

Moment 2 - Digital elektronik

Föreläsning 2

Sekvenskretsar och byggblock

Jan Thim

F2: Sekvenskretsar och byggblock

- Innehåll:
 - Sekvenser
 - Latchar och vippor
 - Register
 - Introduktion - byggblock
 - Kodare / Avkodare
 - Multiplexer / Demultiplexer
 - Adderare
 - Paritetskretsar

Sekvenskretsar – Exempel

Exempel: Bankomat



Sekvens av operationer för att göra ett uttag:

- Sätt in kortet
- Mata in PIN-kod
- Mata in storleken på uttaget
- Vänta på pengarna
- Ta ut kort och pengar

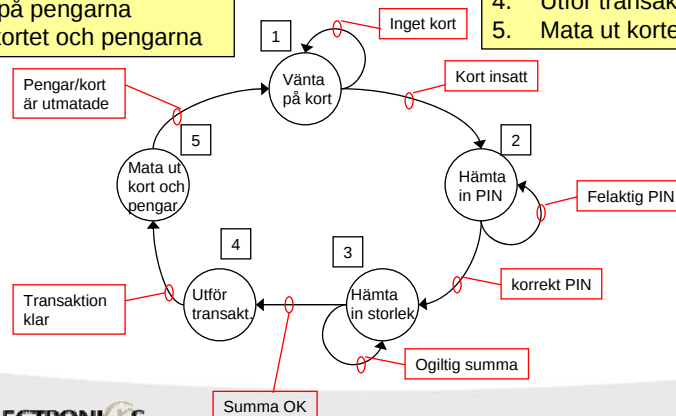
Forts. Exempel – Tillståndsdigram

"Operationer" utförda av dig:

1. Sätt in kortet
2. Mata in PIN-kod
3. Mata in storleken på uttaget
4. Vänta på pengarna
5. Ta ut kortet och pengarna

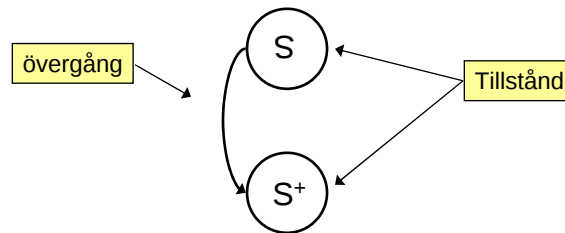
"Operationer" utförda av maskinen

1. Vänta på kortet
2. Hämta in PIN-kod
3. Hämta in storleken på uttaget
4. Utför transaktionen
5. Mata ut kortet och pengarna



Tillståndsdigram

- Tillståndsdigram (*eng. State Transition Graph*)
 - Visar varje individuellt tillstånd
 - Samtliga möjliga sekvenser av tillstånd sekvensnätet kan ha

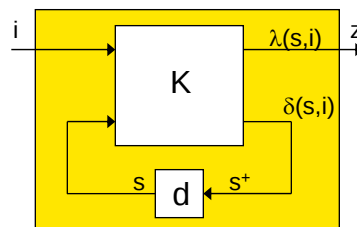


Tillståndsmaskin (*eng. Finite-State Machine*)

- Implementerar ett tillståndsdigram
- Består av
 - Tillståndsminne – innehåller maskinens tillstånd (s)

$$s(t_k) \xleftarrow{\text{d}} s^+(t_k) \quad s(t_k) = s^+(t_{k-1})$$

- Funktion för att beräkna nästa tillstånd (d)
- Funktion för att beräkna utgångarnas värde (l)



Tillståndsminne

Synkront D-element som tillståndsminne

$Q^+ = D$

Klocksignalen delar upp tiden i steg

- ett tidsdiskret system
- minnet fördröjer signalen en klockcykel
- alla förändringar i minnet sker samtidigt på aktiv flank
- klockfrekvensen $f = 1/T$

