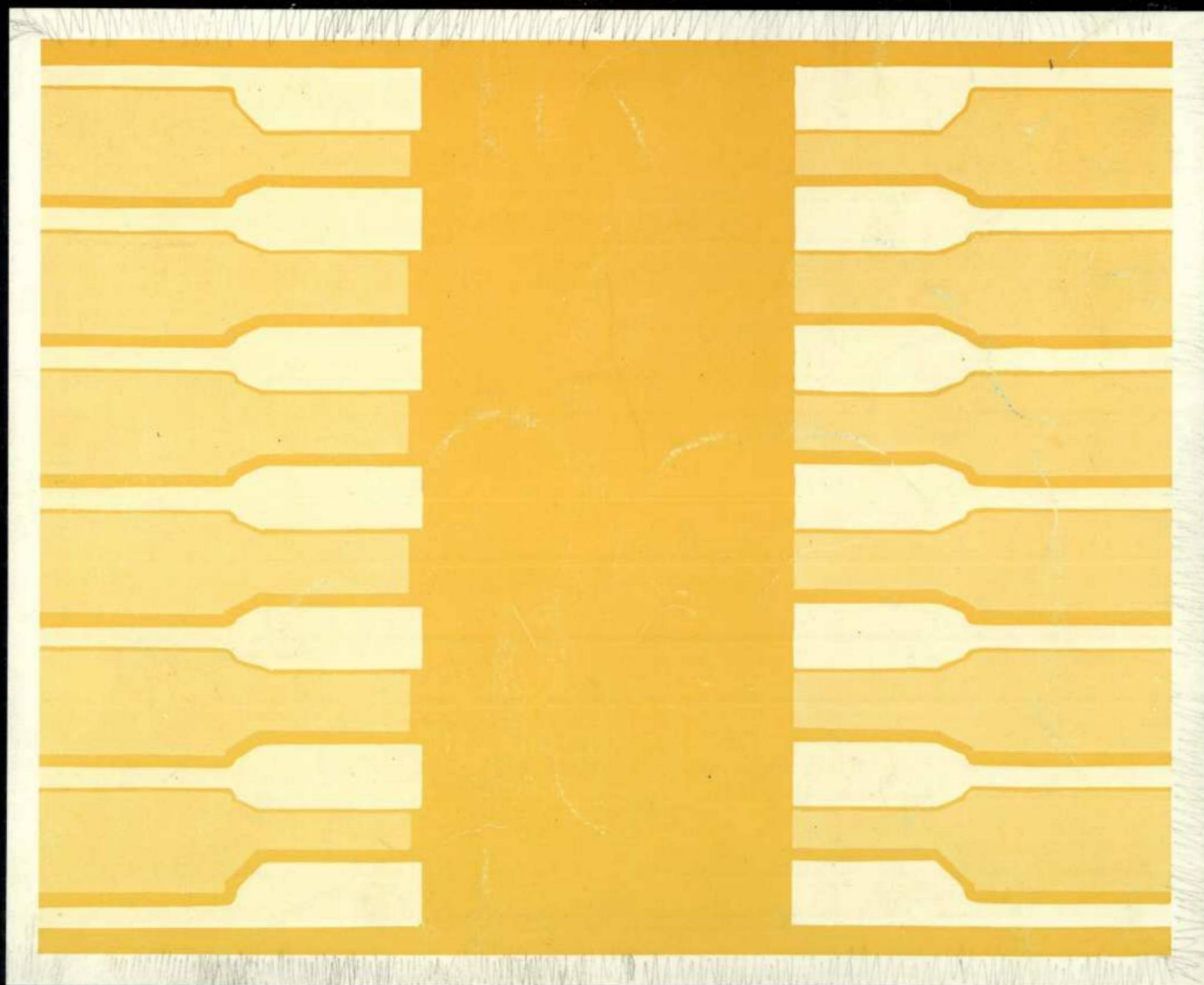


elektronik

Digitala kretsar

Gunnar Markesjö



93

95

98-99

102

104

109

125 dioderna

Memory enable

lay out

143

OBS! Stegförhöjning i datablad
FAN OUT (gamla skrivningen)

Birgitta Stenar TE4a

Elektronik

Digitala kretsar

Gunnar Markesjö

ISBN 91-24-26209-9

Best. nr 24-26209-9

(24-26209-9)B

Teckningar: Nils Svensson

Fotografier: Erwin Bennedich

Omslag efter material från Svenska AB Philips

© 1976, Gunnar Markesjö och ESSELTE STUDIUM AB

Första upplagan, första tryckningen

Esselte Herzogs, Nacka 1976

Innehåll

Studieplanering V

Inledning VI

D1 Kombinationskretsar 1

D2 Sekvenskretsar 29

D3 TTL-kretsar 57

D4 CMOS-kretsar 89

D5 Halvledarminnen 115

Svenskt sakregister 148

Engelskt sakregister 151

Förord

I den nya studieplanen för elektronikundervisningen i gymnasiets fjärde årskurs (TTt 4) har puls- och digitaltekniken givits större utrymme jämfört med tidigare studieplaner. En omfattande enkät till svenska industrier och institutioner har visat att denna förskjutning väl avspeglar behovet på arbetsmarknaden.

Böckerna Pulskretsar och Digitala kretsar är utarbetade efter studieplanens riktlinjer och med erfarenheter från tidigare kurser både för gymnasieundervisning, högskoleundervisning och fortbildning inom industrin.

Varje kapitel är avsett att lägga grunden för ett 12-timmars studiepaket. Varje paket kan lämpligen inledas med en introduktiv stillfilm. Sveriges Radio Utbildningsavdelning har år 1975 reviderat 19 tidigare stillfilmer samt dessutom producerat 6 nya som väl täcker det i studieplanen utökade omfånget för digitaltekniken.

Häften innehållande övningar och laborationer utarbetas läsåret 75/76 av Jan Rustan Törnquist.

Omfånget av varje kapitel är tilltaget i överkant för att ge lärare större valfrihet och högpresterande elever sysselsättning. Förslag till ett lämpligt urval som "normalkurs" kommer att göras i övnings- och laborationshäftet. Urvalet kommer därvid att göras med hänsyn till laborativa moment och lämpliga övningsuppgifter.

En provisorisk upplaga av kursböckerna har hösten 1975 körts vid internutbildningen vid LME.

Manuskriptet har granskats av konsulent Hans Blomqvist, SÖ.

Jag vill tacka alla som bidragit till kursböckernas tillkomst och därvid speciellt min hustru Friede som tillbringat hela sommaren vid skrivmaskinen.

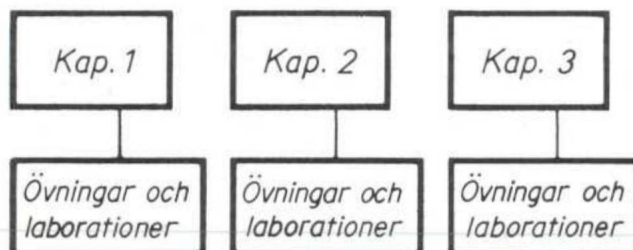
Aspö i augusti 1975

Gunnar Markesjö

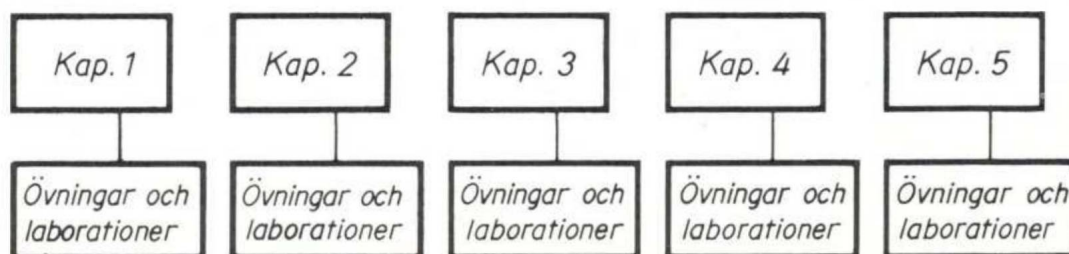
Studieplanering

Innehållet i kursböckerna Pulskretsar och Digitala kretsar kan lämpligen uppdelas i 8 avsnitt enligt nedanstående figur. Varje avsnitt kan utgöra ett beting om totalt 12 timmar schemalagd tid (inklusive laboration och övning).

Pulskretsar



Digitala kretsar



Till kursböckerna utarbetas övningsböcker med laborationshandledning. Kursmaterialet är så utformat att det skall ge elever och lärare möjlighet att fritt välja ut vissa avsnitt som överkurs.

Inledning

Digitala kretsar (digital circuits) utgör byggblocken i digitala system. Det är tack vare de integrerade kretsarna som digitaltekniken har utvecklats så enormt det senaste decenniet. I detta häfte skall vi ge en översikt över de viktigaste digitala byggblocken och deras egenskaper.

Man skulle förenklat kunna säga att det är två typer av grundkretsar som bygger upp alla digitala system, nämligen grindar och vippor. I första kapitlet skall vi behandla grindarna och hur de kan kopplas ihop till kombinationskretsar av olika slag.

I andra kapitlet skall vi bekanta oss med vippor och hur de kan kopplas samman till räknare och register, eller med ett finare namn sekvenskretsar av olika slag.

Digitalteknikens snabba utveckling hänger som sagt ihop med den teknologiska utvecklingen av planartekniken. Det är i dag två logikfamiljer som tävlar om 90 % av marknaden. Vi skall ägna dessa logikfamiljer var sitt kapitel, ett om TTL-kretsarna och ett om CMOS-kretsarna.

Utvecklingen av halvledarminnen och mikroprocessorer har givit upphov till en ny trend inom digitaltekniken. Man övergår i stor utsträckning till "programstyrda kretsar" i stället för de komplicerade grindnät, "wired logic" eller "random logic", som tidigare användes och där man vid konstruktioner utnyttjade Boolesk algebra och olika minimeringsmetoder. Vi skall därför ägna det avslutande kapitlet åt halvledarminnen.

D1 Kombinationskretsar

Kombinationskretsar 2

1 Logiska grundfunktioner 2

2 Diod- och transistorgrindar 6

- 2.1 Positiv och negativ logik 6
- 2.2 NAND- och NOR-kretsar 7
- 2.3 Grindar i en logikfamilj 9
- 2.4 Exempel på användning av grindar 15

3 Planarteknik 19

- 3.1 Typgrinden i RTL-familjen 20
- 3.2 Tillverkningen av en RTL-krets 21

4 Logikfamiljernas mångfald 23

- 4.1 Sving och störmarginaler 24
- 4.2 HNL – en störsäker logik 25
- 4.3 ECL – en snabb logik 26

Repetitionsfrågor 28

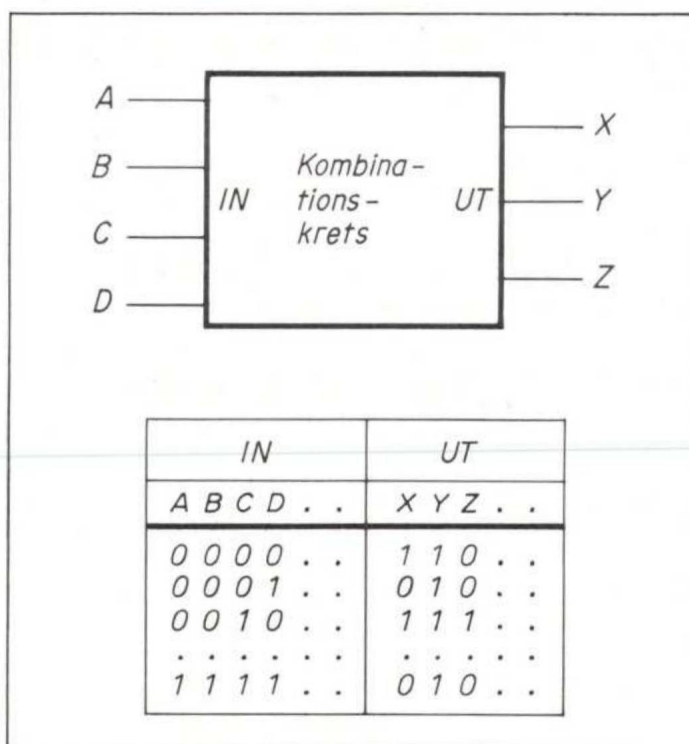
Målruta

När du läst detta kapitel skall du kunna:

- rita symboler och sanningstabeller för AND-, OR-, NAND-, NOR- och EXOR-funktionerna
- rita och förklara funktionen hos typgrindarna i DTL-, RTL-, HNL- och ECL-familjerna
- rita grindscheman för AOI-krets, jämförare, multiplexer och demultiplexer samt adderare
- förklara bussens funktion och användning
- rita kristallstruktur för integrerad bipolartransistor, diod och resistor
- jämföra ECL- och HNL-familjerna med hänsyn till sving, störmarginaler och snabbhet

Kombinationskretsar

Det här kapitlet skall behandla grindar och hur dessa kan byggas samman till mer komplexa kombinationskretsar eller kombinatoriska kretsar. En kombinationskrets är enligt IEC "en digital krets för vilken det existerar en och endast en kombination av utsignaler för varje möjlig kombination av insignaler". Figur 1.1.



