

# Kapittel 5

## Sekvensiell logikk

M.H

# Hovedpunkter

- Låsekretser (latch'er)
  - SR latch bygget med NOR
  - S'R' latch bygget med NAND
  - D latch
- Flip-Flops
  - Master-slave D flip-flop
  - JK flip-flop
  - T flip-flop

2

M.H

# Definisjoner

## Kombinatorisk logikk

Utgangsverdiene er entydig gitt av nåværende kombinasjon av inngangsverdier

## Sekvensiell logikk

Inneholder hukommelse (låsekretser)

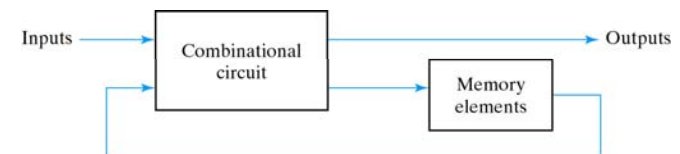
Utgangsverdiene er gitt av nåværende kombinasjon av inngangsverdier, samt sekvensen (tidligere inngangs-/utgangsverdier)

3

M.H

# Sekvensiell logikk

## Sekvensiell krets - generell oversikt



4

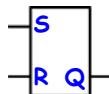
M.H

# Eksempel - låsekrets

Et eksempel basert på et vanlig behov innen digital design:

Ønsker å ha en to-inputs krets med følgende egenskaper:

- 1) Kretsen skal **sette** utgangen **Q** til "1" hvis den får "1" på inngang **S**. Når inngang **S** går tilbake til "0" skal utgangen forbli på "1"
- 2) Kretsen skal **resette** utgangen til "0" hvis den får "1" på inngang **R**. Når inngang **R** går tilbake til "0" skal utgangen forbli på "0"



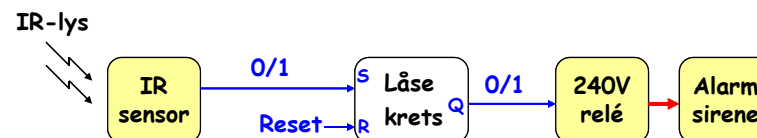
5

M.H

# Praktiske eksempler

Logikk som behandler signaler fra fysiske sensorer:

- Varmefølede persondetektor



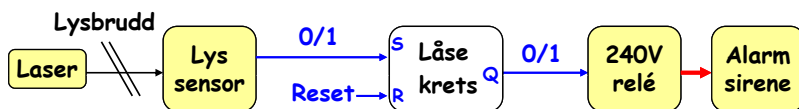
Når IR-lyset varierer mottar logikken et "ras" av kortvarige "1"er pulser ( $\mu$ sek). Logikken skal sette sirenen **permanent på** på første mottatte puls

6

M.H

# Praktiske eksempler

- Laserbasert tyveridetektor



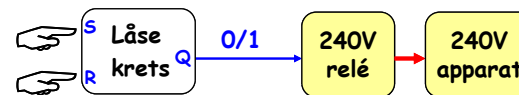
Når laserlyset blir brutt mottar logikken en eller flere "1"er pulser. Logikken skal sette sirenen **permanent på** på første mottatte puls

7

M.H

# Praktiske eksempler

- Touch-bryter



CMOS logikk har meget **høy inngangsmotstand**. Løse innganger reagerer derfor på **statiske spenninger**. Når man berører en løs inngang mottar inngangen et "ras" av **spenningspulser** i mellom ca. -0.6 til +5.5V. Dette kan brukes til å lage en touch-bryter.

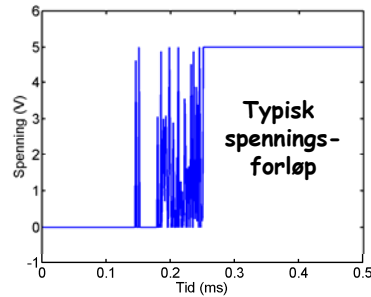
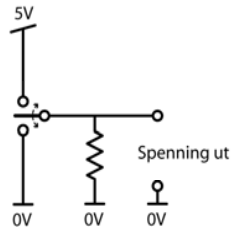
8

M.H

# Praktiske eksempler

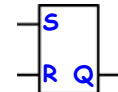
Kontaktprelling fra mekanisk bryter:

Mekaniske brytere gir ikke "rene" logiske nivå ut i overgangsfasen. Slike signal må ofte "renses" ved bruk av låsekretser.