2009-03-05 Jan Thim

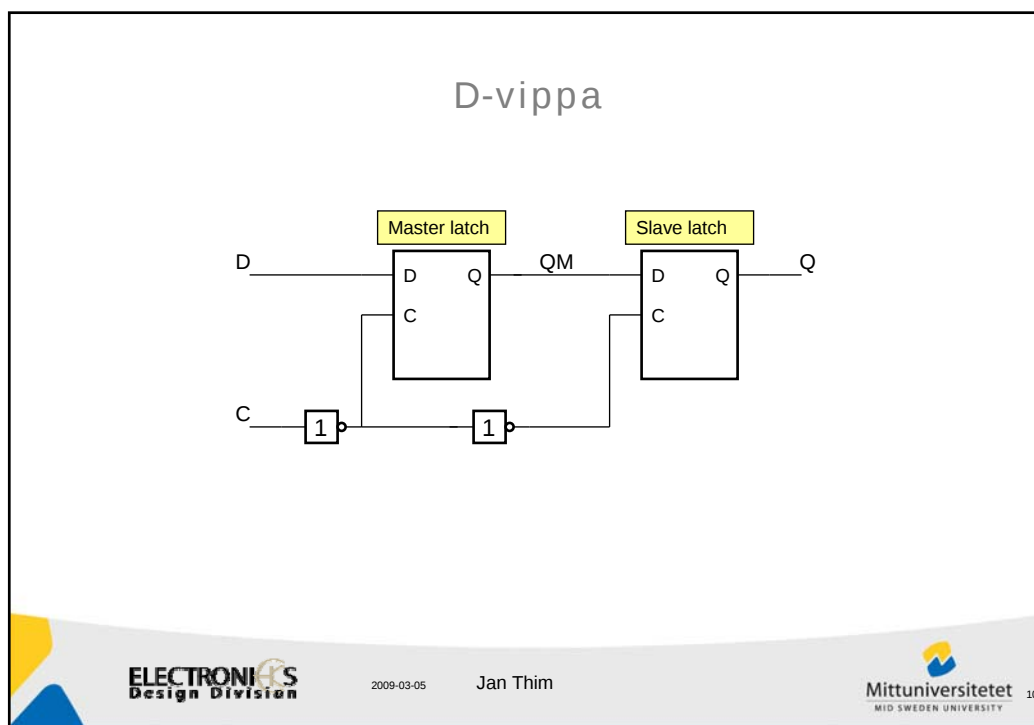
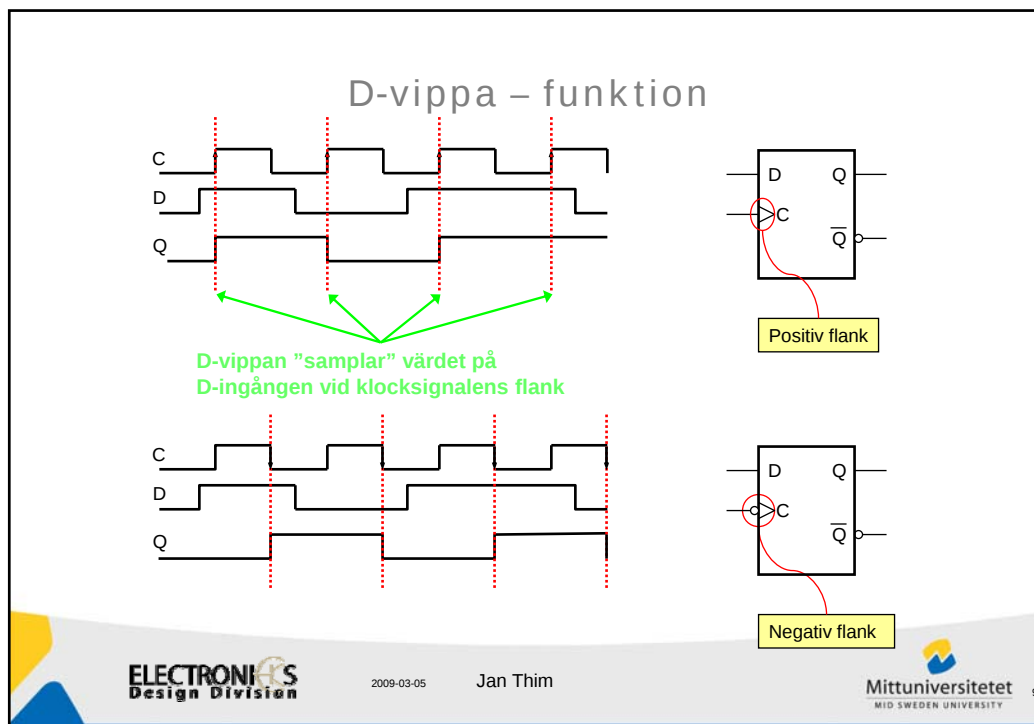
7

