpuls. Ur binär-tabellen kan man se att en viss vippa ska slå om när alla vippor som är före den står på "1".

0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

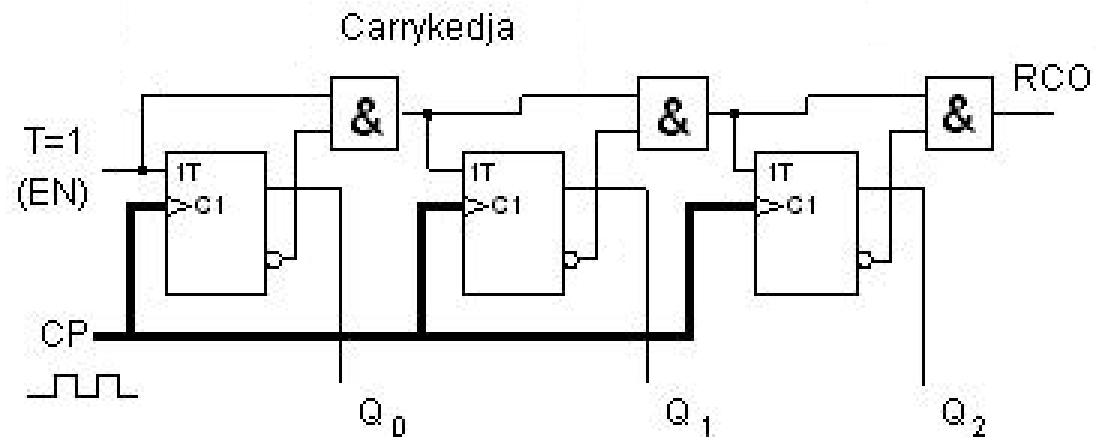
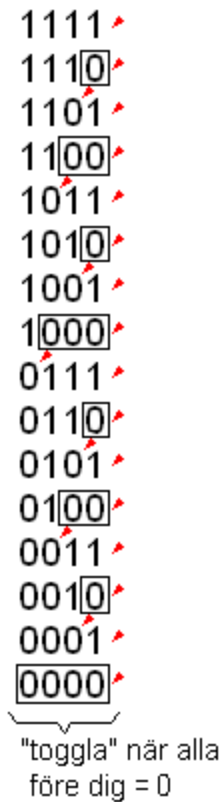
"toggla" när alla före dig = 1



Det villkoret får man från AND-grindarna i den sk. Carrykedjan och det är dessa som styr T-ingångarna. Vill man utöka räknaren sker det med en vippa och en AND-grind per steg.

*Up: Togglar om alla före dig är 1*

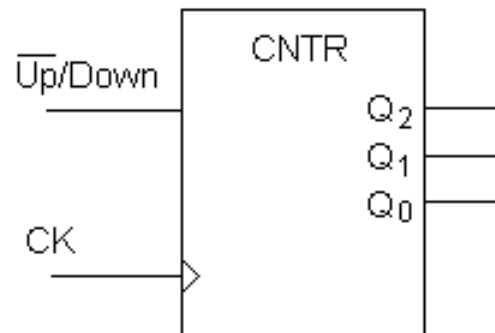
# Synkron ner-räknare



*Down: Togglar om alla före dig är 0*

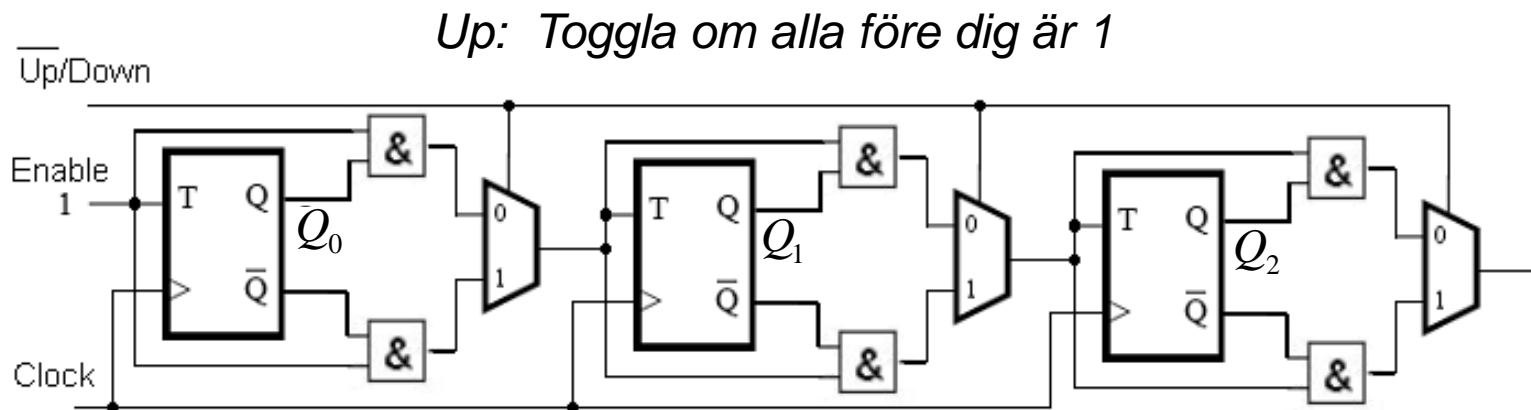
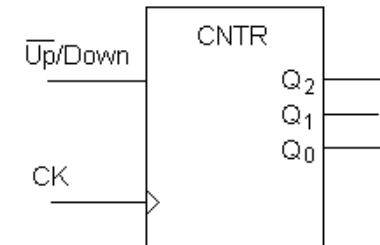
# BV 7.16

Design a three-bit up/down counter using T flip-flops. It should include a control input called UP/Down. If UP/Down = 0, then the circuit should behave as an up-counter. If UP/Down = 1, then the circuit should behave as a down-counter.