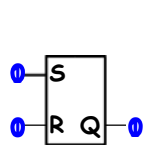
# SR Latch - funksjonell beskrivelse

- 1) Kretsen skal sette Q til "1" hvis den får "1" på inngang S. Når inngang S går tilbake til "0" skal Q forbli på "1"
- 2) Kretsen skal resette Q til "0" når den får "1" på inngang R. Når inngang R går tilbake til "0" skal Q forbli på "0"
- 3) Tilstanden "1" på både S og R brukes normalt ikke



SR		Q
0	0	låst
0	1	0
1	0	1
1	1	0

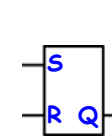
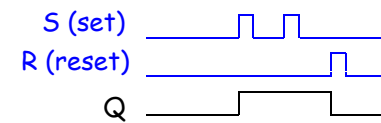
# SR Latch - funksjonell beskrivelse



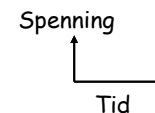
SR		Q
0	0	låst
0	1	0
1	0	1
1	1	0

# SR Latch - funksjonell beskrivelse

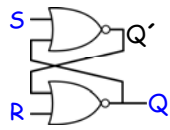
SR latch - en sekvensiell krets



SR		Q
0	0	låst
0	1	0
1	0	1
1	1	0



# SR Latch laget ved NOR

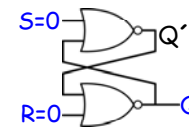


Øvre NOR		Nedre NOR	
S	Q'	Q'	Q
0	1	0	1
0	1	0	1
1	0	1	0
1	0	1	0
1	1	1	0

\*Signalet  $Q'$  er ikke invertert av  $Q$  for tilstand  $S=1, R=1$

# SR - analyse

Tilstand  $S=0, R=0$ : En NOR port med fast "0" inn på en av inngangene er ekvivalent med NOT



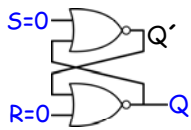
SR		Q
S	R	Q
0	0	1
0	1	0
1	0	0
1	1	0

Øvre NOR		Nedre NOR	
S	Q'	Q'	Q
0	1	0	1
0	1	0	1
1	0	1	0
1	0	1	0
1	1	1	0

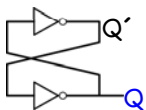
Ser bort i fra tilstand  $S=1$  og  $R=1$

# SR - analyse

Tilstand  $S=0, R=0$ : En NOR port med fast "0" inn på en av inngangene er ekvivalent med NOT



||

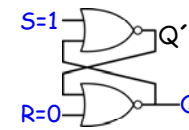


To inverters i ring: stabil låsekrets. Utgangsverdien holdes låst

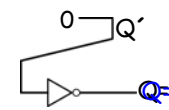
Øvre NOR		Nedre NOR	
S	Q'	Q'	Q
0	1	0	1
0	1	0	1
1	0	1	0
1	0	1	0
1	1	1	0

# SR - analyse

Tilstand  $S=1, R=0$ : En NOR port med fast "1" inn på en av inngangene gir alltid ut "0"



||

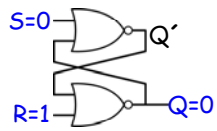


SR		Q
S	R	Q
0	0	låst
0	1	0
1	0	1
1	1	0

Øvre NOR		Nedre NOR	
S	Q'	Q'	Q
0	1	0	1
0	1	0	1
1	0	1	0
1	0	1	0
1	1	1	0

# SR - analyse

Tilstand S=0, R=1: En NOR port med fast "1" inn på en av inngangene gir alltid ut "0"



SR		Q
S	R	Q
0	0	låst
0	1	0
1	0	1
1	1	

Øvre NOR

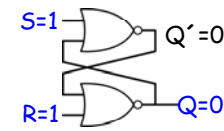
S	Q	Q'
0	0	1
0	1	0
<del>1</del>	<del>0</del>	<del>0</del>
<del>1</del>	<del>1</del>	<del>0</del>