Typer av minneselement

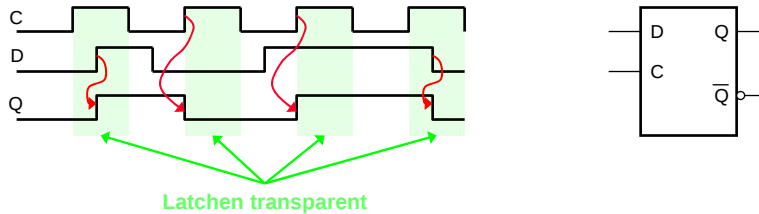
Symbol	Karakteristisk ekvation	Syntestabell																				
	$q^+ = D$	<table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <th>q</th><th>q+</th><th>D</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> </table>	q	q+	D	0	0	0	0	1	1	1	1	1	1	0	0					
q	q+	D																				
0	0	0																				
0	1	1																				
1	1	1																				
1	0	0																				
	$q^+ = T \cdot \bar{q} + \bar{T} \cdot q$	<table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <th>q</th><th>q+</th><th>T</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> </table>	q	q+	T	0	0	0	0	1	1	1	1	0	1	0	1					
q	q+	T																				
0	0	0																				
0	1	1																				
1	1	0																				
1	0	1																				
	$q^+ = J \cdot \bar{q} + \bar{K} \cdot q$	<table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <th>q</th><th>q+</th><th>J</th><th>K</th></tr> <tr><td>0</td><td>0</td><td>0</td><td>-</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>-</td></tr> <tr><td>1</td><td>1</td><td>-</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>-</td><td>1</td></tr> </table>	q	q+	J	K	0	0	0	-	0	1	1	-	1	1	-	0	1	0	-	1
q	q+	J	K																			
0	0	0	-																			
0	1	1	-																			
1	1	-	0																			
1	0	-	1																			

2009-03-05 Jan Thim

8

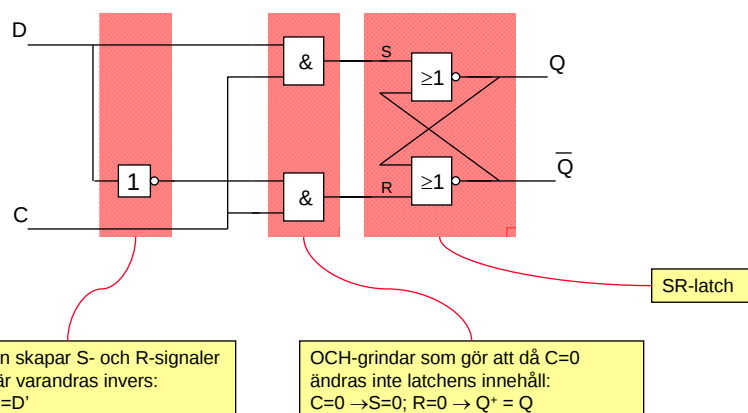


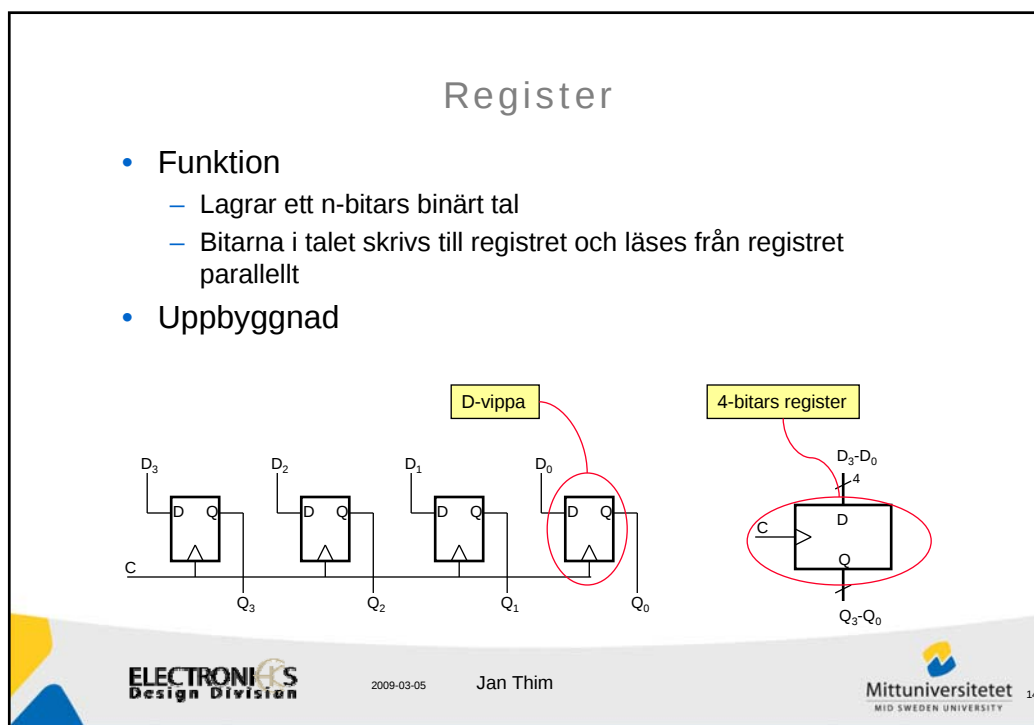
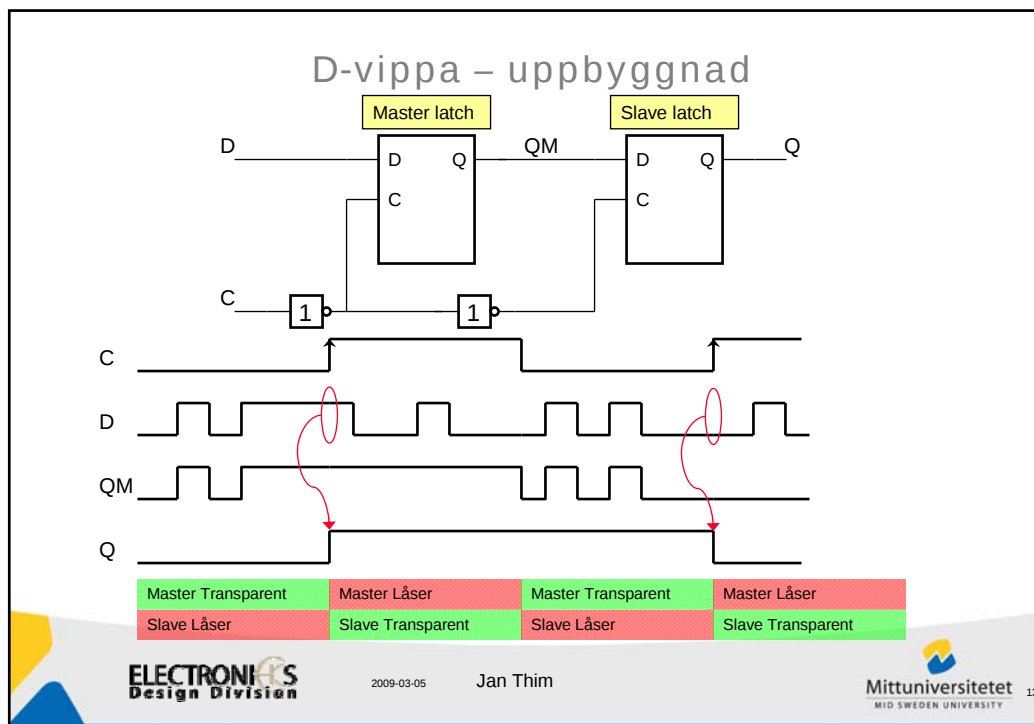
D-latch – funktion