Figur 1.1
Exempel på kombi-
nationskrets och
dess sanningstabell.

Eftersom utsignalerna alltid är entydiga funktioner av insignalkombinationerna kan kombinationskretsens funktion beskrivas med en tabell. Betecknar vi signalvärdena med 0 och 1 kallas en sådan tabell för sannings-tabell (truth table).

Innan vi ger oss in på den elektriska uppbyggnaden av kombinationskretsar måste vi rekapitulera några vanliga uttryck från den binära algebra, Boolesk algebra, som brukar användas för beskrivning och realisering av olika kombinatoriska funktioner.

Vi skall därefter bekanta oss med grindar av olika komplexitet och för att framställningen skall bli lättförståelig använder vi härvid enbart grindar av DTL-typ (diod-transistor-logik).

Som avslutning på kapitlet skall vi repetera de viktigaste stegen i planar-processen och därefter visa typgrindarna i några av de många logik-familjer som förekommer på marknaden.

1 Logiska grundfunktioner

För att beskriva digitala kretsars funktion brukar man använda den algebra som George Boole utvecklade på 1800-talet för behandling av logiken i påståenden. Det är en binär algebra som använder variabelvärdena "sann" och "falsk" och därför ser man ibland dessa uttryck även i samband med elektroniska kretsar.

Eftersom man i digitaltekniken ständigt möter logiska uttryck för beskrivning av logikkretsarnas funktion skall vi nedan ge en kort sammanfattning. (Det förutsätts här att binära tal och logisk algebra tidigare introducerats i ämnet matematik och kommer att behandlas utförligare i ämnet regler-teknik.)

Man kan bygga upp godtyckliga kombinatoriska funktioner (dvs samband specificerade i tabellform enligt figur 1.1) enbart med hjälp av tre logiska grundfunktioner: OCH, ELLER samt ICKE. Det kan vara värt att observera att logiskt ELLER inte exakt motsvarar innebörden av ordet "eller" i normalt språkbruk. Tänk efter vad det är för skillnad! I data-blad och fackpress används normalt de engelska termerna AND, OR och NOT och vi kommer i fortsättningen att använda både svenska och engelska termer. Figur 1.2 visar dessa grundfunktioner, dels i tabellform, dels som grafiska symboler. När det gäller symboler håller man sakta på att övergå till de internationellt standardiserade fyrkantiga symbolerna. Under lång tid framöver får man emellertid räkna med att både data-blad och facklitteratur kommer att använda de amerikanska symbolerna. Vi kommer i fortsättningen att använda såväl IEC:s symboler som de amerikanska.

Den logiska algebran har alltså binära variabler, dvs variabler som enbart kan anta två värden "noll" eller "ett". Grundfunktionerna har egna funktionsbeteckningar, vilket framgår av figur 1.2c.

I figur 1.3 återges några viktiga räkneregler och satser ur den Booleska algebran. Ekv (1.1) till (1.4) postulerar enkla räkneregler för AND- och OR-funktionerna. De motsvarar den vanliga algebras multiplikations-

NOT	AND	OR																																				
a) <u>Sanningstabeller</u>																																						
<table><tr><th>A</th><th>X</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	X	0	1	1	0	<table><tr><th>B</th><th>C</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	B	C	Y	0	0	0	0	1	0	1	0	0	1	1	1	<table><tr><th>D</th><th>E</th><th>Z</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>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	D	E	Z	0	0	0	0	1	1	1	0	1	1	1	1
A	X																																					
0	1																																					
1	0																																					
B	C	Y																																				
0	0	0																																				
0	1	0																																				
1	0	0																																				
1	1	1																																				
D	E	Z																																				
0	0	0																																				
0	1	1																																				
1	0	1																																				
1	1	1																																				
b) <u>Grafiska symboler</u>																																						
<u>IEC</u> 																																						
<u>USA</u> 																																						
c) <u>Funktionsbeteckningar</u>																																						
$X = \bar{A}$ eller $X = A^*$ $X = A'$	$Y = B \cdot C$ eller $Y = B \wedge C$	$Z = D + E$ eller $Z = D \vee E$																																				

Figur 1.2
De logiska grundfunktionerna NOT, AND och OR.

och additionsregler och varje ekvation är uppställd i två "duala" former, den ena för AND och den andra för OR-funktionen. "Postulaten" i ekv (1.1) till (1.4) är egentligen endast ett annat sätt att uttrycka sannings-tabellerna som definierar grundfunktionerna i figur 1.2a.

<u>Postulat (räkneregler):</u>		
$0 \cdot 0 = 0$	(1.1a)	$1 + 1 = 1$ (1.1b)
$1 \cdot 1 = 1$	(1.2a)	$0 + 0 = 0$ (1.2b)
$1 \cdot 0 = 0$	(1.3a)	$0 + 1 = 1$ (1.3b)
$\overline{0} = 1$	(1.4a)	$\overline{1} = 0$ (1.4b)
<u>Satser för en variabel:</u>		
$A \cdot 1 = A$	(1.5a)	$A + 0 = A$ (1.5b)
$A \cdot 0 = 0$	(1.6a)	$A + 1 = 1$ (1.6b)
$A \cdot A = A$	(1.7a)	$A + A = A$ (1.7b)
$A \cdot \overline{A} = 0$	(1.8a)	$A + \overline{A} = 1$ (1.8b)
$\overline{(\overline{A})} = A$	(1.9a)	$\overline{(\overline{A})} = A$ (1.9b)
<u>Satser för flera variabler:</u>		
<u>"Kommutativa lagen":</u>		
$AB = BA$	(1.10a)	$A + B = B + A$ (1.10b)
<u>"Associativa lagen":</u>		
$A(BC) = (AB)C$	(1.11a)	$A + (B + C) = (A + B) + C$ (1.11b)
<u>"Distributiva lagen":</u>		
$A(B + C) = AB + AC$	(1.12a)	$A + BC = (A + B)(A + C)$ (1.12b)
<u>"Absorptionssatsen":</u>		
$A + AB = A$	(1.13a)	$A(A + B) = A$ (1.13b)
<u>de Morgans teorem:</u>		
$\overline{A \cdot B \cdot C} = \overline{A} + \overline{B} + \overline{C}$	(1.14a)	$\overline{A + B + C} = \overline{A} \cdot \overline{B} \cdot \overline{C}$ (1.14b)

Figur 1.3
Några viktiga räkne-
regler och satser ur
Booles algebra.

Med utgångspunkt från de postulerade räknereglererna kan olika satser för en eller flera variabler ställas upp och verifieras. Satserna i ekv (1.5) till (1.9) avser en variabel. Satserna i ekv (1.10) till (1.13) avser flera variabler.

Låt oss för övnings skull verifiera ekv (1.13a). Vänstra ledet i ekvationen, $A + AB$, innehåller två variabler A och B. Två binära variabler ger 4 möjliga kombinationer av vänstra ledet (VL):

VL:

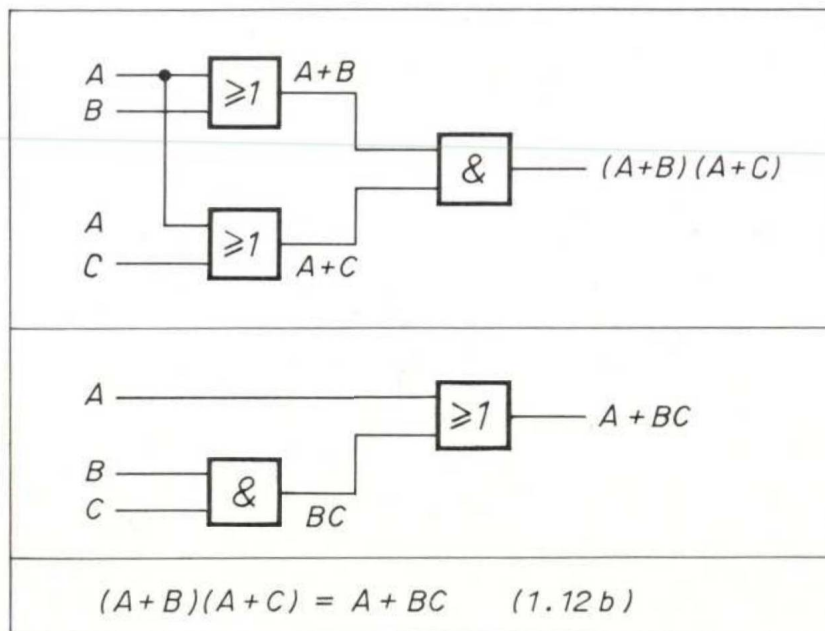
A	B	$A + AB$	enligt ekv
0	0	$0 + 0 \cdot 0 = 0$	(1.1a) och (1.2b)
0	1	$0 + 0 \cdot 1 = 0$	(1.3a), (1.10a) och (1.2b)
1	0	$1 + 1 \cdot 0 = 1$	(1.3a), (1.10b) och (1.3b)
1	1	$1 + 1 \cdot 1 = 1$	(1.2a) och (1.1b)

Vänstra ledet stämmer alltså i samtliga positioner med värdet på A (i vänstra kolumnen). Därmed är satsen verifierad.

En av satserna för flera variabler är särskilt viktig. Det är ekv (1.14) som också kallas de Morgans teorem. Vi kommer att använda de Morgans teorem flitigt i fortsättningen.

Vi kan också använda de uppställda satserna för att förenkla logiska uttryck, vilket är liktydigt med att förenkla ("minimera") grindnät. Figur 1.4 visar som exempel hur VL och HL i ekv (1.12b) kan ritas ut som grindnät. Enligt ekv (1.12b) ger näten samma funktion. Grindnätet i figur 1.4a kan därför ersättas av det enklare nätet i figur 1.4b utan att funktionerna ändras.

Vi har nu sett att det går att behandla kombinatoriska funktioner med hjälp av logisk algebra. Vi skall nu realisera några viktiga logikfunktioner med diod- och transistorgrindar.

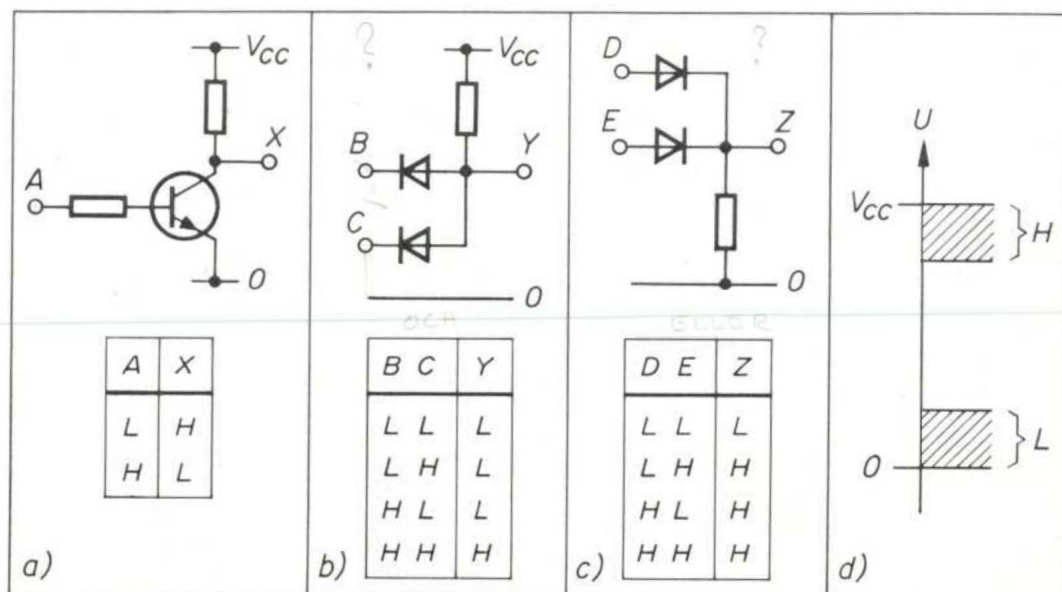


Figur 1.4
Förenkling av grindnät med hjälp av Booles algebra.

2 Diod- och transistorgrindar

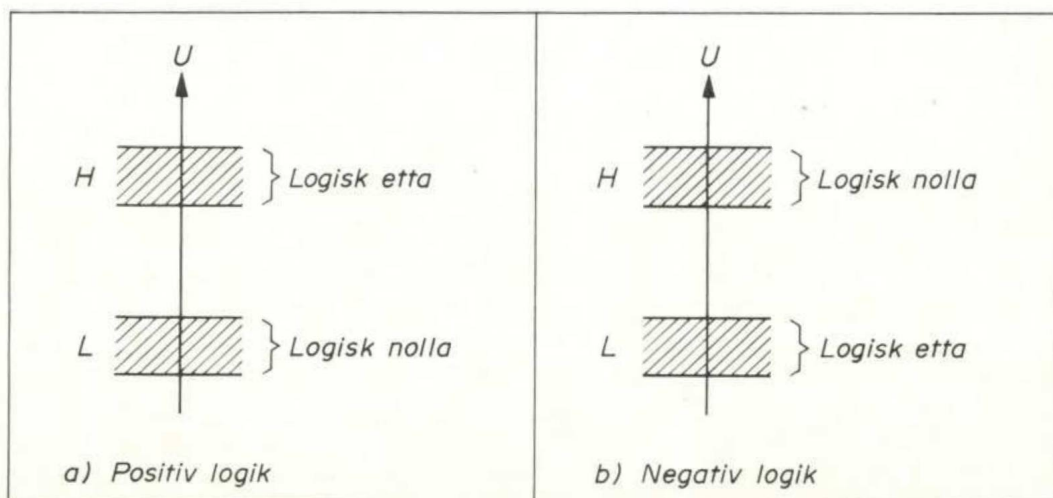
2.1 Positiv och negativ logik

I figur 1.5 har vi ritat tre välkända kretsar, en inverterare och två diodgrindar. Insignalerna (A - E) är binära och vi antar att de ligger inom de spänningsområden som betecknats H (dvs hög) och L (dvs låg) i figur 1.5d. Under varje krets har vi ritat en funktionstabell som anger samhörande värden på utspänning vid varje möjlig kombination av inspanningar. Vilken logisk funktion har nu våra kretsar i figur 1.5? Ja, det beror på hur vi representerar de logiska variabelvärdena noll och ett.