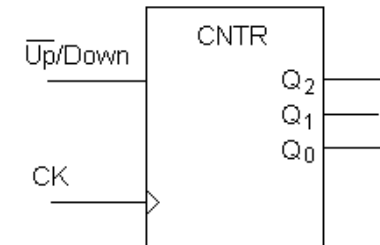


# BV 7.16



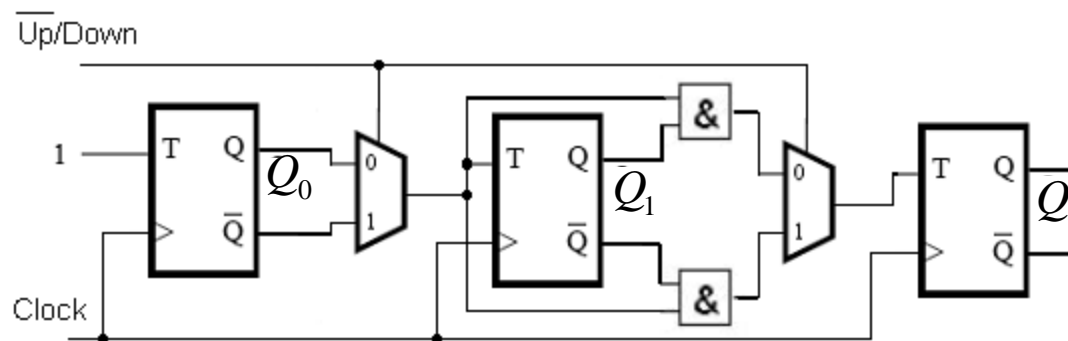
*Down: Togglas om alla före dig är 0*

# BV 7.16



- Det finns inga krav på att räknaren skall kunna *utökas* med fler bitar.
- Det finns inget krav på att räknaren skall kunna *enablas*.

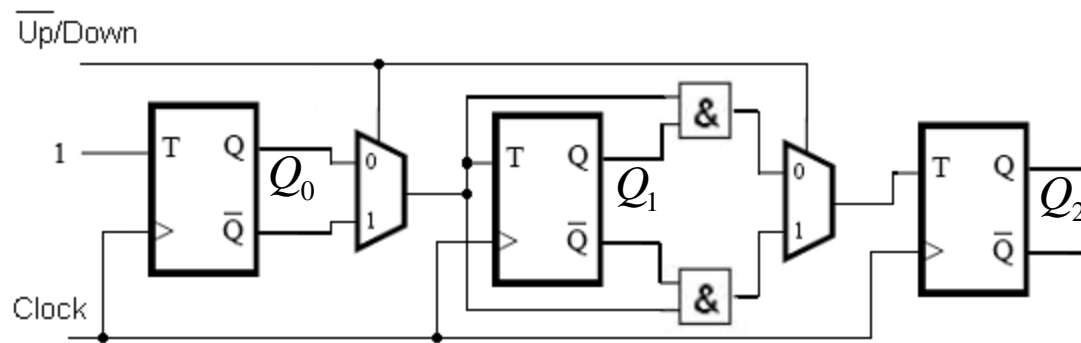
Kretsen kan därför förenklas:



# BV 7.17

Repeat problem 7.16 using D flip-flops.

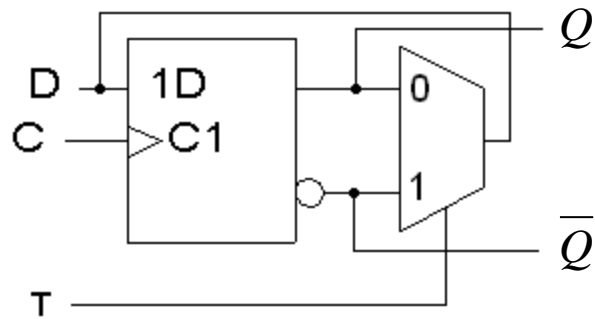
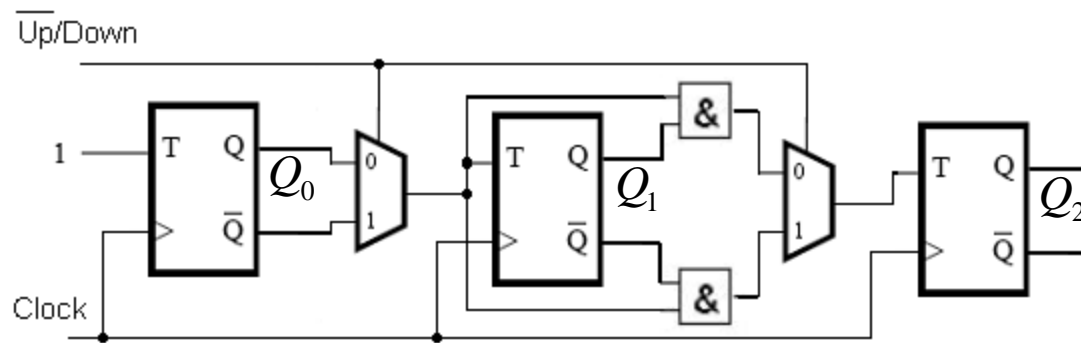
T:



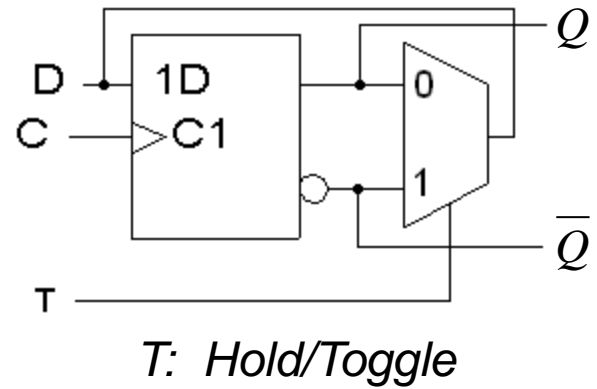
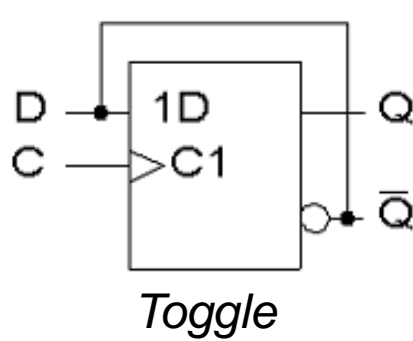
# BV 7.17

Repeat problem 7.16 using D flip-flops.