- En D-latch kan vara i ett av två lägen
 - Transparent – Utgången (Q) följer värdet på ingången (D)
 - Lås – Värdet på utgången (Q) bibehålls och är oberoende av ingången (D)
- Ingången C kontrollerar i vilket läge latchen befinner sig i

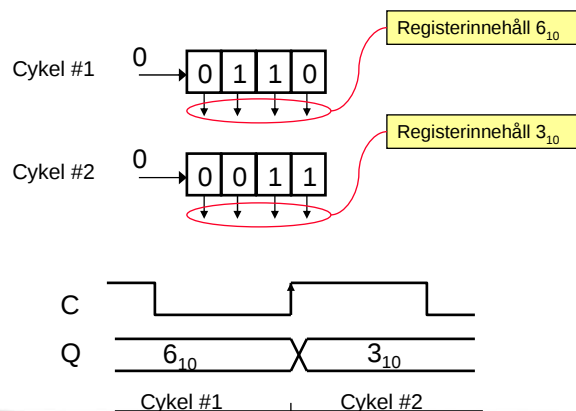
D-latch – uppbyggnad



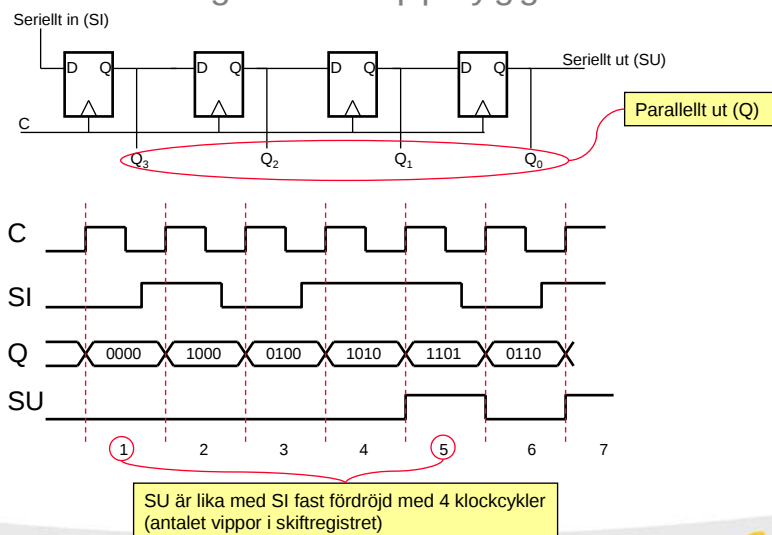


Skiftregister – funktion

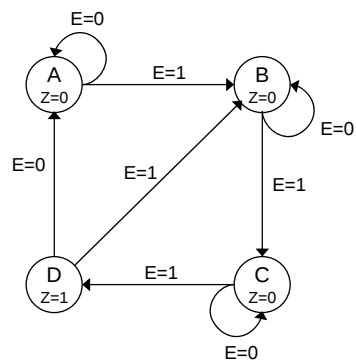
- Funktion
 - Flyttar talet i ett register i "sidled" åt vänster eller höger