Figur 1.5
Inverterare och två diodgrindar.

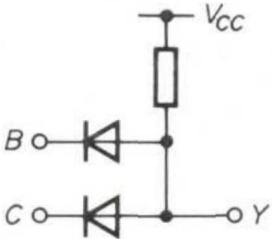
Man kan enligt figur 1.6a använda hög representation som också kallas positiv logik eller aktiv hög. Vi låter därvid det högre (mest positiva) spänningsområdet betyda logisk nolla. Våra funktionstabeller i figur 1.5 övergår då till de sanningstabeller vi tidigare mött i figur 1.2. Diodgrinden i figur 1.5b blir vid positiv logik en AND-grind och grinden i figur 1.5c blir en OR-grind.



Figur 1.6
Positiv och negativ logik.

Om vi som i figur 1.6b låter hög spänning betyda logisk nolla och låg spänning logisk etta får vi det omvända förhållandet. Grinden i figur 1.5b blir en OR-grind vid negativ logik och grinden i figur 1.5c en AND-grind vid negativ logik. För inverteraren i figur 1.5a spelar däremot representationen ingen roll.

Det är alltså väsentligt när man talar om logiska funktioner (som AND och OR) att ha klart för sig vilken representation som används! En och samma grind kan - beroende på representationen - ge två olika logiska funktioner, figur 1.7.

Diodgrind	Representation	
	Positiv logik hög nivå = 1  $Y = B \cdot C$	Negativ logik låg nivå = 1  $Y = B + C$

Figur 1.7
En diodgrind och dess logiska funktion.

För att undvika missförstånd kommer vi enbart att använda positiv logik i fortsättningen.

I databladen för digitala kretsar vill man undvika missförstånd och därför har man inom IEC beslutat att digitala kretsar skall specificeras med funktionstabeller, dvs med variabelvärdena H och L, och inte med sanningstabeller, dvs med variabelvärdena 0 och 1.

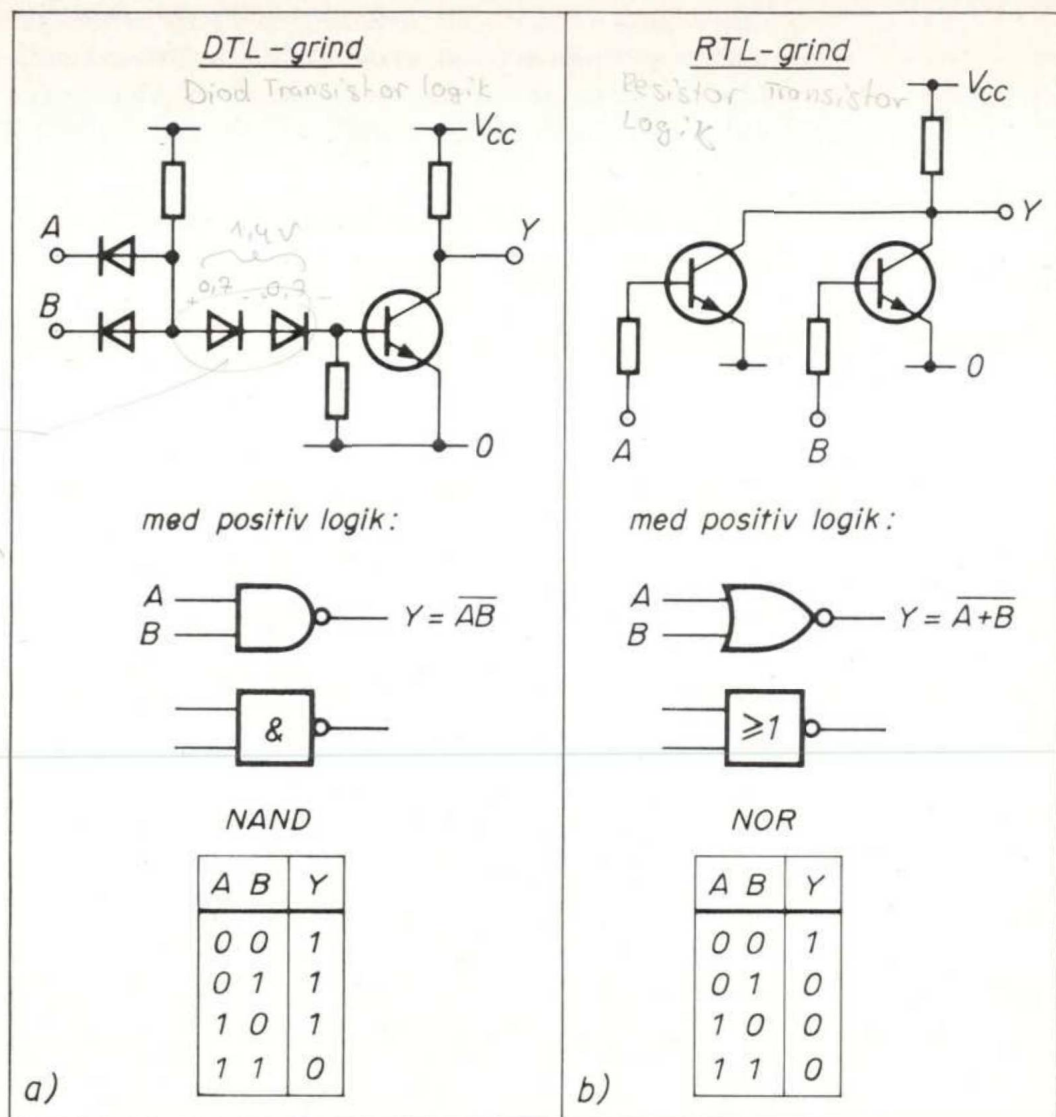
2.2 NAND- och NOR-kretsar

För att man skall få önskade logiska funktioner måste som regel ett flertal logikgrindar kopplas samman. Därvid kan man inte använda enbart diodgrindar, exempelvis sådana som i figur 1.5b och c, för då skulle diodernas framspänningsfall adderas och signalnivåerna skulle hamna utanför de tillåtna områdena, figur 1.5d.

För att hålla de logiska nivåerna intakta använder man i praktiken enbart kombinationer mellan diodgrindar och transistorswitchar. Figur 1.8a visar ett exempel där dioderna utför den logiska operationen och transistorn återställer spänningsnivåer så att efterföljande grind får rätta signalnivåer. Transistorn medför invertering och därför blir funktionen i figur 1.8a AND med invertering eller kortare NAND, dvs NOT AND. Inverteringen är i symbolen markerad med en ring på utgången.

NAND-kretsen i figur 1.8 är "typgrind" i DTL-familjen. (DTL = diode transistor logic). Genom att variera antalet ingångar eller kombinera typgrindar till nya funktioner får man ett helt sortiment grindar som bildar en "familj".

två stycken
för att öka
stör marginalen



Figur 1.8

En NAND-grind och en NOR-grind (positiv logik).

invertering därefter
låg signal in ger hög signal ut

Med logikfamilj menar man normalt ett antal olika logikkretsar som är avsedda att samarbeta i ett och samma digitala system. Kretsarna i en familj kännetecknas av att de har likartad uppbyggnad, samma miljötolighet, samma matningsspänning, gemensamma regler för drivning och belastning, ungefär samma snabbhet och en sådan mekanisk utformning (kapseltyp) att de bildar ett byggsystem inte enbart i elektrisk mening utan även rent byggtkniskt.

Det är väsentligt att lära känna typgrinden för en logikfamilj, eftersom alla övriga kretsar i samma familj har samma typ av diodgrindar på ingången och samma typ av transistor på utgången.

RTL (resistor transistor logic) är en logikfamilj som ej har dioder i sin typgrind, figur 1.8b. Eftersom transistorswitcharnas kollektorer är hopkopplade får man låg utgång så snart någon av transistorerna bottnas. Det blir en OR-funktion med invertering, eller kortare NOR.

NAND- och NOR-kretsarna är i praktiken de viktigaste byggblocken i alla digitala system.

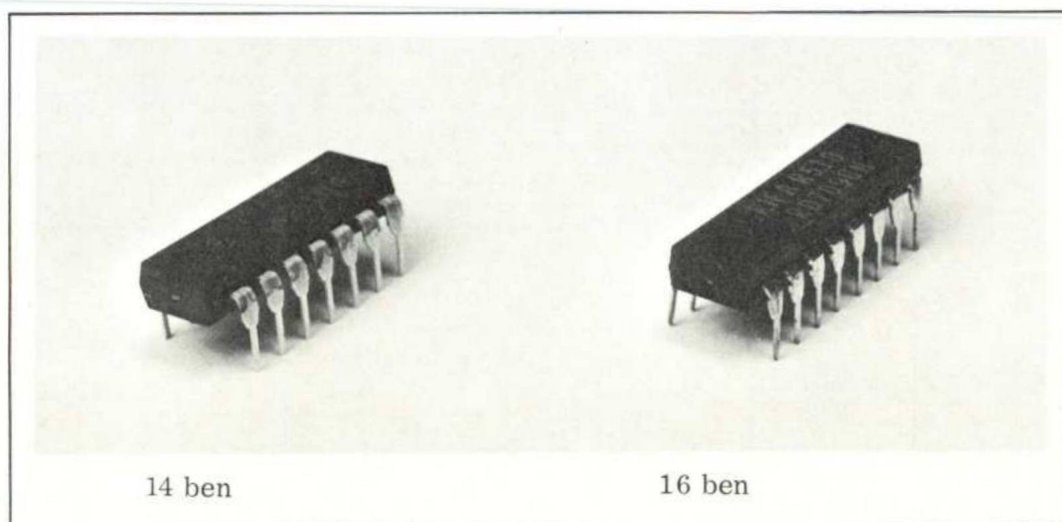
Typgrindarnas specifikation och egenskaper skall vi återkomma till senare. Vi skall endast kort kommentera de två dioder som lagts i serie med basen i NAND-grinden i figur 1.8a. Dessa dioder har till uppgift att förskjuta omslagsnivån på ingången till en spänning av ca 1,4 V och därmed förhindra att små störsignaler på ingången kan förorsaka omslag på utgången. Dioderna ökar på detta sätt kretsens störmarginaler - men mer om detta senare!

2.3 Grindar i en logikfamilj

NAND- och NOR-grindar utgör typgrindar i många logikfamiljer. Dessa familjer byggs upp på varianter av typgrindarna. Vi skall nedan som exempel se några typiska kretsar ur DTL-familjen och den närbesläktade TTL-familjen.

2.3.1 Flera grindar i samma kapsel

Det mest ekonomiska sättet att kapsla in IC-kretsar är att använda DIL-kapslar (DIL = dual in line) med 14 eller 16 uttag (pinnar, ben). Figur 1.9.



Figur 1.9
DIL-kapslar.