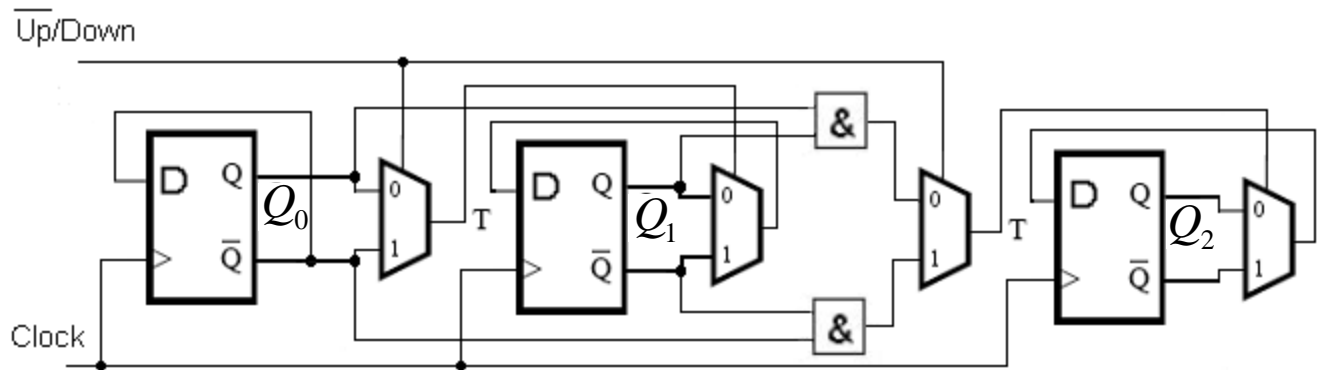
T:

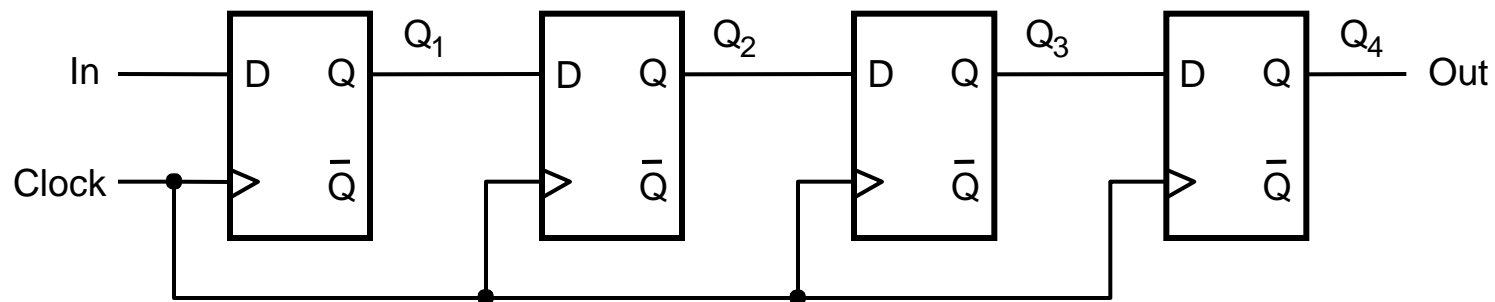


# BV 7.17



D:





(a) Circuit

Ett skift-register  
med en typisk  
sekvens.

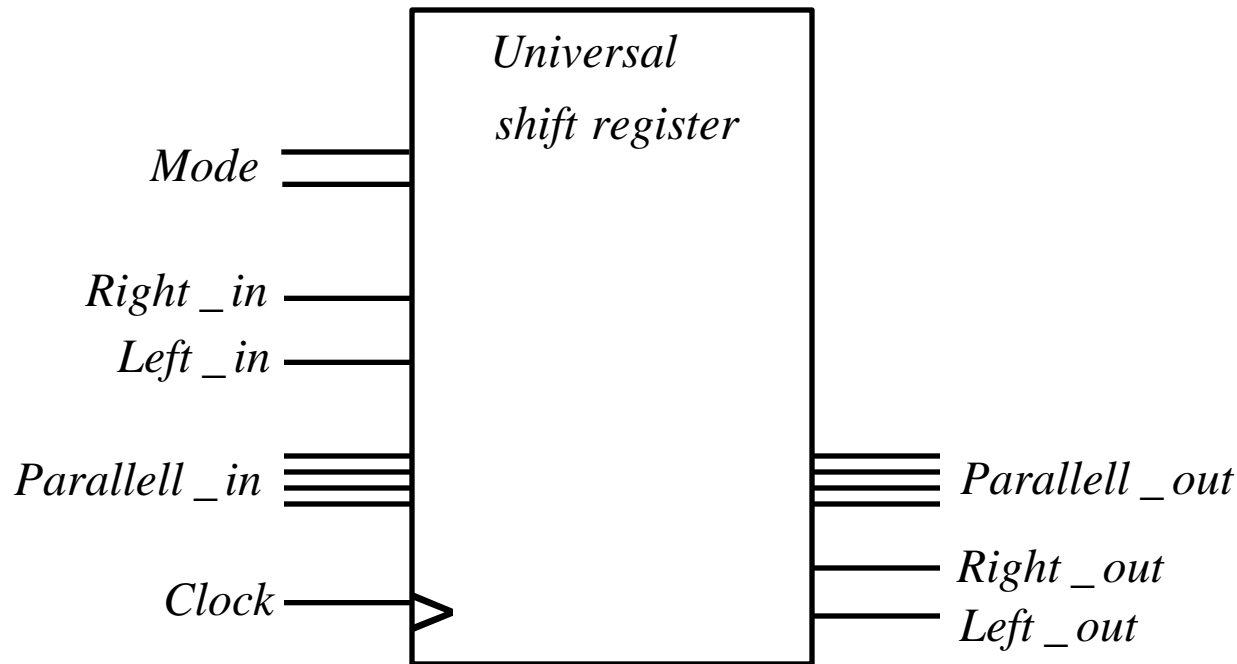
	In	$Q_1$	$Q_2$	$Q_3$	$Q_4 = \text{Out}$
$t_0$	1	0	0	0	0
$t_1$	0	1	0	0	0
$t_2$	1	0	1	0	0
$t_3$	1	1	0	1	0
$t_4$	1	1	1	0	1
$t_5$	0	1	1	1	0
$t_6$	0	0	1	1	1
$t_7$	0	0	0	1	1

(b) A sample sequence

Figure 7.18. A simple shift register.