Skiftregister – uppbyggnad

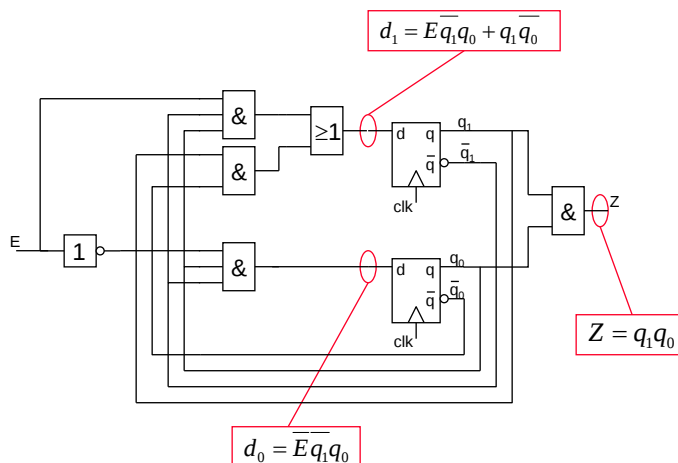


Exempel: tillståndsmaskin med D-vipor



S	E		Z
	0	1	
A	A	B	0
B	B	C	0
C	C	D	0
D	A	B	1
S ⁺			

Exempel: tillståndsmaskin med D-vipor



Minneselement, sammanfattning

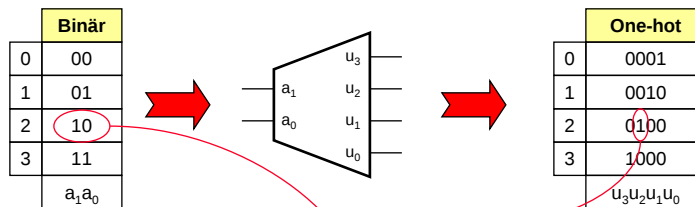
- Tillåter lagring av binära variabler för framtida beräkningar
 - Latch
 - Dess värde ändras vid klocksignalens aktiva nivå (t.ex C=1)
 - Grundläggande bistabilt minneselement
 - Lagrar 1 bit
 - Vippa (Flip-flop)
 - Dess värde ändras endast vid klocksignalens flank
 - Lagrar 1 bit
 - Används t.ex i tillståndsmaskiner och register
 - Register
 - En grupp Flip-flops som klockas med samma klocka
 - Lagrar en binär variabel med flera bitar

Introduktion, byggblock

- En del kretsar används så ofta att de inte konstrueras på nytt utan återanvänds som ett grundläggande byggblock
- Exempel på sådana typer av byggblock
 - Kod-omvandlare
 - Multiplexer
 - Aritmetiska enheter (+, -, ·)

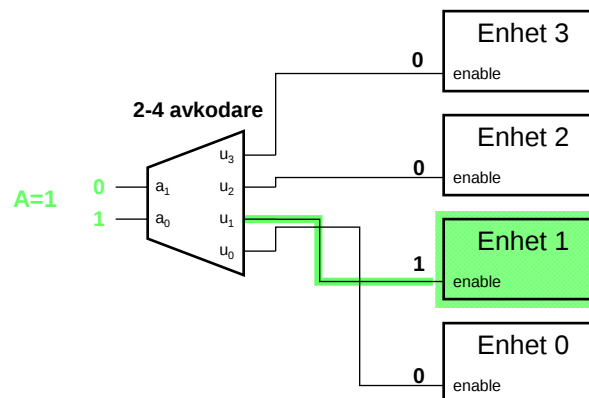
Binär avkodare

- Konverterar ett n-bitars binärt tal till ett 2n-bitars *One-hot* kod



Det binära talet A ($A=\{a_1, a_0\}$) sätter utsignalen u_A till '1' och övriga utsignaler till '0'.