Nedre NOR

Q'	R	Q
<del>0</del>	<del>0</del>	<del>1</del>
0	1	0
<del>1</del>	<del>0</del>	<del>0</del>
1	1	0

# SR - analyse

Tilstand S=1, R=1: En NOR port med fast "1" inn på en av inngangene gir alltid ut "0"



SR		Q
S	R	Q
0	0	låst
0	1	0
1	0	1
1	1	0

Øvre NOR

S	Q	Q'
<del>0</del>	<del>0</del>	<del>1</del>
<del>0</del>	<del>1</del>	<del>0</del>
1	0	0
1	1	0

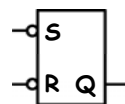
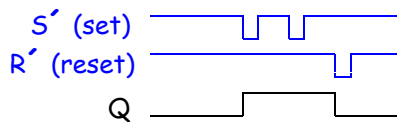
Nedre NOR

Q'	R	Q
<del>0</del>	<del>0</del>	<del>1</del>
<del>0</del>	<del>1</del>	<del>0</del>
1	0	0
1	1	0

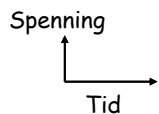
I denne tilstanden er utgang Q' ikke den inverterte av Q. Denne tilstanden brukes normalt ikke

# S' R' Latch

S' R' latch - lik funksjon som SR latch, men reagerer på "0" inn (negativ logikk).



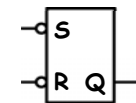
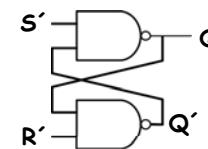
S'	R'	Q
1	1	låst
1	0	0
0	1	1
0	0	1



Tilstanden S' = 0, R' = 0 brukes normalt ikke

# S' R' Latch laget ved NAND

Kretsen kan analyseres på samme måte som for SR latch



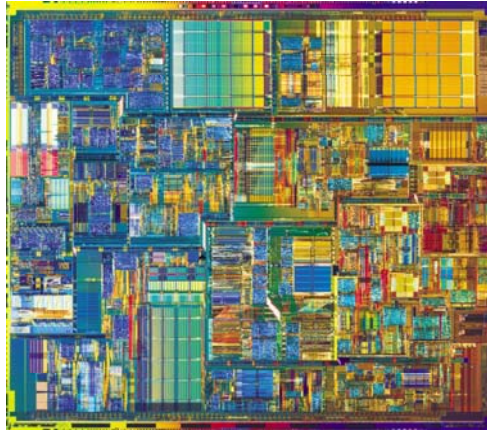
S'	R'	Q
1	1	uforandret
1	0	0
0	1	1
0	0	1

# Synkron logikk

I større digitale system har man behov for å **synkronisere** dataflyten. Til dette bruker vi et globalt **klokkesignal**.

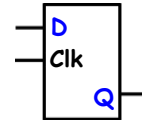
Alle store digitale systemer i dag er synkrone. Eksempel: **Pentium4**

Uten global synkronisering ville vi hatt **totalt kaos**



# D Latch

Dataflyten gjennom en **D latch** kontrolleres av et klokkesignal



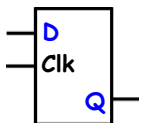
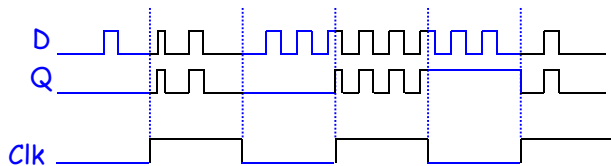
**D latch:**

- 1) Slipper gjennom et digitalt signal så lenge klokkeinngangen er "1" (**transparent**)
- 2) I det øyeblikk klokkeinngangen går fra "1" til "0" **låser** utgangen seg på sin nåværende verdi. Forandringer på inngangen vil ikke påvirke utgangsverdien så lenge klokkesignalet er "0"

# D Latch

**Clk = 1** : kretsen slipper gjennom signalet

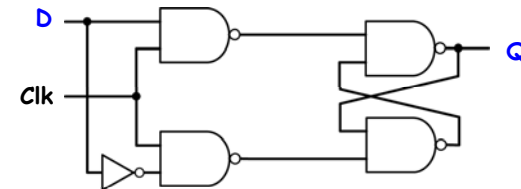
**Clk = 0** : kretsen holder (låser) utgangssignalet