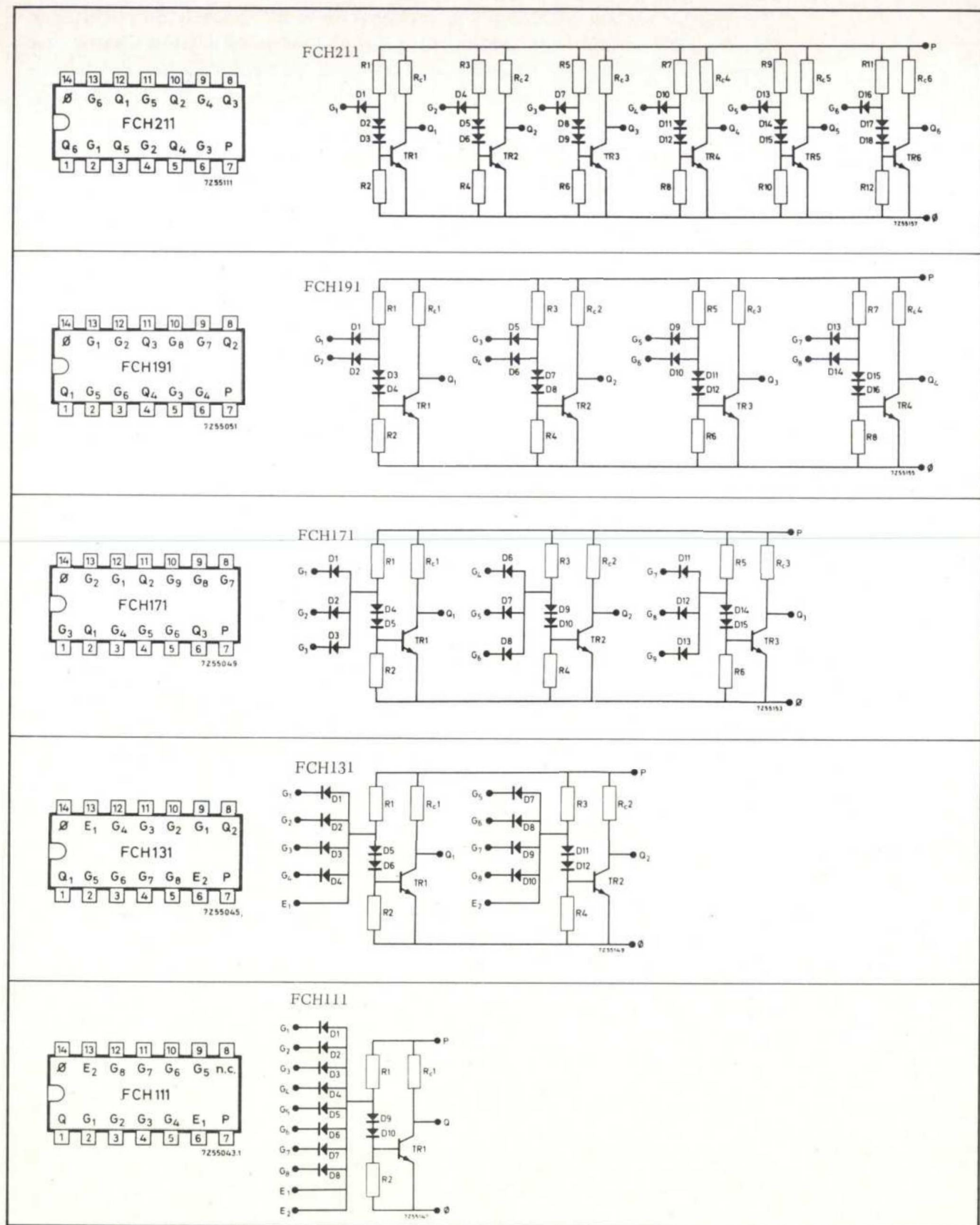
Det gäller därför för fabrikanten att integrera så många och så generellt användbara funktioner som är möjligt i kapslar med 14 eller 16 anslutningar.

I figur 1.10 visas några kretsar ur DTL-familjen:

De minsta kapslarna har 14 ben varav två måste reserveras för matningsspänning (V_{CC}) och jord (GND). Det återstår alltså 12 ben och vi kan därmed låta en kapsel innehålla något av följande alternativ, figur 1.10:

- 6 st inverterare (hex inverter)
- 4 st 2-ingångars NAND-grindar (quadruple 2-input NAND-gate)
- 3 st 3-ingångars NAND-grindar (triple 3-input NAND-gate)
- 2 st 4-ingångars NAND-grindar (dual 4-input NAND-gate)
- 1 st 8-ingångars NAND-grind (8-input NAND-gate)

I två av kapslarna utnyttjas inte alla ben.



Figur 1.10
Många grindar i samma kapsel (Philips).

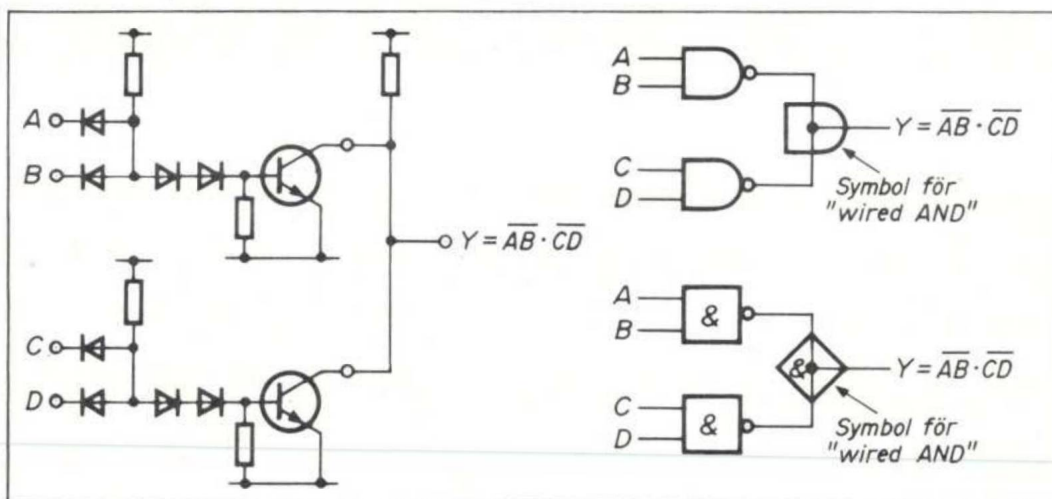
2.3.2 Grindar med speciella funktioner

Vi skall nedan bekanta oss med några vanliga kombinationskretsar, från enkla grindar till några exempel på mer komplexa funktionsenheter.

2.3.2.1 Trådade grindar

Ett enkelt sätt att utöka funktionen hos två NAND-grindar är att koppla ihop eller "tråda" deras utgångar, figur 1.11.

Data till TTL
kretsar finns på
DIL
Just in line i
katalogblad.



Figur 1.11
Trådning av NAND-grindar.

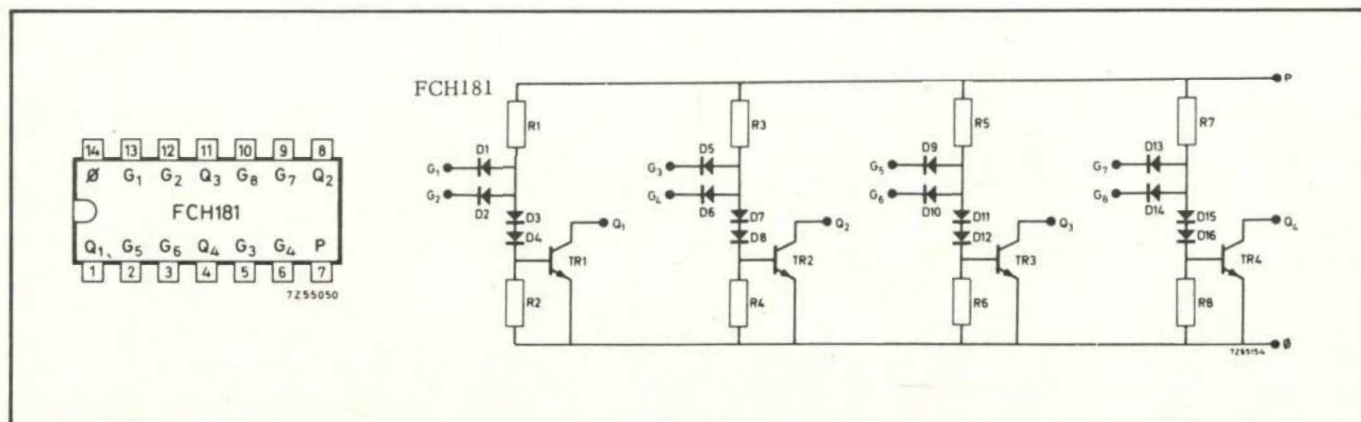
Man får därvid (utan någon extra grind) en ny funktion. Varje NAND-grind i figur 1.11 har samma funktion på den gemensamma utgången som de enskilda transistorerna i NOR-grinden i figur 1.8b. Funktionen blir:

$$Y = \overline{AB + CD} = \overline{AB} \cdot \overline{CD}$$

OR AND

och kopplingen går därför ofta under namnet "wired-OR" eller "wired-AND".

Om många NAND-grindar skall trådas bör man använda en gemensam yttre kollektor-resistor. Fabrikanterna brukar därför tillverka en serie grindar (motsvarande figur 1.10 b - e) utan integrerade kollektorresistorer, figur 1.12. Dessa grindar sägs ha öppen kollektor (open collector).

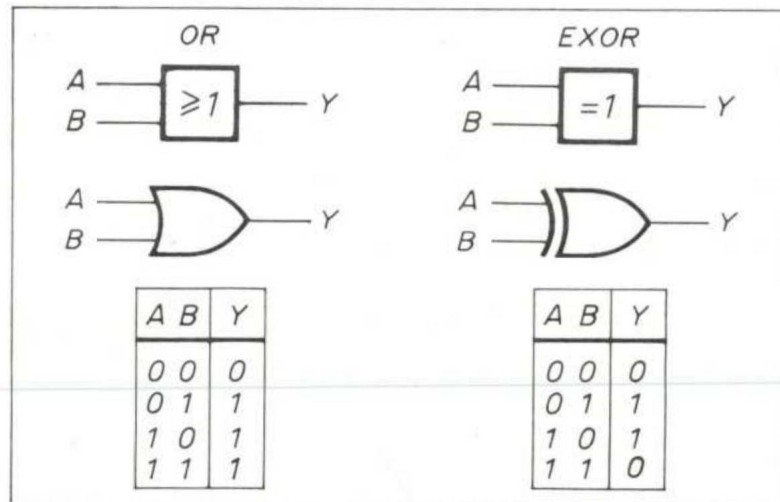


Figur 1.12
NAND-grindar avsedda för trådning, dvs utan integrerade kollektorresistorer.

Vill man jämföra två signaler kan man använda en komparator (comparator) eller jämförare. Den avger utsignal, exempelvis H, om insignalerna är lika och en annan utsignal, exempelvis L, om insignalerna är olika.

2.3.2.2 EXOR-kretsen

EXOR-kretsen, kallas också "exclusive OR" eller "XOR", kan användas för detta ändamål. I figur 1.13 jämförs den vanliga OR-funktionen, som ibland kallas "inclusive OR" eftersom den inkluderar fallet med ett or på båda ingångarna, och EXOR-funktionen.



EXOR

A — $\boxed{=1}$ — X

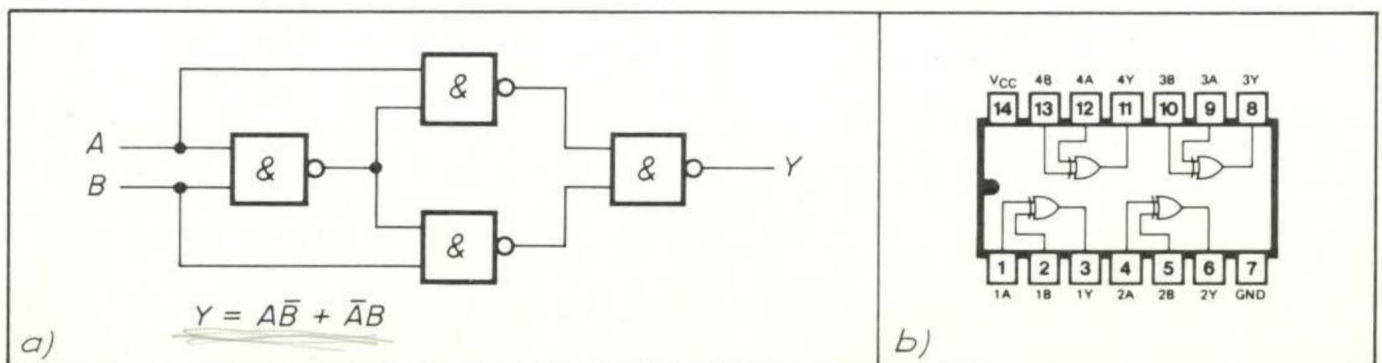
$X = A \oplus B =$
 $= A\bar{B} + \bar{A}B$

Figur 1.13
OR- och EXOR-funktionerna.

EXOR-kretsen har många användningsområden och förpackas därför i kapsel på liknande sätt som 2-ingångars NAND-grindar i figur 1.10b.

Den inre uppbyggnaden av en EXOR-krets kan göras på olika sätt. Figur 1.14 visar hur fyra NAND-kretsar kan bilda en EXOR-funktion som kan tecknas:

$$Y = A\bar{B} + \bar{A}B$$

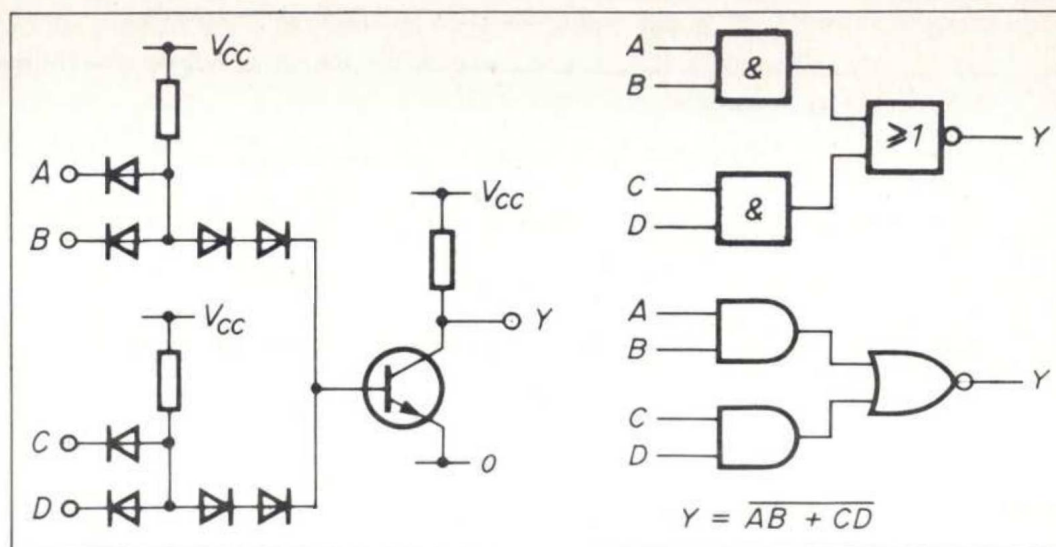


Figur 1.14

- a) EXOR-funktion uppbyggd av fyra NAND-grindar.
b) Kapsel innehållande fyra EXOR-kretsar (7486).