# BV 7.13

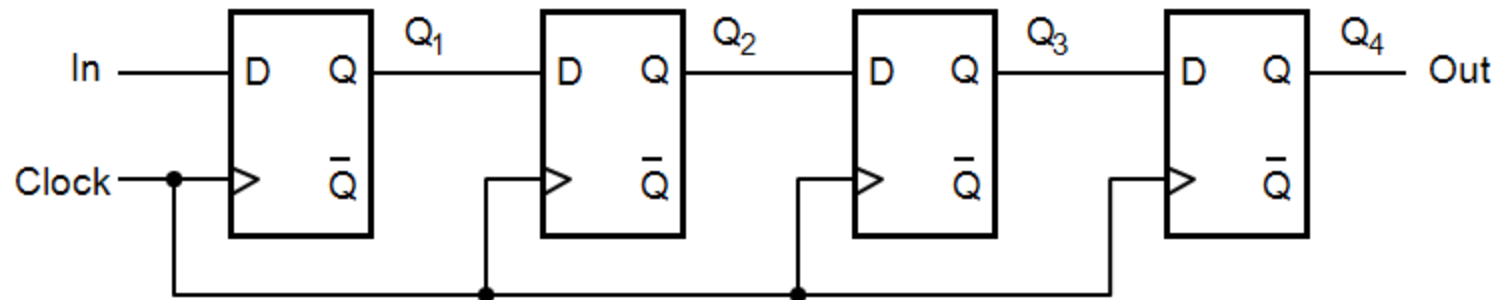
A universal shift register can shift in both the left-to-right and right-to-left directions, and it has parallel-load capability. Draw a circuit for such a shift register.



# BV 7.13

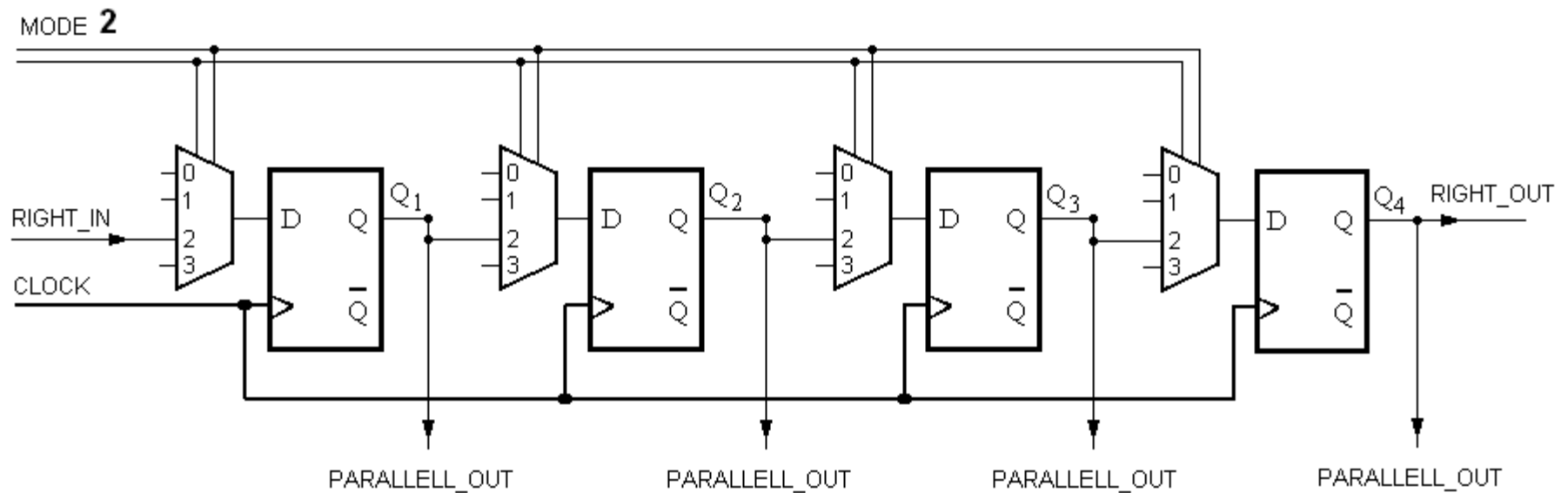
A universal shift register can shift in both the left-to-right and right-to-left directions, and it has parallel-load capability. Draw a circuit for such a shift register.