Logisk verdi på **D** i det øyeblikk **Clk** går i fra "1" til "0" bestemmer verdien som holdes på **Q**

# D Latch - implementasjon

Implementasjon ved NAND porter

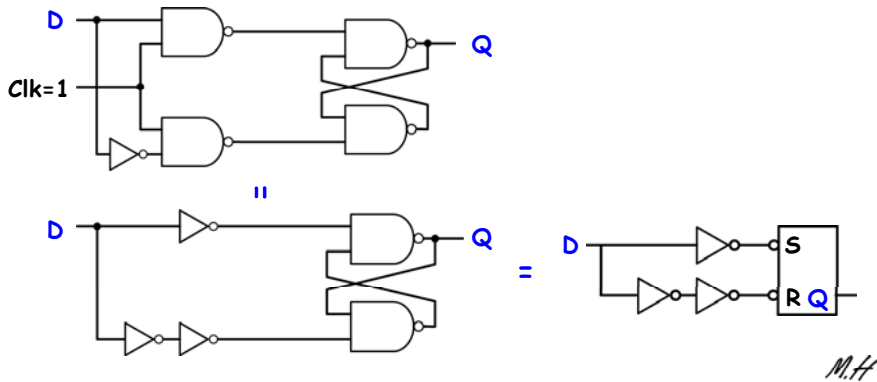


# D Latch - analyse

Analyserer virkemåten for 2 tilstander:

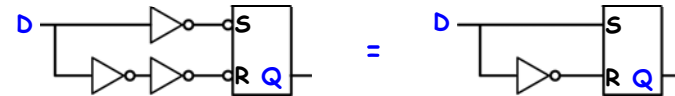
## Tilstand 1: Clk=1

NAND med fast "1" på en inngang er ekvivalent med NOT



# D Latch - analyse

## Tilstand 1: Clk=1



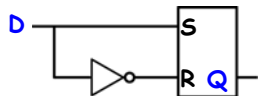
Dette er en  $S'R'$  latch med inverterte innganger  
Tilstand  $S' = R'$  oppstår aldri, og kretsen er ekvivalent med en  $S'R''$  eller SR latch

26

M.H

# D Latch - analyse

## Tilstand 1: Clk=1



SR		Q
<del>0</del>	<del>0</del>	uforandret
0	1	0
1	0	1
<del>1</del>	<del>1</del>	<del>0</del>

S og R vil alltid være forskjellige

Fra sannhetstabellen ser vi at  $Q=S=D$ , kretsen slipper signal D rett gjennom (transparent)

27

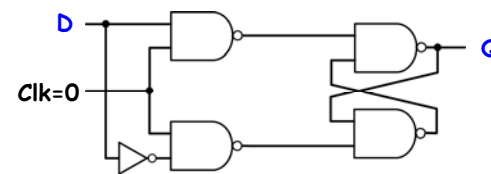
M.H

# D Latch - analyse

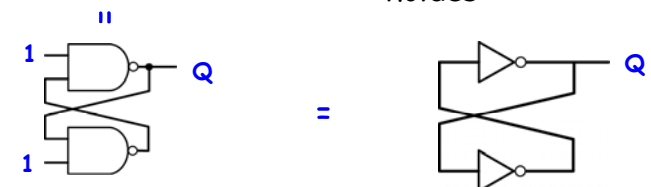
Analyserer virkemåten for siste tilstand:

## Tilstand 2: Clk=0

NAND med "0" på en inngang gir alltid ut "1"



NAND med "1" på en inngang er ekvivalent med NOT. Får stabil låsekrets. Utgang holdes



28

M.H

# Flip-Flop'er

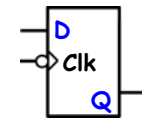
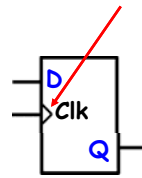
Flip-Flop'er kommer i to varianter:

- Positiv flanke-triggeret
- Negativ flanke-triggeret

På en positiv flanke-triggeret Flip-Flop kan utgangen kun skifte verdi i det øyeblikk klokkesignalet går fra "0" til "1".