2.3.2.3 AOI-kretsen

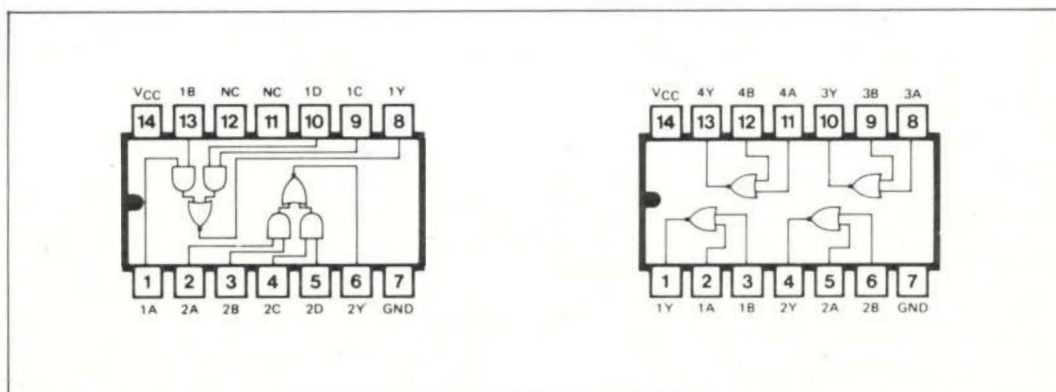
Kopplar man in två diodgrindar till samma inverterartransistor på det sätt som visas i figur 1.15 får man en krets som brukar kallas AOI-krets (AND OR INVERT). Den är utomordentligt användbar vid realisering av mer komplexa logiska funktioner.



Figur 1.15
AOI-kretsen och dess logiska funktion (positiv logik).
and or invert

Om man enbart utnyttjar en av diodingångarna på varje grind får man en NOR-krets. Även NOR-kretsen kan alltså enkelt realiseras med hjälp av vår typgrind.

Figur 1.16 visar exempel på kapslar innehållande AOI- och NOR-kretsar ur TTL-familjen.



Figur 1.16
AOI- och NOR-
kretsar (7451
och 7402).

2.3.2.4 Adderaren

Vid addering av två binära siffror gäller - precis som vid decimal addering - bestämda räkneregler, figur 1.17.

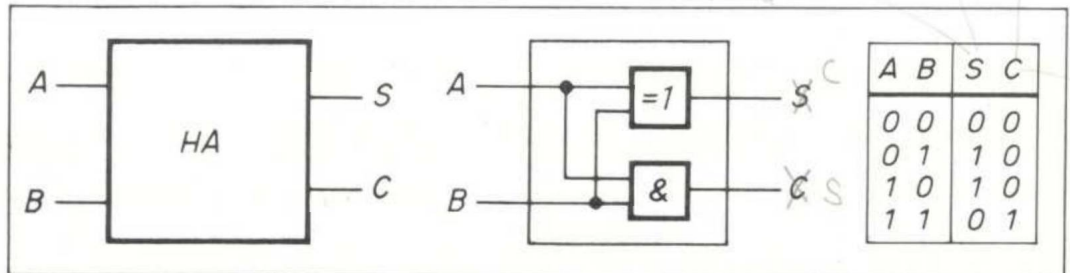
augend	addend	falsk summa (sum)	överföringssiffra (carry)
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Figur 1.17
Räkneregler för
binär addition.

Eftersom reglerna specificerar två funktioner, en summa (S) och en carry (C), kan vi med enkla grindnät bygga ihop motsvarande kombinationskrets, figur 1.18.

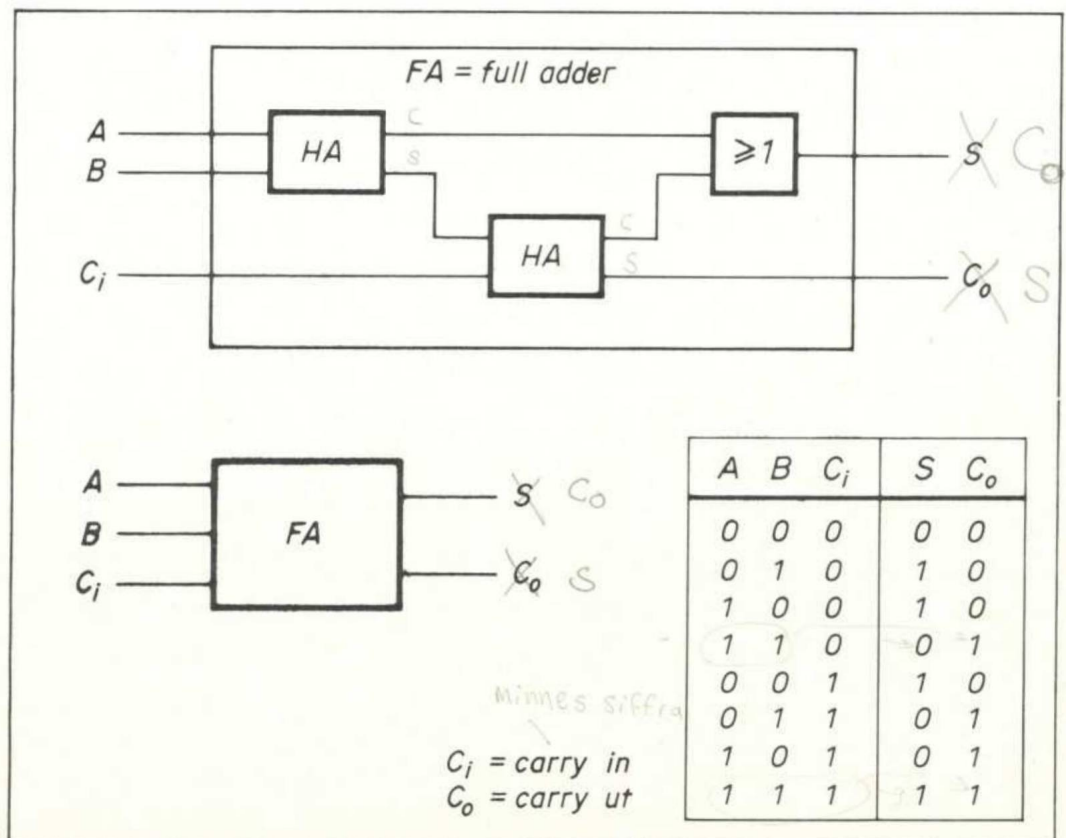
I figur 1.13 och figur 1.2 har vi tidigare mött funktionerna S och C men då kallade EXOR och AND. Adderaren i figur 1.18 kan alltså enkelt byggas upp av en EXOR- och en AND-krets och de logiska funktionerna kan tecknas:

$$\begin{cases} S = A\bar{B} + \bar{A}B \\ C = AB \end{cases}$$



Figur 1.18
Halvadderaren.

Nu har man vid addering normalt att ta hänsyn till en minnessiffra även på ingången. Det klarar inte kretsen i figur 1.18 och den kallas därför halvadderare (HA, half adder). Om två halvadderare kopplas samman så som visas i figur 1.19 får vi en krets som även klarar av en minnessiffra C_i på ingången. Kretsen i figur 1.19 kallas heladderare (FA, full adder).



Figur 1.19
Heladderare uppbyggd av två halvadderare och en OR-grind.

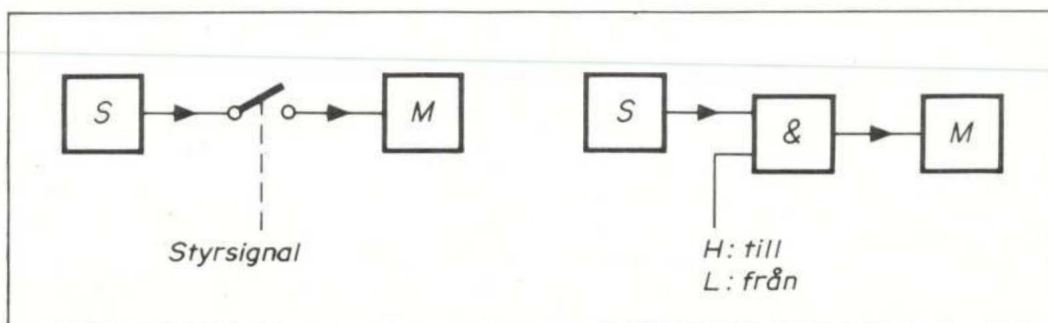
2.4 Exempel på användning av grindar

Vi har nu sett hur man kan beskriva olika grundfunktioner med logisk algebra och hur man kan bygga upp olika grindfunktioner med hjälp av typgrindar i olika varianter eller kombinationer. För att visa var dessa byggblock kommer in i digitala system skall vi nu se grindarnas funktion för lösning av en rad praktiska uppgifter.

2.4.1 En signalgrind

Ofta skall data flyttas över från ett register till ett annat. Detta kan ske med hjälp av grindar som på en given styr- (eller kontroll-) signal öppnar en signalväg från det ena registrets utgångar till det andras ingångar. Figur 1.20 visar ett exempel där en AND-grind fungerar som signalgrind mellan en sändare S och en mottagare M. Det är just tillämpningar av denna typ som givit upphov till användningen av ordet "grind" (gate) för enkla kombinationskretsar.

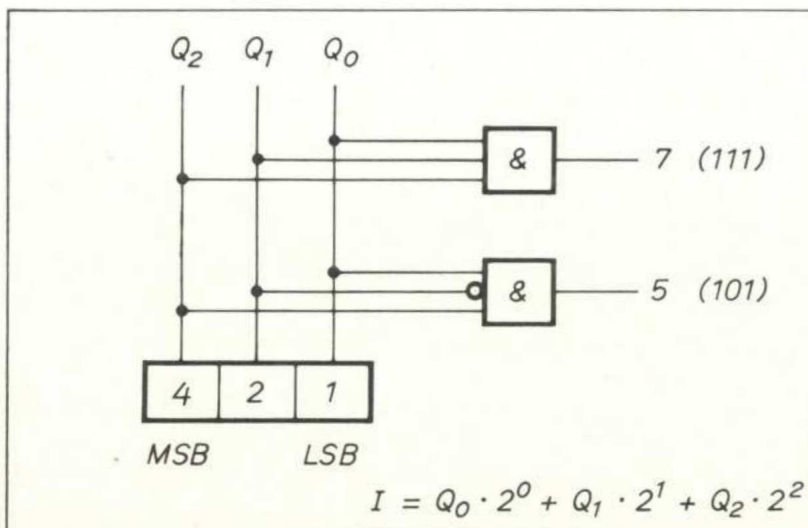
Lägg märke till att grinden i figur 1.20 enbart fungerar i en riktning till skillnad från den symboliska switchen till vänster i figuren.



Figur 1.20
Signalgrind med AND-krets.

2.4.2 En avkodare

Ett antal vippor bildar ett register. Registrets innehåll kan avläsas på vippornas ställning. Vi kan "avkoda" ett önskat värde med hjälp av en AND-grind, figur 1.21.



Figur 1.21
Avkodare med AND-grindar.