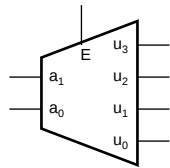
Exempel på användning av binär 2-4 avkodare



Väljer ut endast en enhet av flera

Avkodare med *enable*

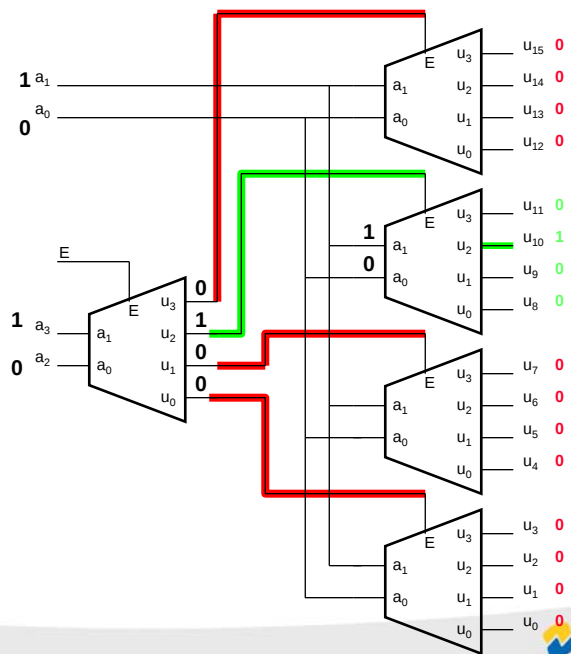
- Med *enable*-signal kan man från 2-4 avkodare bygga större avkodare



E	$a_1 a_0$	$u_3 u_2 u_1 u_0$
0	00	0000
0	01	0000
0	10	0000
0	11	0000
1	00	0001
1	01	0010
1	10	0100
1	11	1000

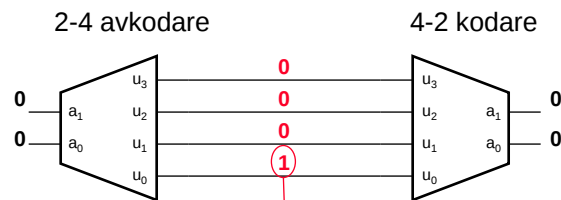
4-16 avkodare

$A=10_{10}=1010_2 \Rightarrow$
 $a_3 = 1$
 $a_2 = 0$
 $a_1 = 1$
 $a_0 = 0$



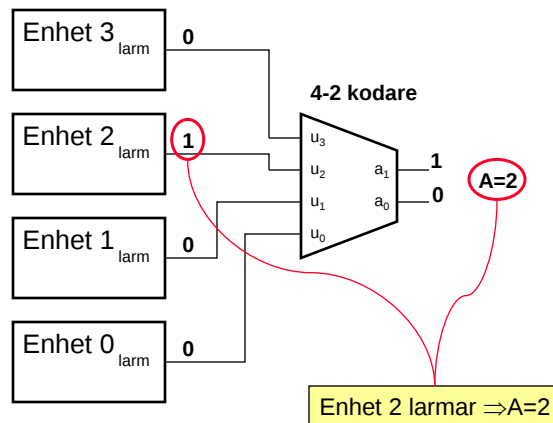
Kodare

- Har "invers" funktion av en avkodare



En och endast en ingång till kodaren får vara '1'.

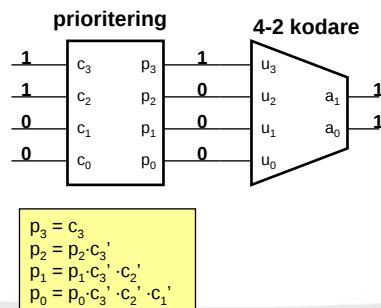
Exempel på användning av binär 4-2 kodare



Vad händer om två enheter larmar samtidigt?

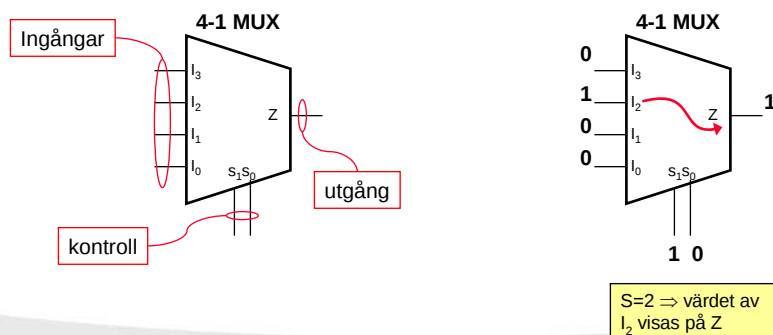
Prioritetskodare

- Ingångarna har inbördes prioritet
 - När mer än en ingång är aktiv så genereras en kod för den ingång som har högst prioritet
 - Exempel:
 - Låt insignal 3 ha högst prioritet, 2 näst högst, o.s.v