*Så här ser ett normalt högerskiftande skiftregister ut:*

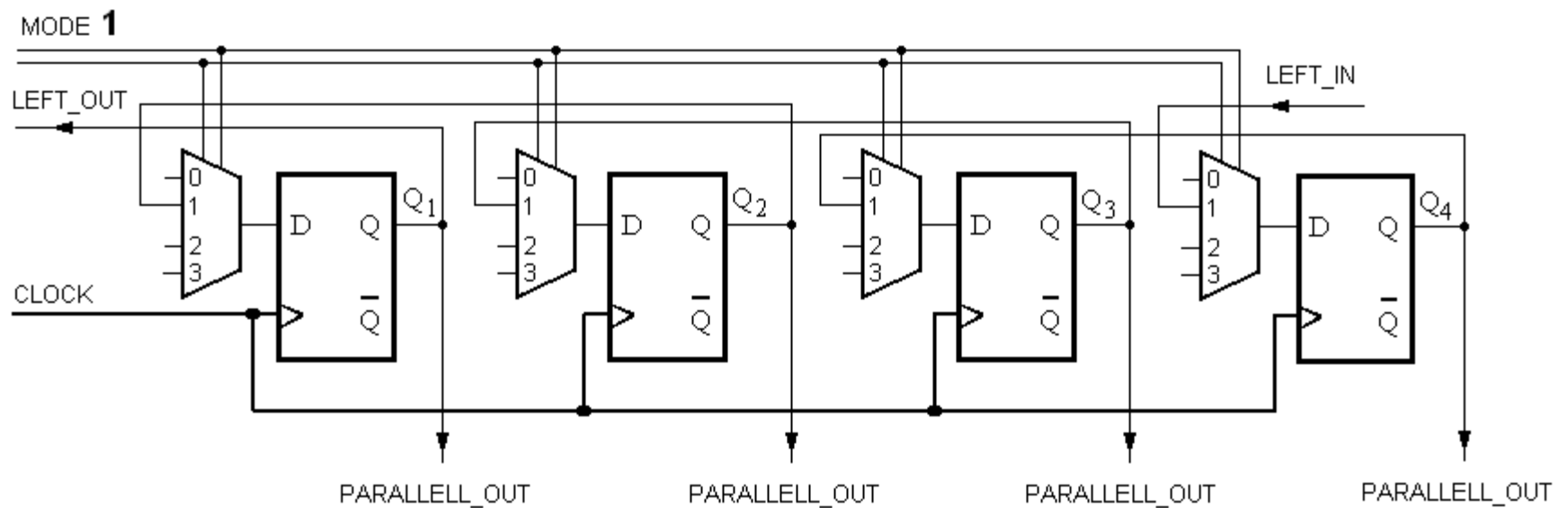




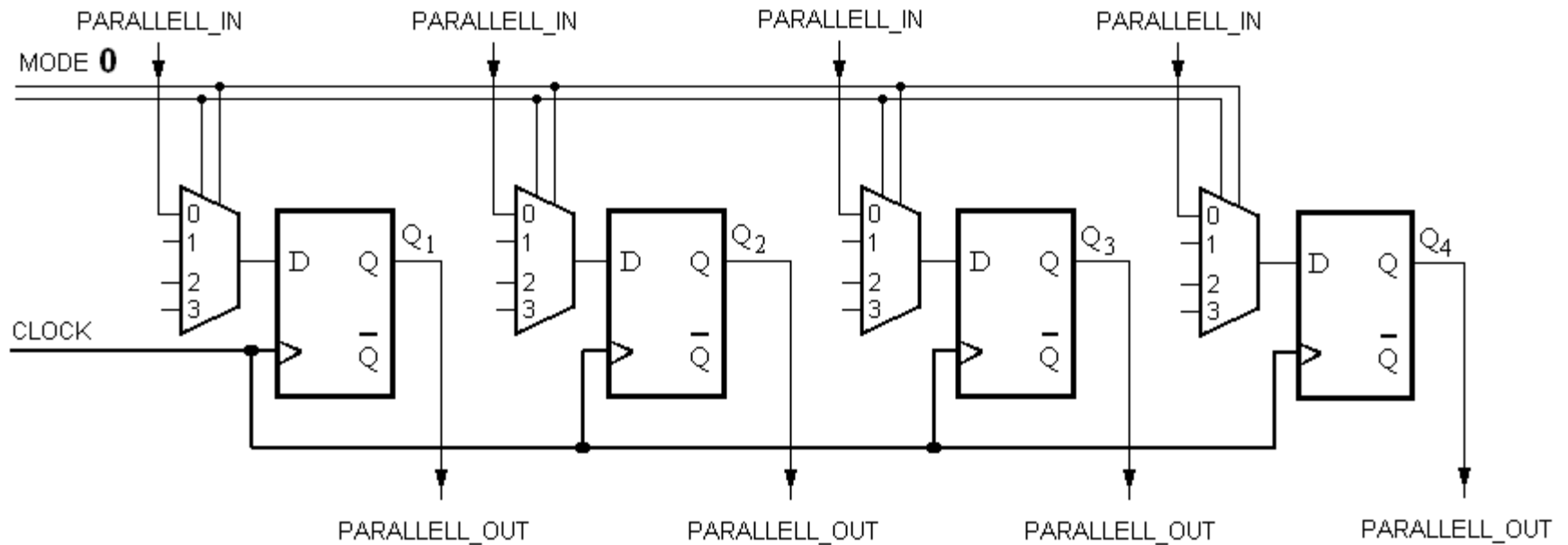
# Mode = 2 Right shift



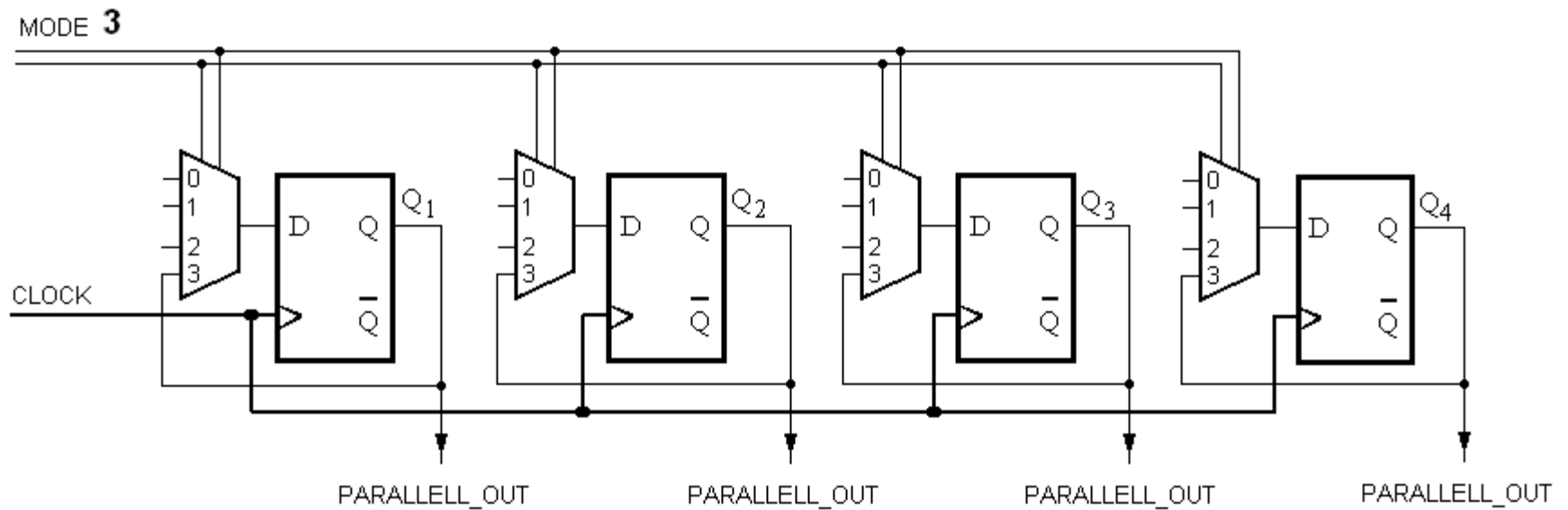
# Mode = 1 Left Shift



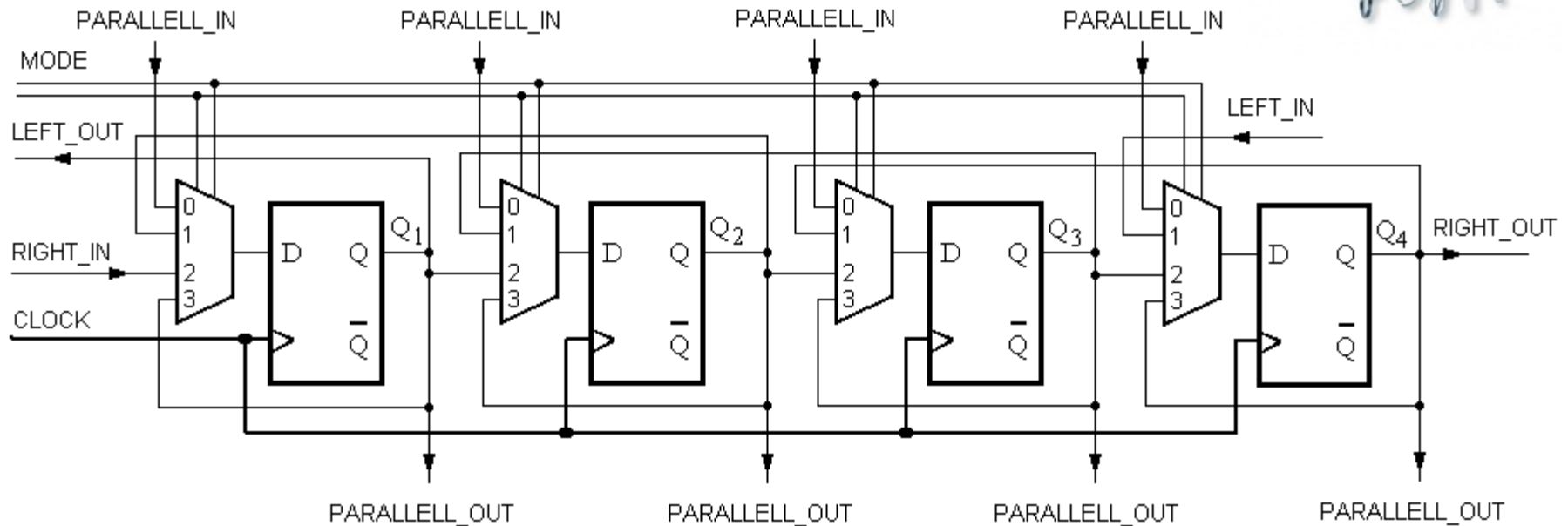