Antal kombinationer
 2^n

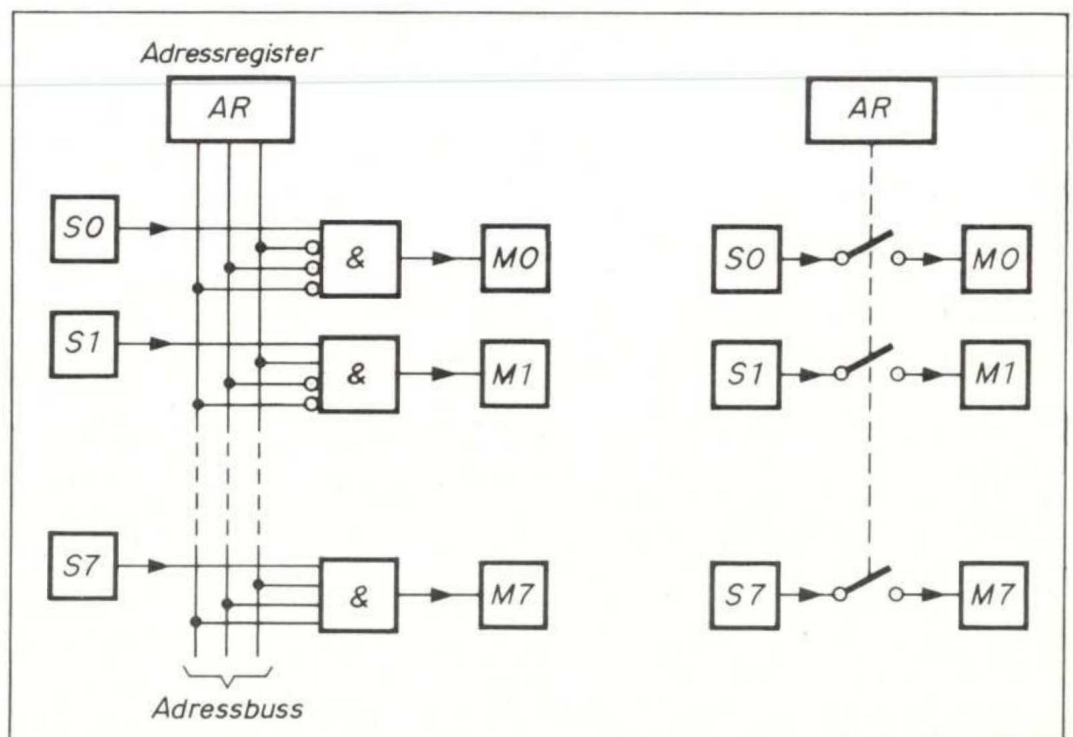
Vippornas utgångar är betecknade Q_0 , Q_1 och Q_2 där Q_0 avser den minst signifikanta siffran (LSB = least significant bit) och Q_2 den mest signifikanta (MSB). Att utgångarna vanligen indicerats 0 - 2, och inte 1 - 3, beror på att registrets innehåll (I) enkelt kan tecknas:

$$I = Q_2 \cdot 2^2 + Q_1 \cdot 2^1 + Q_0 \cdot 2^0$$

Kopplar vi in en AND-grind till samtliga utgångar avkodar vi det innehåll som motsvarar innehållet 111 binärt, dvs 7. Läger vi inverterare på vissa av AND-grindens ingångar kan godtyckliga värden mellan 0 och 7 avkodas. Inverterarna kan markeras med ringar på AND-symbolens ingångar så som framgår av figur 1.21.

2.4.3 Buss med avkodare

Om det är många signalgrindar i ett digitalt system och om dessa grindar skall öppnas vid olika tillfällen kan man styra dem från ett antal parallella ledningar som kallas buss (bus). Figur 1.22 visar ett exempel.



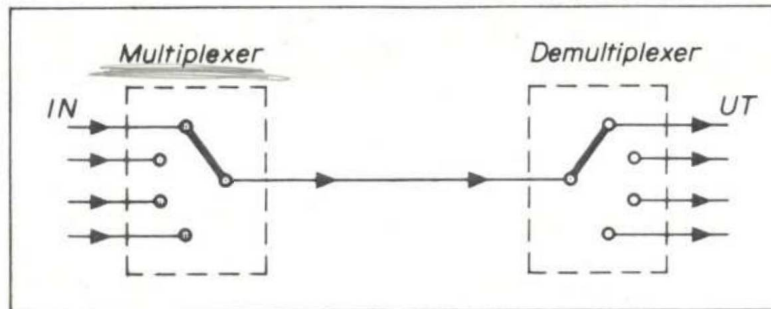
Figur 1.22
Styrning av signalgrindar via adressbuss.

Registret AR är här ett adressregister och dess innehåll ger en adress till den grind som skall öppnas. Med ett trebitars adressregister kan vi tydligen styra $2^3 = 8$ grindar. (Med en 8 ledningars adressbuss kan man adressera $2^8 = 256$ grindar och det är ju bra mycket enklare än att ha 256 olika styrledningar att hålla reda på!)

I figur 1.22 används tre av AND-grindens ingångar för adressering medan den fjärde är signalingång.

2.4.4 Multiplexer och demultiplexer

En multiplexer, eller MUX som det ibland förkortas, och en demultiplexer är digitala omkopplare. De kallas också dataväljare (data selector) och datafördelare. Eftersom signalen enbart kan passera grindar i en riktning har den digitala omkopplaren två namn beroende på signalriktningen, figur 1.23.

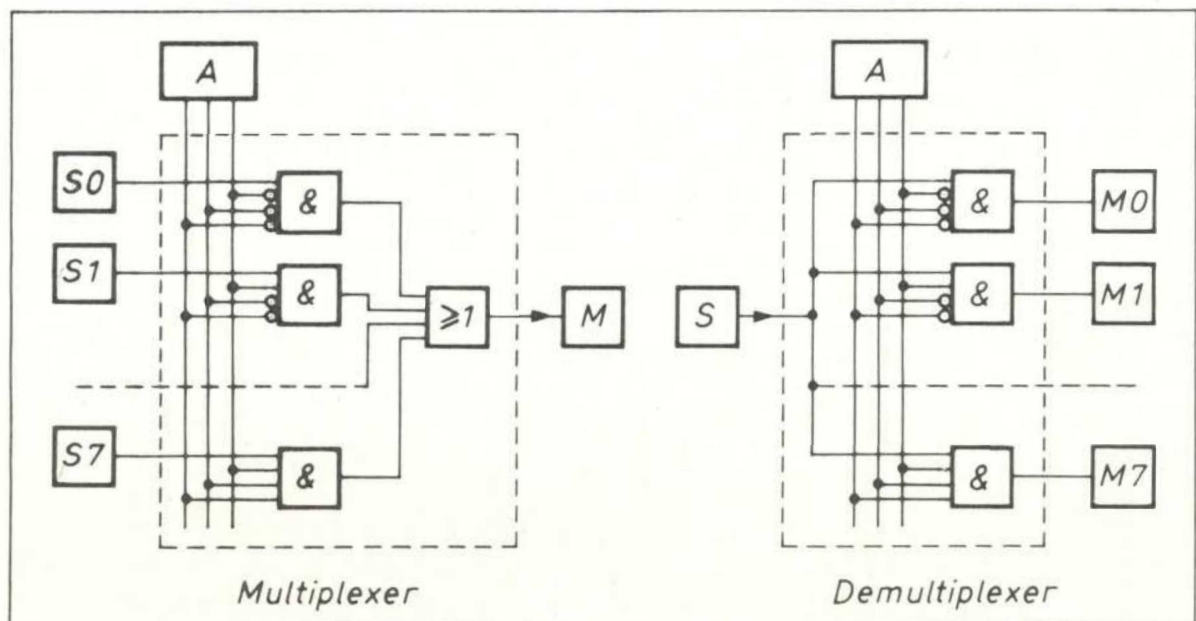


Figur 1.23

Principen för en MUX. *= Multiplexer*

Multiplexern har många användningsområden. Genom att köra multiplexern och demultiplexern synkront kan flera signaler överföras på samma ledning - man kallar detta tidsmultiplex. Men detta är bara en av många användbara tillämpningar.

Multiplexern och demultiplexern har en uppbyggnad som mycket liknar de adressbusstyrda signalgrindarna i figur 1.22. Enda skillnaden är att i multiplexern är det enbart en mottagare och i demultiplexern enbart en sändare. Figur 1.24.



Figur 1.24

Multiplexer och demultiplexer uppkopplade av AND- och OR-grindar.

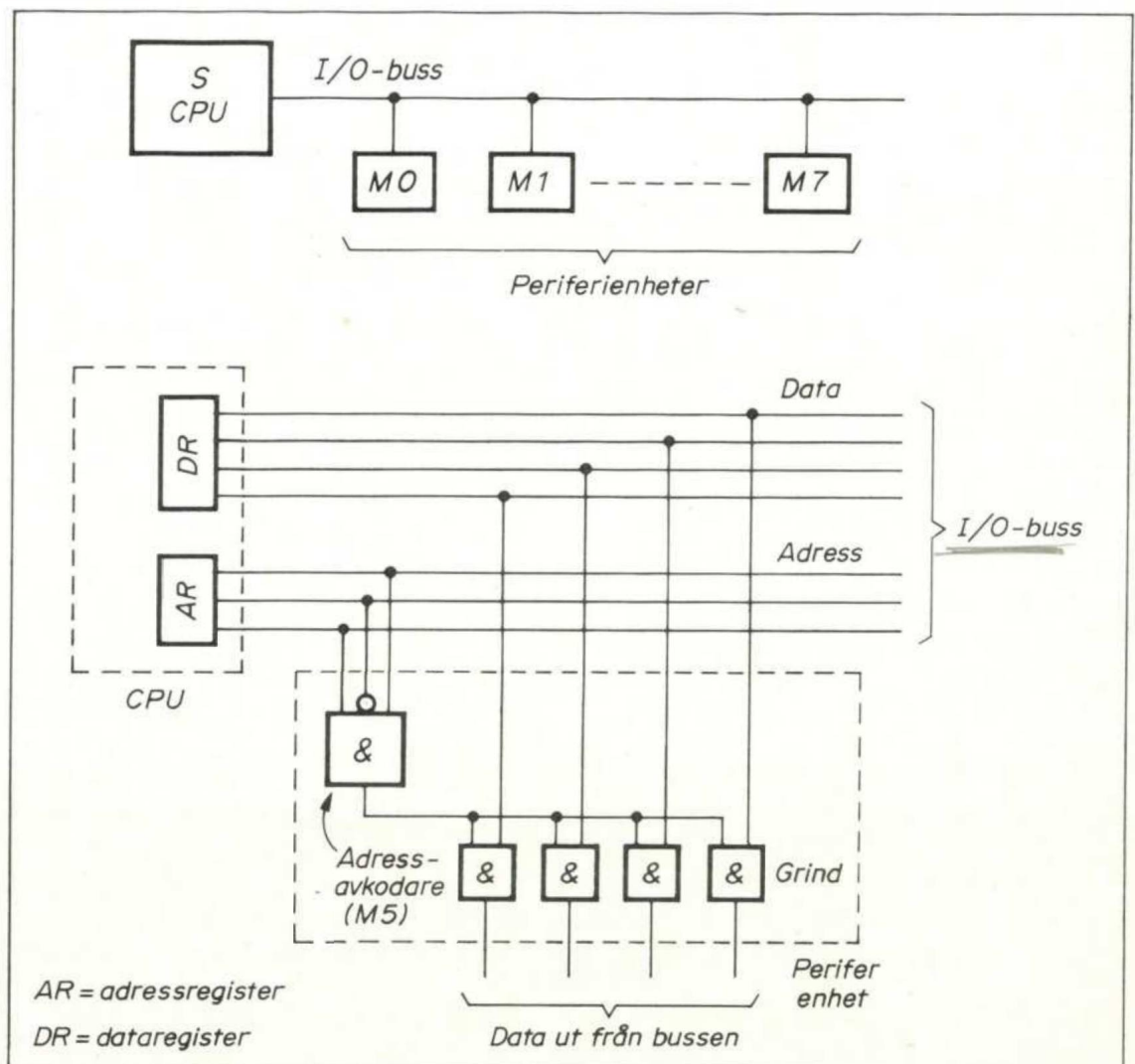
Observera att vi i multiplexern kan använda kombinationen AND-OR. AOI kretsar är alltså användbara i dessa sammanhang (inverteringen är ej något problem).

Multiplexrar och demultiplexrar finns att köpa som färdiga kretsar och man behöver som regel inte bygga samman dem av grundkretsar som i figur 1.24.

2.4.5 En I/O-buss

I ett datorsystem står processorn (CPU, central processing unit) normalt i kontakt med de yttre enheterna (exempelvis bandstation, teletype etc) via en I/O-buss (input/output bus). Figur 1.25a visar ett exempel på ett blockschema där sändaren S är en processor och mottagaren ($M_0 - M_7$) olika periferienheter.

När CPU:n vill sända data till en viss periferienhet lägger den ut adressen (via adressregistret AR) på I/O-bussen. Därvid kopplas den önskade periferienheten in via sin adressavkodare. När CPU:n nu lägger ut data på bussen (via dataregistret DR) kommer dessa att överföras till periferienheten.



Figur 1.25
Grindar för avkodning och avläsning av I/O-buss.

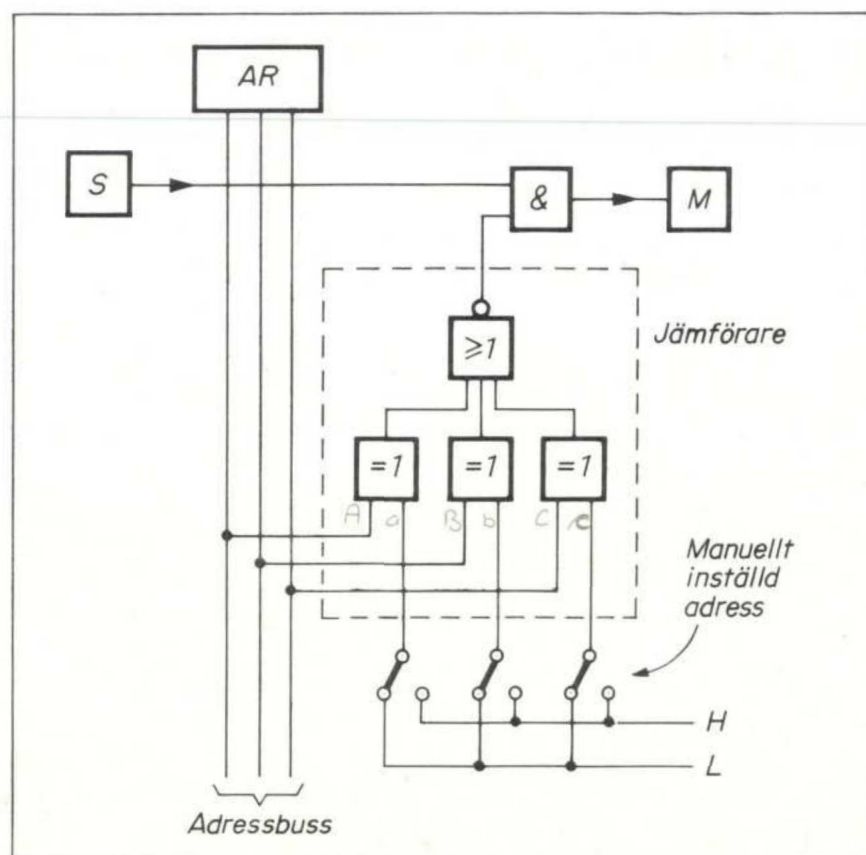
Normalt ingår en tredje kategori ledningar i I/O-bussen. Dessa kallas styrledningar (control lines). Med styrledningarna ges kommando exempelvis om läsning eller skrivning. Vi har utelämnat styrledningarna i figur 1.25 och enbart visat en av I/O-bussens många funktioner nämligen överföring av data från CPU:n till en periferienhet.