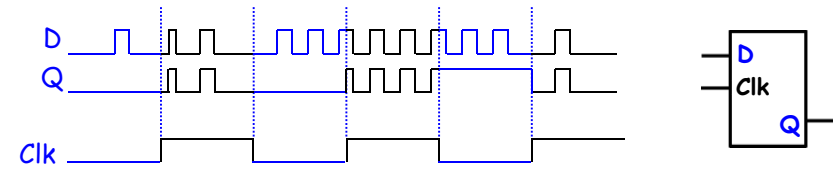
På en negativ flanke-triggeret Flip-Flop kan utgangen kun skifte verdi i det øyeblikk klokkesignalet går fra "1" til "0".

Hakk, indikerer flanke-triggeret

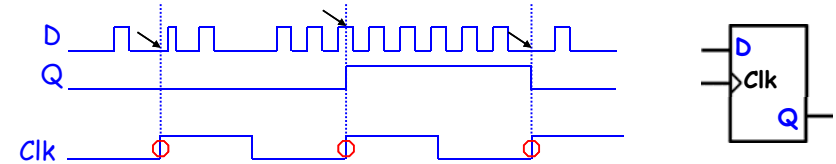


# D Flip-Flop

En D latch er transparent for Clk=1

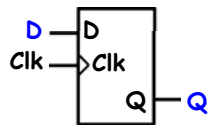


En positiv flanke-triggeret D flip-flop sampler verdien på D i det øyeblikk Clk går fra "0" til "1" (positiv flanke). Denne verdien holdes fast på utgangen helt til neste positive flanke



# Karakteristisk tabell/ligning

For flip-flop'er kan man generelt beskrive neste utgangsverdi  $Q(t+1)$  som funksjon av nåværende inngangsverdi(er), og nåværende utgangsverdi  $Q(t)$



Karakteristisk tabell for D flip-flop

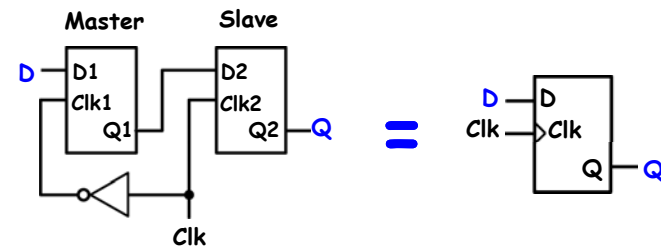
D	$Q(t+1)$
0	0
1	1

Karakteristisk ligning for D flip-flop

$$Q(t+1) = D$$

# D Flip-Flop

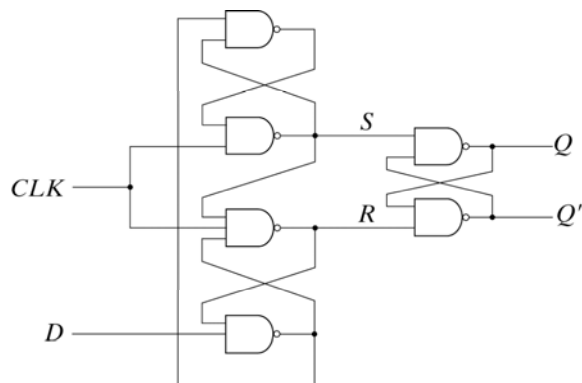
En positiv flanke-triggeret D flip-flop kan lages av to D latcher (Master-Slave)



Under Clk=0 er første D latch (master) transparent  
Under Clk=1 er siste D latch (slave) transparent