# Mode = 0 Parallel Load



# Mode = 3 Hold



# Multifunction shiftregister



Mode 0 Parallell Load  
Mode 2 Right Shift

Mode 1 Left Shift  
Mode 3 Hold

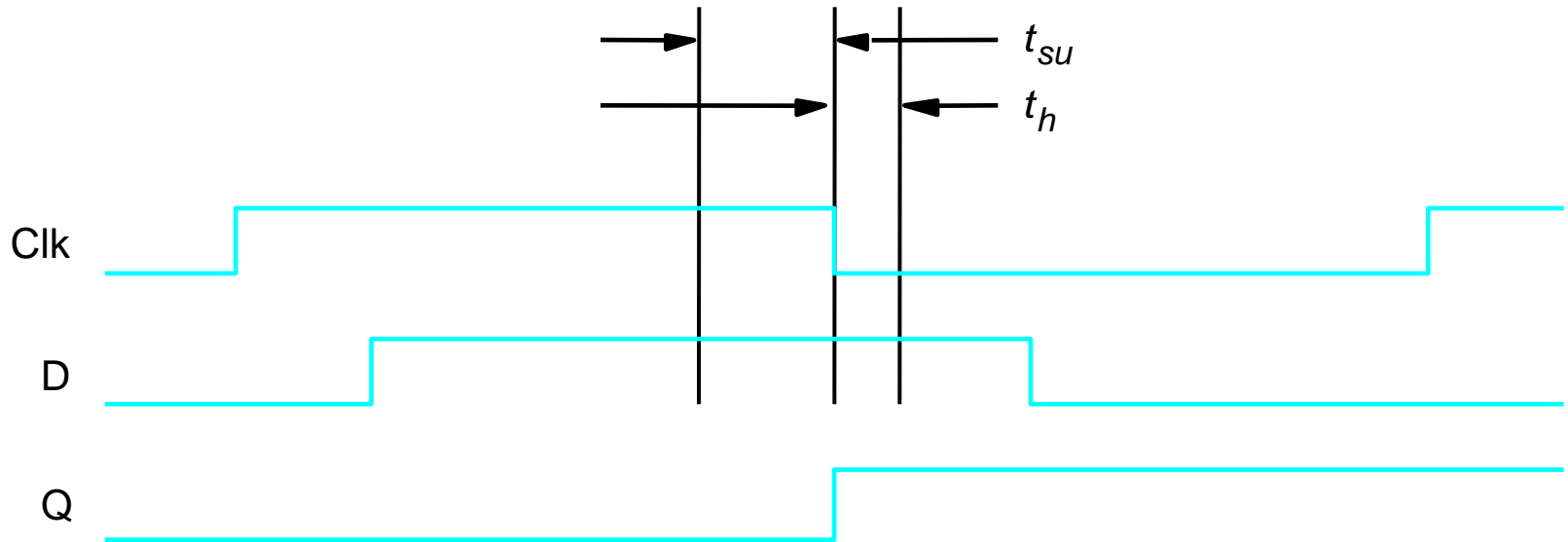


Figure 7.9. Setup and hold times.