2.4.6 En jämförare

Vi skall avsluta med ett exempel på användningen av en jämförare. Antag att vi har en signalgrind vars adress vi vill kunna ställa in för hand, bi-närt, genom att ställa tre switchar.

I figur 1.26 sitter en jämförare (comparator) som jämför adressbussens ställning med switcharnas. När båda är lika öppnas signalswitchen av jämföraren.

Jämföraren i figur 1.26 är uppbyggd av EXOR-kretsar samt en NOR-grind.



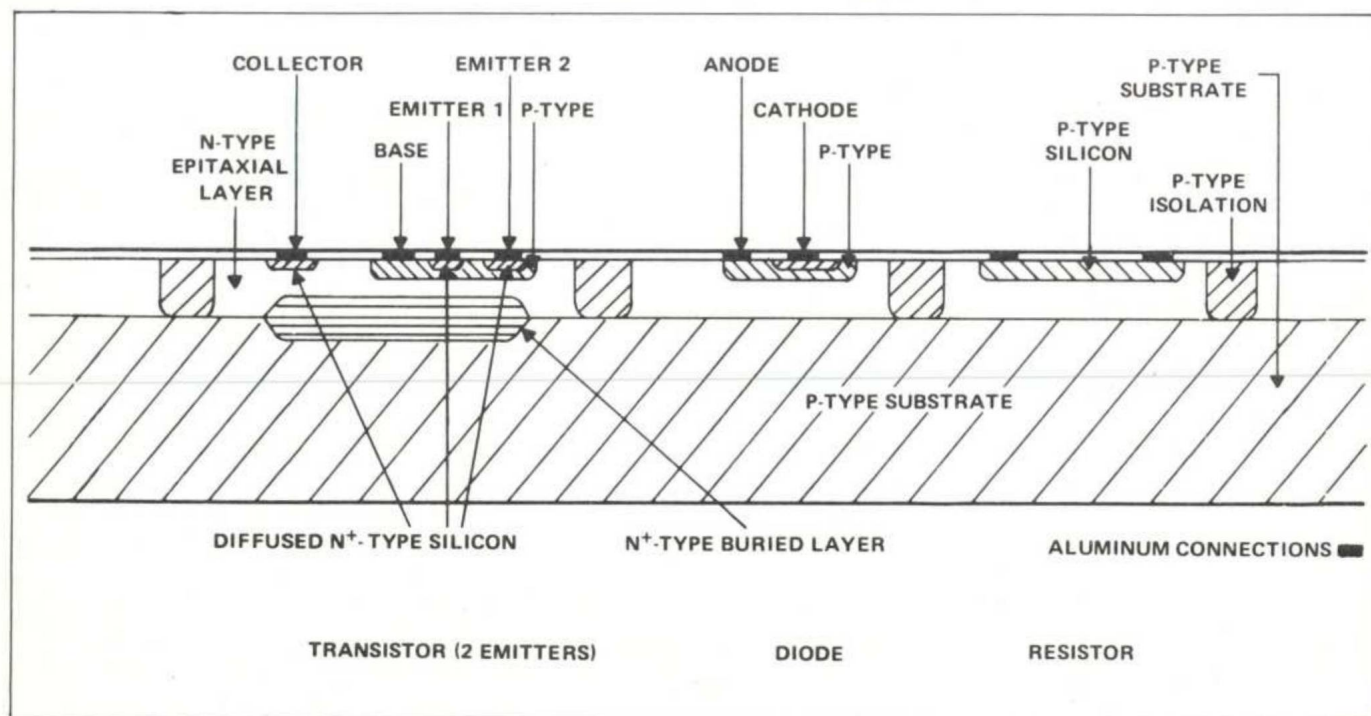
Figur 1.26
Jämförare använd vid adressavkodning.

3 Planarteknik

Det är utvecklingen av kiseltransistorer och mikrokretsar under slutet av 50-talet och början av 60-talet som lagt grunden till den moderna digitaltekniken. Den viktigaste milstolpen i denna utveckling är planarprocessen, dvs den teknik med vars hjälp man kan integrera transistorer, dioder och resistorer i en och samma monokristall.

Figur 1.27 visar i en principskiss hur dessa element sitter gömda under oxiden i kiselkristallen. Elementen är isolerade från varandra genom områden av P-typ (P-type isolation). För att resistansen mellan kollektoranslutningen och transistorns kollektorskikt skall bli mera lågohmig har kollektorn en högdopad, dvs väl ledande, kollektorbotten (N^+ -type buried layer). Samtliga element är uppbyggda i ett tunt kristallskikt av N-typ som pålagts kristallen med s k epitaxi (N-type epitaxial layer).

Som frivillig merläsning skall vi nedan gå igenom momenten vid tillverkningen av en enkel NOR-grind av RTL-typ.



Figur 1.27

Genomsnittsskärning av kiselkristall som innehåller en transistor (med två emitterar) en diod och en resistor (Texas Instruments).

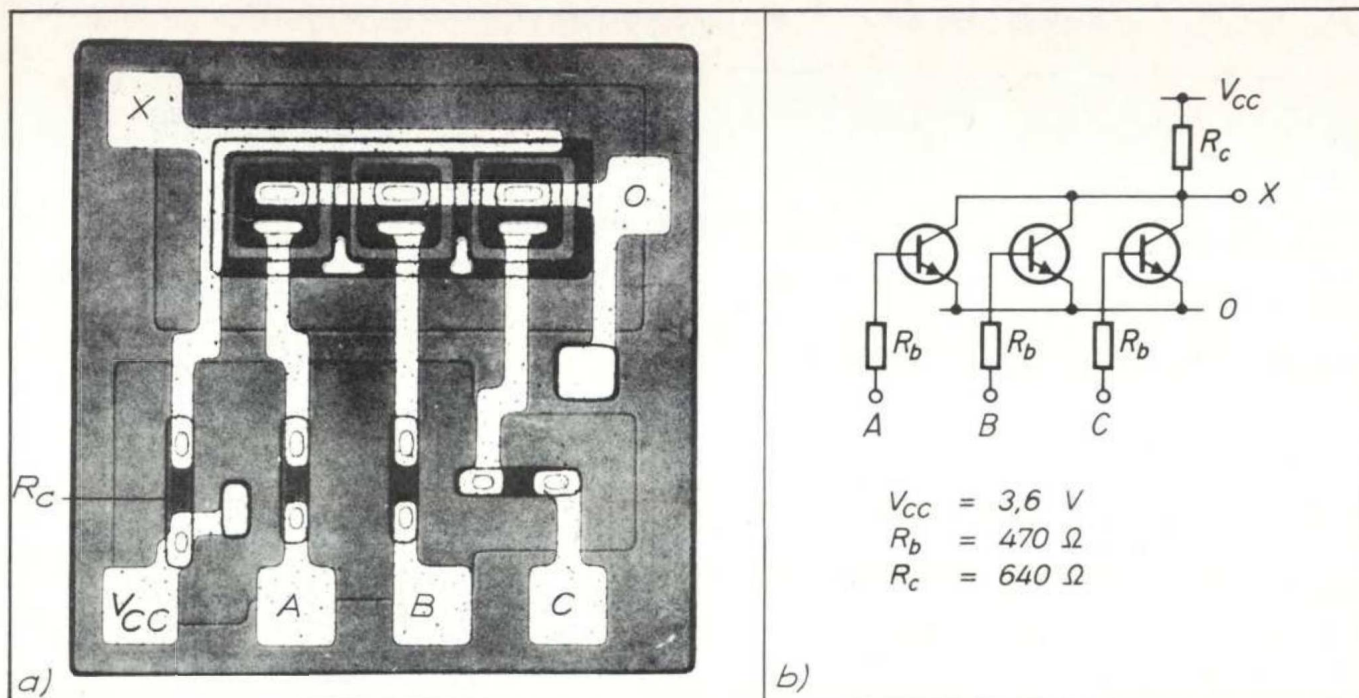
3.1 Typgrinden i RTL-familjen

I figur 1.28 ser vi en NOR-grind av samma typ som den vi tidigare mött i figur 1.8b, men nu med tre transistorer. Kristallfotot visar tydligt de tre transistorerna överst i en rad med tillhörande basresistorer samt kollektorresistorn nederst till vänster. Transistorerna ligger i en N-ledande "ö" som är isolerad från omgivningen av ett omgivande P-område, isolationsdiffusionen.

Bas- och kollektorresistorer ligger i en annan N-ledande och isolerad "ö" som via ett kontakthål är ansluten till den positiva matningsspänningen V_{CC} (3,6 V). Isolationsdiffusionen och transistorernas emitterar är jordade.

Ledarmönstret syns tydligt liksom även oxidhålen genom vilka kontaktningen till underliggande kristall sker.

Slöseriet med kristallyta är här påfallande, men det är å andra sidan lättare att bekanta sig med kretselementen i en äldre logikkrets där detaljerna ligger väl åtskilda och är tydligt synliga.



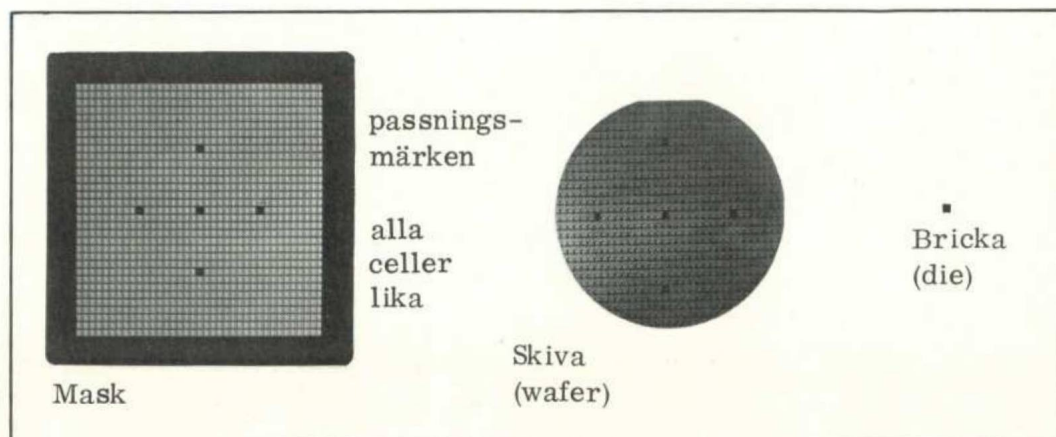
Figur 1.28
 Integrerad RTL-krets (Motorola). a) Kristallstruktur b) Kretsschema