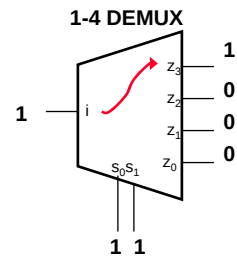
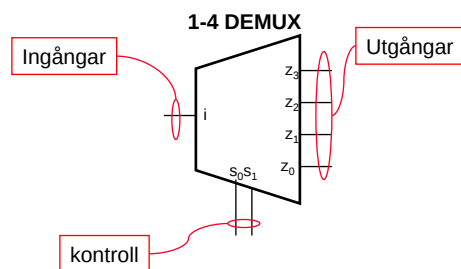
Multiplexer

- Ett binärt tal (S) kontrollerar vilken av ingångarna (Ij) som ska bestämma utsignalens (Z) värde



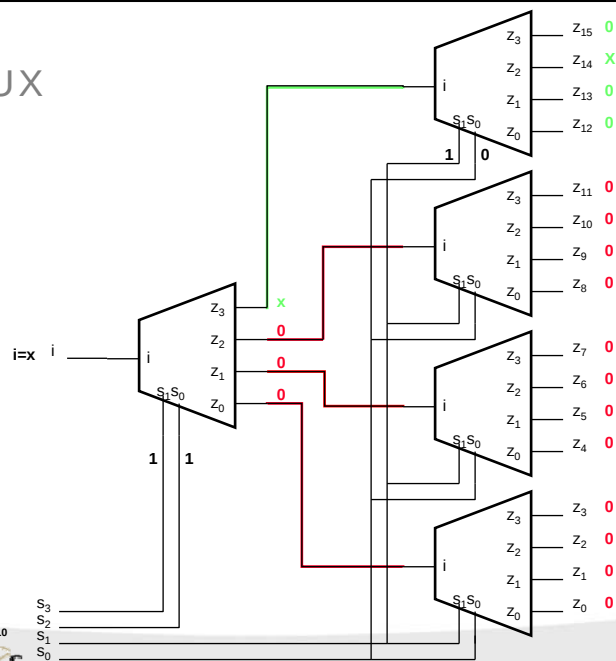
Demultiplexer

- Ett binärt tal (S) kontrollerar vilken av utgångarna (z_j) som ska tilldelas värdet av ingången (i)

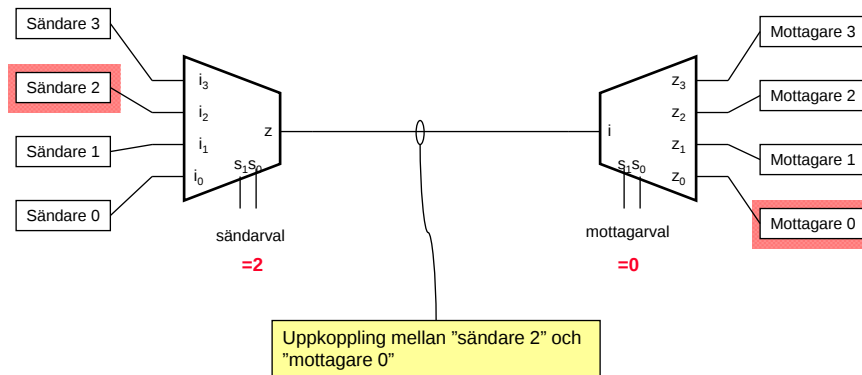


S=3 $\Rightarrow$ värdet av i visas på z_3

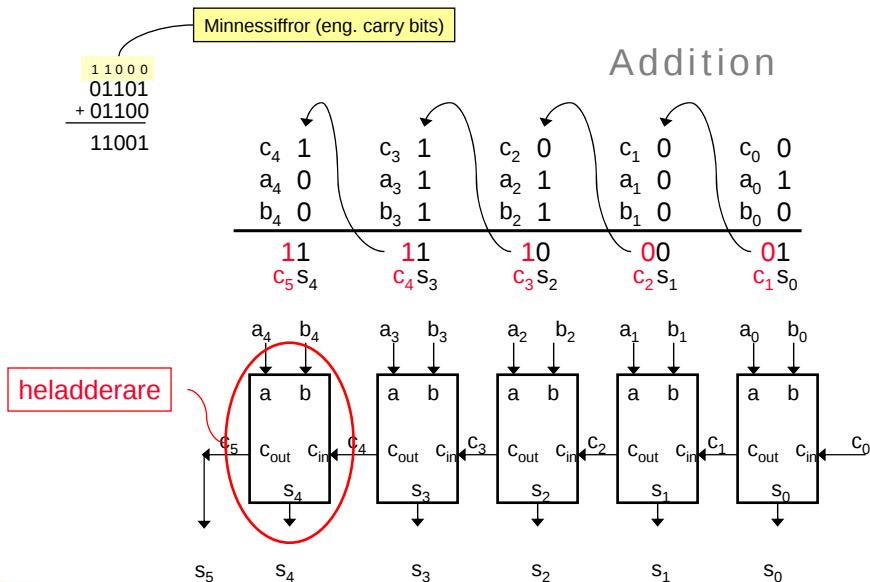
1-16 DMUX



Exempel på användning av MUX och DMUX



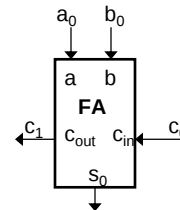
Addition



Heladderare (*eng. Full-adder*)