# BV 7.24

For the flip-flops in the counter in Figure 7.25, assume that  $t_{su} = 3\text{ ns}$ ,  $t_h = 1\text{ ns}$  and the propagation delay  $t_{pd}$  is 1ns, and the gates and the 2-to-1 multiplexer has the propagation delay  $t_{pd}$  equal to 1ns.

What is the maximum clock frequency for which the circuit will operate correctly?

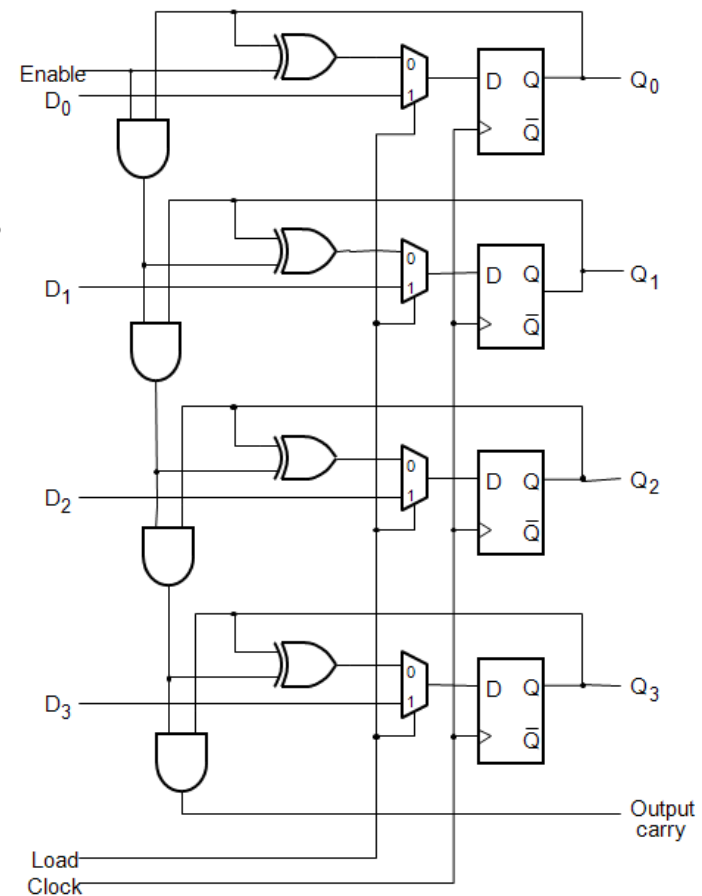


Figure 7.25. A counter with parallel-load capability.

# BV 7.24

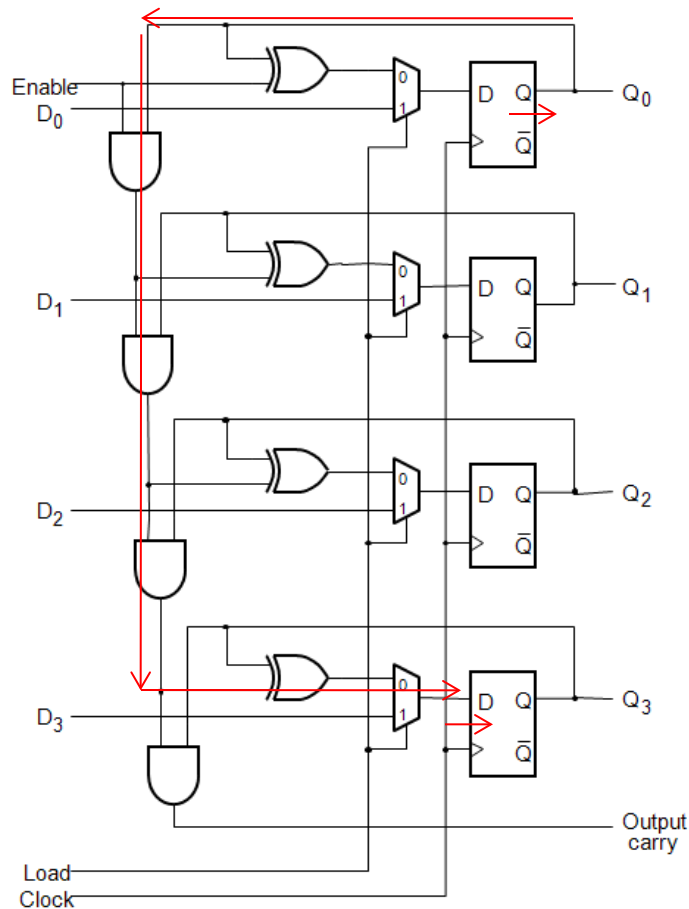


Figure 7.25. A counter with parallel-load capability.

Signalen börjar från vippa 0 efter  $t_{pd}$  och passerar 3 AND-grindar, 1 XOR-grind och 1 MUX på väg till vippa 3.

Vipporna har dessutom krav på att D-signalen skall ha varit stabil  $t_{su}$  före klockpulsen.