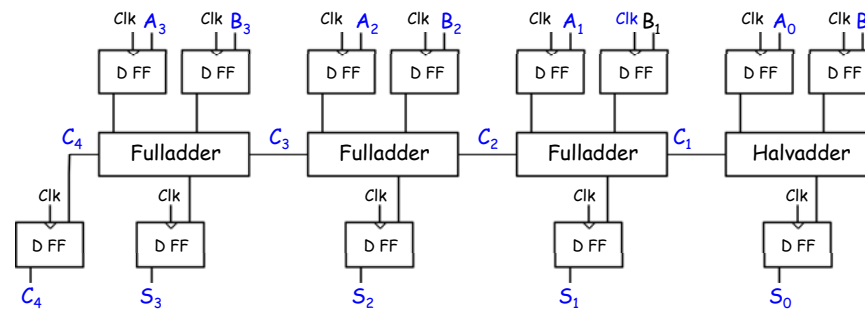


# D Flip-Flop, kompakt versjon



# D Flip-Flop, praktisk eksempel

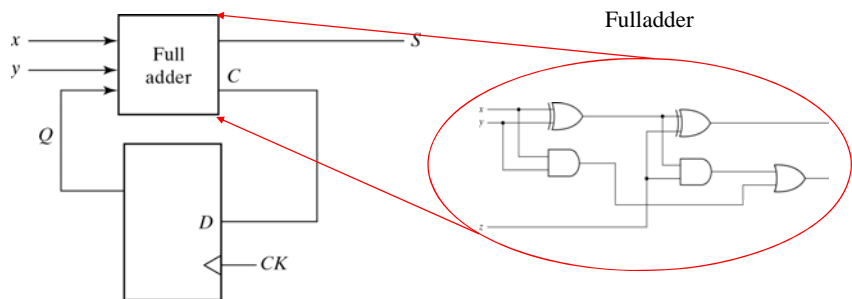
En rippeladder vil i et kort tidsrom gi gal sum ut.  
 Styring av signalflyt med D flip-flops kamouflerer dette



På positiv Clk flanke kommer nye data inn til adderen. I samme øyeblikk leses forrige (stabiliserte) sum ut.

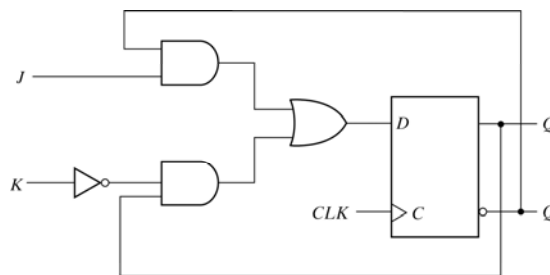
# D Flip-Flop, praktisk eksempel

Seriell adder

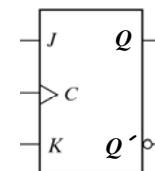


# JK Flip-Flop

Kretsoppbygging



Grafisk symbol



## JK Flip-Flop

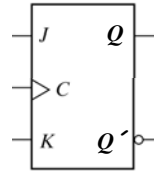
En JK flip-flop har følgende egenskaper

J=0, K=0: Utgang låst

J=0, K=1: Resetter utgang til "0"

J=1, K=0: Setter utgang til "1"

J=1, K=1: Inverterer utgang  $Q \rightarrow Q'$



Utgangen kan kun forandre verdi på stigende klokkeflanke

J	K	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	Q'(t)

En JK flip-flop er den mest generelle flip-floppen vi har

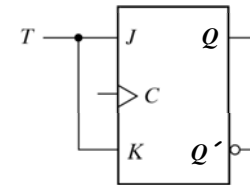
$$Q(t+1) = JQ'(t) + K'Q(t)$$

37

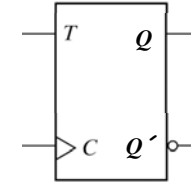
M.H

## T Flip-Flop

Kretsoppbygging



Grafisk symbol



38

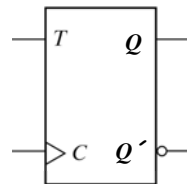
M.H

## T Flip-Flop

En T flip-flop har følgende egenskaper

T=0, Utgang låst

T=1, Inverterer utgang  $Q \rightarrow Q'$



Utgangen kan kun forandre verdi på stigende klokkeflanke

T	Q(t+1)
0	Q(t)
1	Q'(t)

Det er lett å lage tellere av T flip-flop'er

$$Q(t+1) = T \oplus Q(t)$$

39

M.H

## Oppsummering

- Låsekretser (latch'er)
  - SR latch bygget med NOR
  - S'R' latch bygget med NAND
  - D latch
- Flip-Flops
  - Master-Slave D flip-flop
  - JK flip-flop
  - T flip-flop

40

M.H