- Funktion

- Utför 1-bits addition
 - 1 bit för summa
 - 1 bit för *carry-out* (minnesbit)



a	b	C_{in}	C_{out}	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Minimering av K-diagram

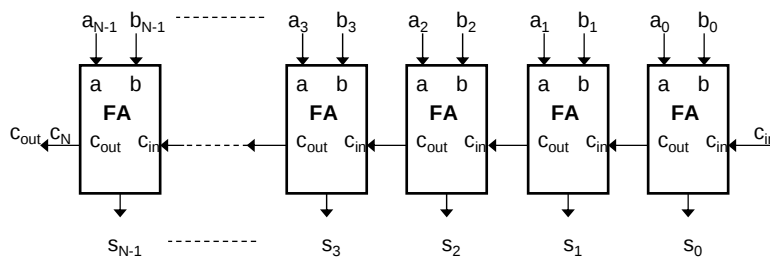


$$C_{out} = ab + ac_{in} + bc_{in}$$

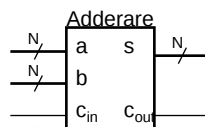
$$s = \overline{a}\overline{b}c_{in} + \overline{a}b\overline{c_{in}} + a\overline{b}\overline{c_{in}} + abc_{in}$$

Fler-bits adderare

- N-bitars adderare
 - Kaskadkopplade Full-adderare

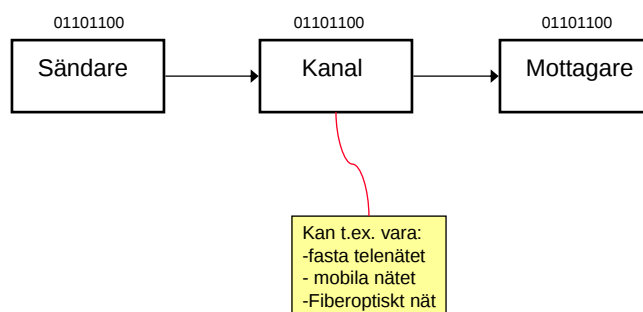


- ♦ Symbol:



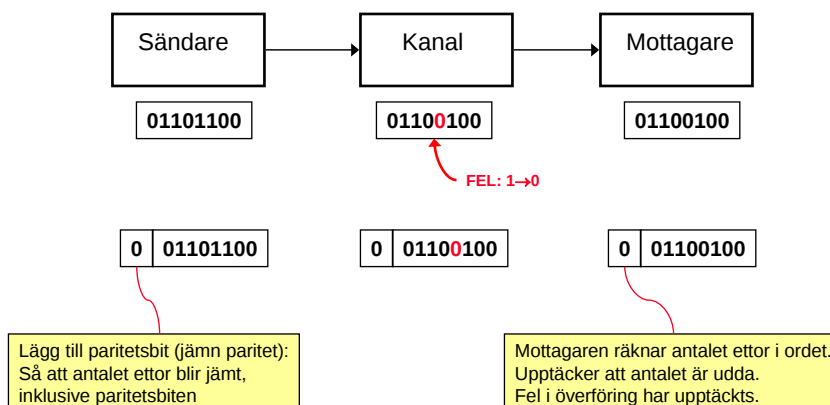
Felupptäckande koder

- Modell för informationsöverföring



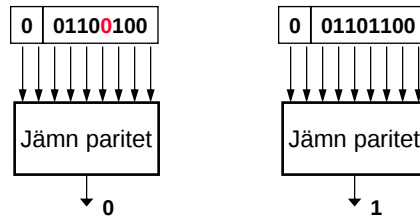
Felupptäckande koder

- Paritetsbit som felupptäckande kod

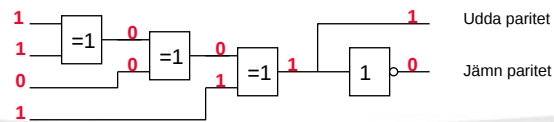


Paritetskretsar

- Krets för jämn paritet
 - Ger '1' ut om antalet 1:or in är jämnt



- ♦ Exempel på 4-bitars paritetskrets med xor-grind



SLUT på Föreläsning 2