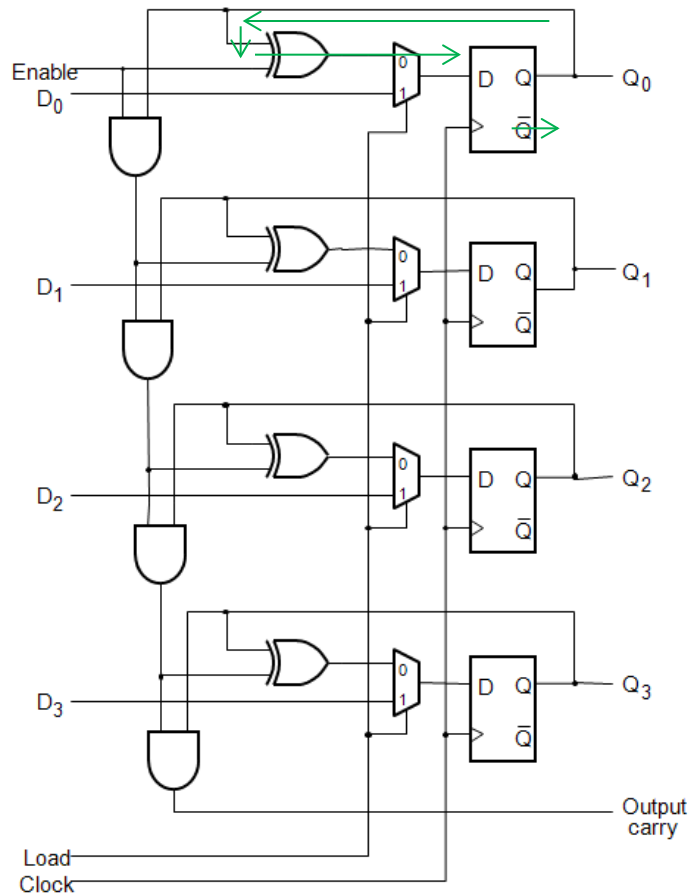
- Check longest path!

$$T_{\min} = t_{pd} + 3 \cdot t_{AND} + t_{XOR} + t_{MUX} + t_{SU} = 1 + 3 + 1 + 1 + 3 = 9 \text{ ns}$$

$$f = \frac{1}{T_{\min}} = \frac{1}{9 \text{ [ns]}} = 111 \text{ MHz}$$



# BV 7.24



D-signalen måste också fortsätta att vara stabil under  $t_h$ .

$$t_h < t_{pd} + t_{XOR} + t_{MUX}$$

*Check!*

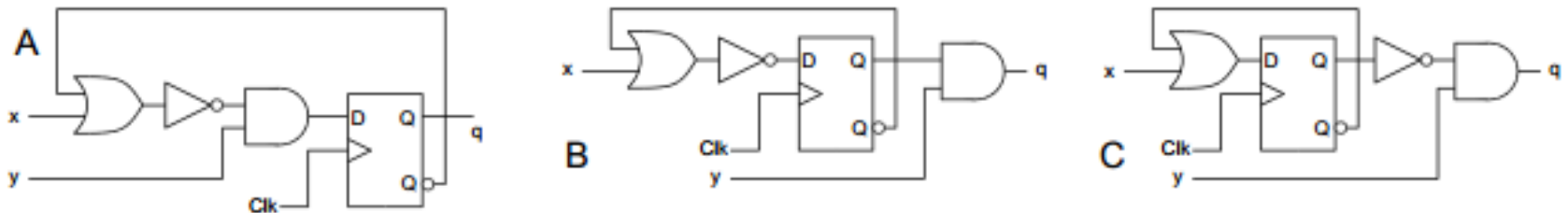
- Check shortest path!

Figure 7.25. A counter with parallel-load capability.

# ( Ex. 9.7 )

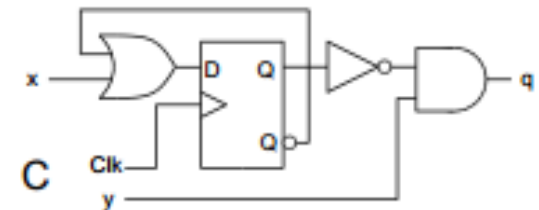
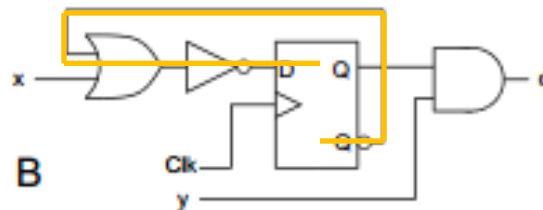
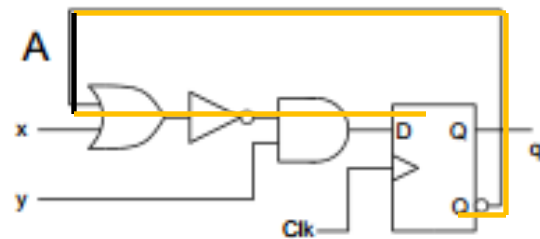
Figuren visar tre olika tillståndsmaskiner. Ange vilken tillståndsmaskin (A, B eller C) som kan operera vid högst klockfrekvens. Markera den kritiska vägen (den väg som begränsar klockfrekvensen) i denna figur samt beräkna periodtiden för klocksignalen Clk.

$$t_{\text{AND}} = 0,4 \text{ ns}, t_{\text{OR}} = 0,4 \text{ ns}, t_{\text{NOT}} = 0,1 \text{ ns}, t_{\text{setup}} = 0,3 \text{ ns}, t_{\text{dq}} = 0,4 \text{ ns}$$

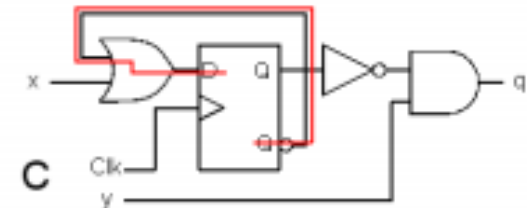


# ( Ex. 9.7 )

$t_{AND} = 0,4 \text{ ns}$ ,  $t_{OR} = 0,4 \text{ ns}$ ,  $t_{NOT} = 0,1 \text{ ns}$ ,  $t_{setup} = 0,3 \text{ ns}$ ,  $t_{dq} = 0,4 \text{ ns}$



$$T = T_{OR} + T_{setup} + T_{dq} = 0,4 + 0,3 + 0,4 = 1,1 \text{ ns}$$



# Heuristiska konstruktionsmetoder?

Denna gång har vi konstruerat **räknare** och **skiftregister** genom att vara "fiffiga".  
(Heuristiska konstruktionsmetoder – erfarenhetsbaserade)

Det är inte alla dagar man är så fiffig, så nästa övning handlar om systematiska konstruktionsmetoder som är generella och därför kan användas på alla sekvenskretsproblem, och alla dagar.

Moore-automaten och Mealy-automaten.