3.2 Tillverkningen av en RTL-krets

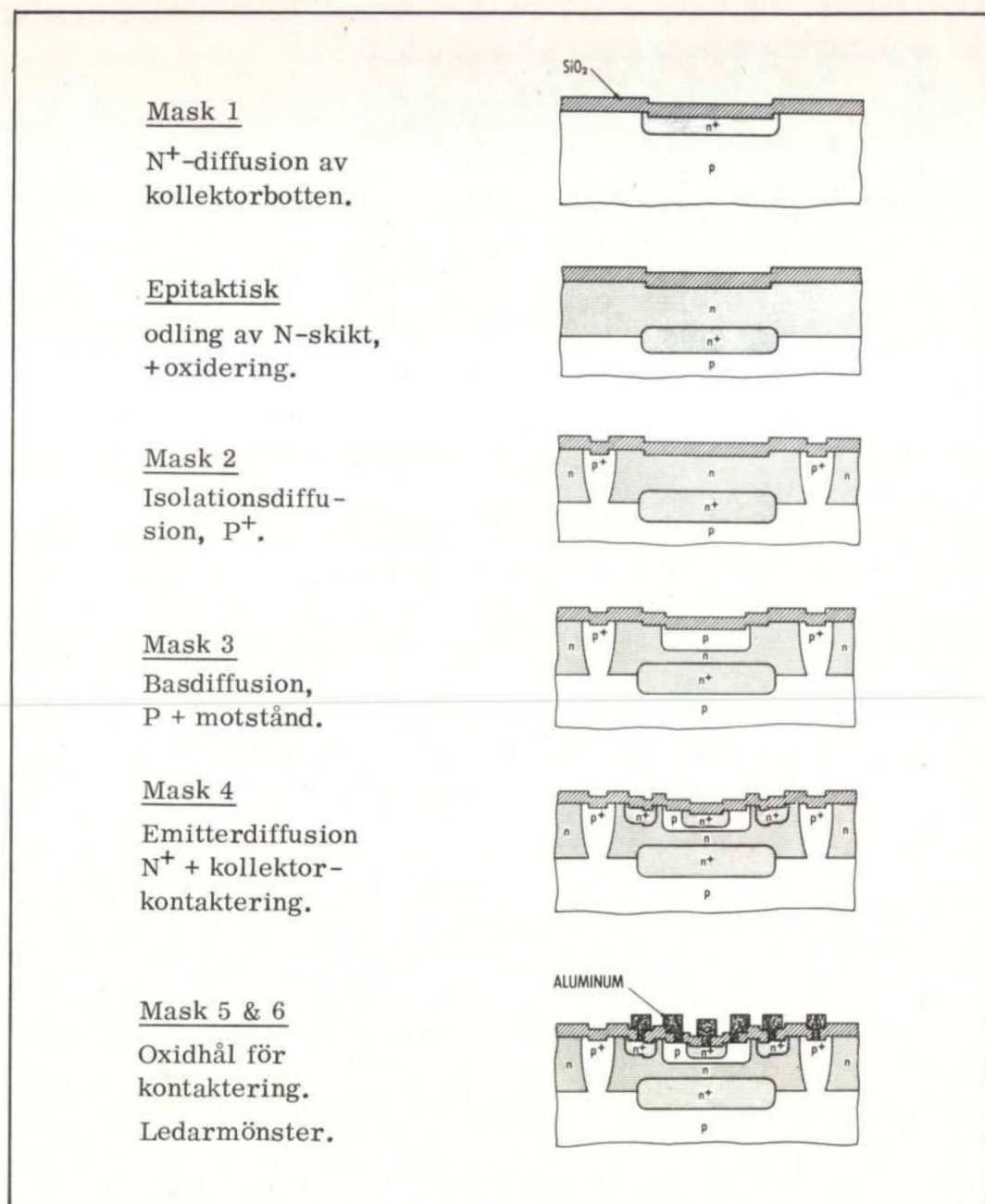
Halvledarteknologin har utvecklats avsevärt under senare hälften av 1960-talet och början av 1970-talet. Framförallt har förbättrade fotolitografiska metoder möjliggjort tillverkning av element med avsevärt mindre dimensioner och därmed högre packningstäthet.

Väsentligt för priset på en IC-krets är kristallskivans (wafer) storlek vid tillverkningen. Vid 1960-talets mitt kunde man processa kristallskivor med ca 2,5 cm diameter (1") medan dagens teknologi tillåter skivor med en diameter bortåt 10 cm (3"). En enda sådan skiva kan innehålla tusentals integrerade kretsar. På grund av ofrånkomliga kristallfel måste vissa kretsar kasseras och man talar här om processens "yield". De senaste årens låga priser på integrerade kretsar visar att yelden har ökat väsentligt.

En faktor som påverkar yelden och därmed priset är det antal steg, diffusioner och masker, som fordras för processen. För den normala bipolära processen erfordras sex masker och fyra diffusioner. Figur 1.29 visar en av de sex maskerna för bipolarprocessen samt en skiva och en bricka.



Figur 1.29
 En av maskerna för planarprocessen, kiselskivan samt en utskuren bricka. (Motorola).



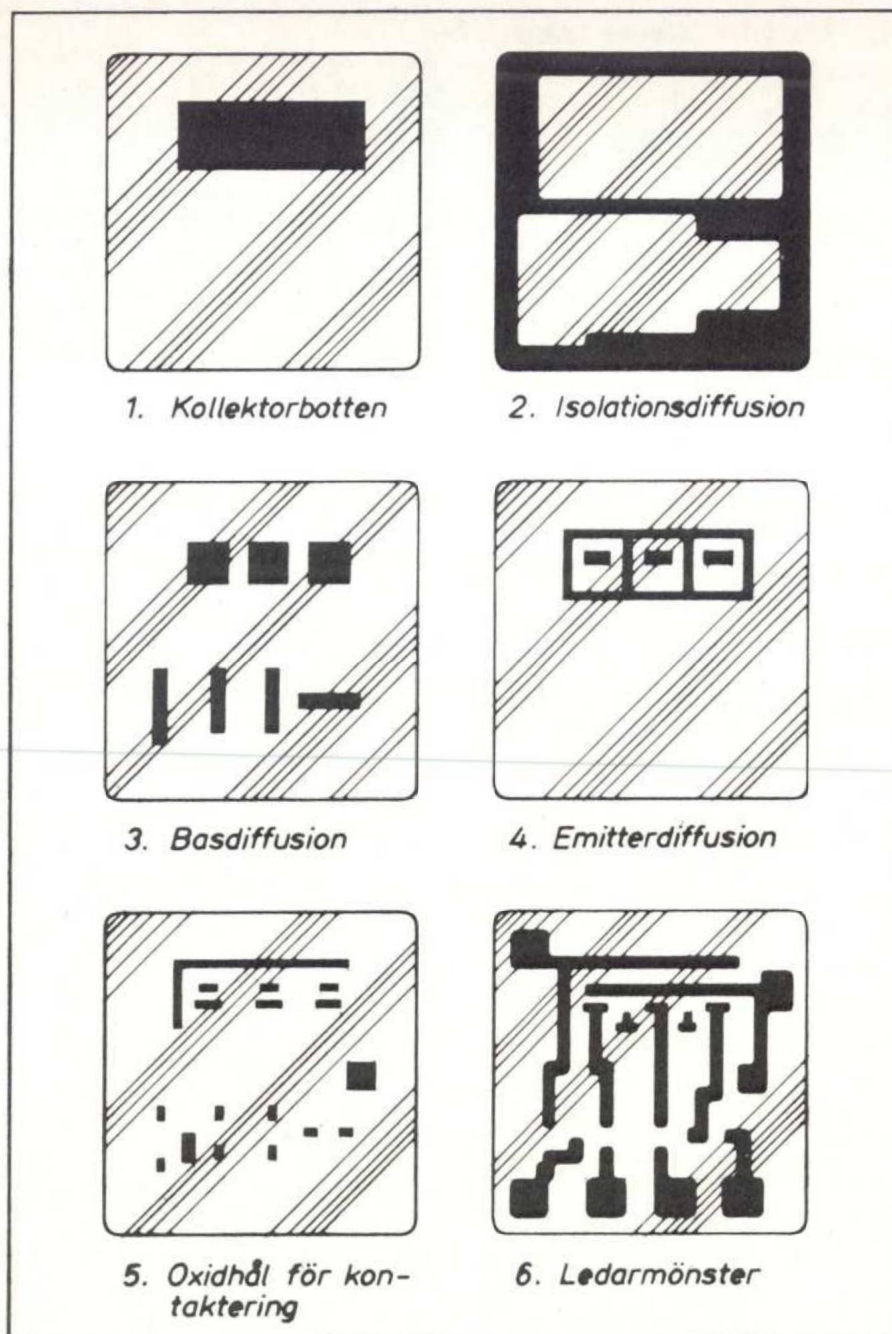
Figur 1.30
Stegen i en bipolarprocess. (Motorola).

Figur 1.30 visar resultatet av de olika stegen i processen. Figuren visar ett snitt genom ett kristallområde i vilket en transistor successivt växer fram.

Figur 1.31 visar en uppförstoring av en cell från var och en av de sex maskerna vid tillverkningen av RTL-kretsen i figur 1.28.

IC-kretsar av MOS-typ kan framställas med färre antal diffusionen och masker. MOS-kretsarnas teknologiska uppbyggnad kommer vi att behandla i ett senare kapitel.

Den ovan, som kristallexempel valda RTL-kretsen tillverkas fortfarande men sedan man utvecklat planarprocessen och numera ej längre är bunden vid enkla kretslösningar har andra logikfamiljer blivit helt dominerande.

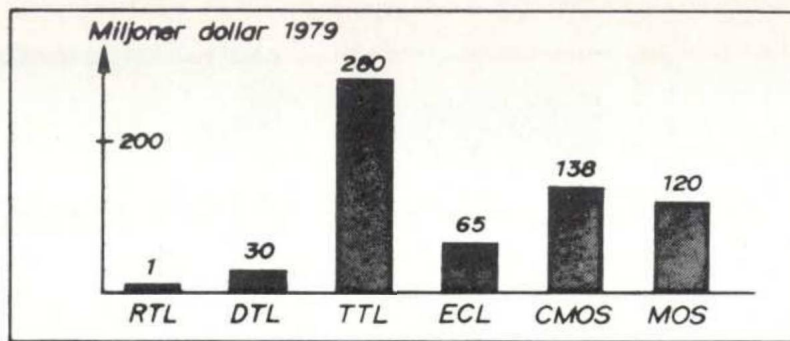


Figur 1.31

Cellmönstret i var och en av de sex maskerna för tillverkning av RTL-kretsen i figur 1.28. (/// = genomskinligt).

4 Logikfamiljernas mångfald

Det finns en mångfald olika sätt att integrera logikgrindar och det är alltså inte så konstigt att det vuxit fram ett otal olika logikfamiljer. Man har optimerat kretsarna med avseende på snabbhet, störkänslighet, effekt-konsumtion, belastbarhet, logiskt sving m m. Om man optimerar en krets i ett av dessa avseenden får man normalt betala med sämre data i andra. Optimeras exempelvis en krets för att bli snabb blir den samtidigt både störkänslig och drar stor effekt. Konstruktörerna talar här om "trade offs", dvs man får betala fördelar i vissa avseende med nackdelar i andra.



Figur 1.32

Uppskattad försäljning 1979 av digitalkretsar i USA.

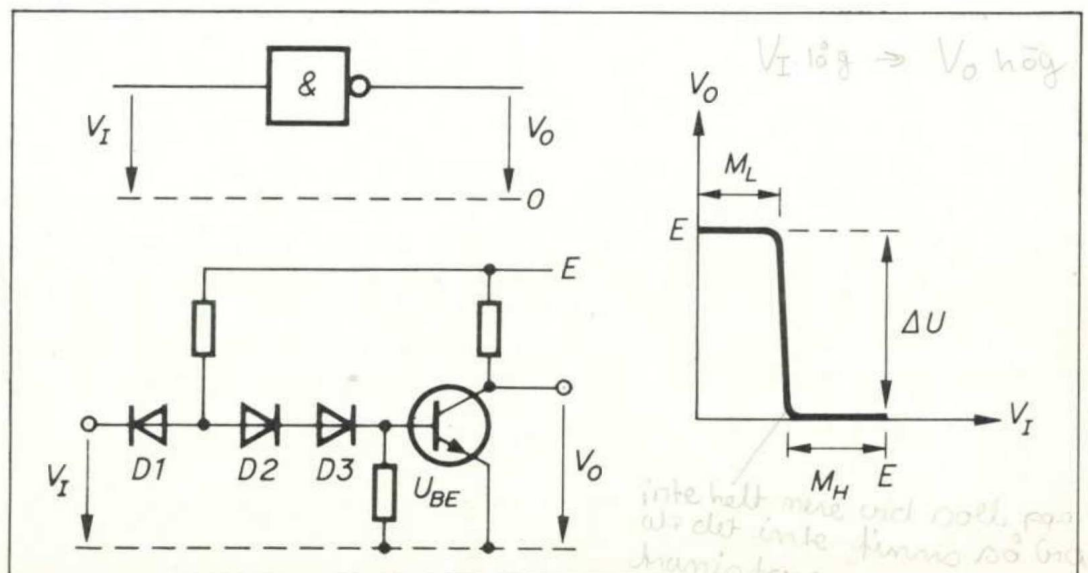
Vi kommer huvudsakligen att ägna oss åt två logikfamiljer, TTL och CMOS. Dessa två familjer delar på större delen av marknaden. Se figur 1.32.

Vi har i detta kapitel redan berört de två logikfamiljerna RTL och DTL. RTL var den enklaste att integrera och därför kom den först ut på marknaden. DTL var en logik som ingenjörerna var vana vid från de tidigare diskret uppbyggda digitalkretsarna, och därför blev DTL snabbt populär när den lanserades i början av 1960-talet.

Vi skall nedan kort beskriva två andra familjer, HNL och ECL. Den ena är optimerad med avseende på störsäkerhet och den andra med avseende på snabbhet. Först måste vi emellertid klargöra ett par begrepp, nämligen "sving" och "störmarginal".

4.1 Sving och störmarginaler

Figur 1.33 visar överföringsdiagrammet mellan utspänningen V_O ($O =$ output) och inspänningen V_I ($I =$ input) för en DTL-grind. När inspänningen är $V_I = 0$ är transistorn strypt och därmed är utspänningen $V_O = E$ (om utgången är obelastad).



Figur 1.33

Sving (ΔU) och störmarginaler (M_L och M_H) hos en DTL-grind.

För att transistorn skall kunna leda ström måste U_{BE} överskrida framspänningen 0,7 V. Spänningen i punkten mellan D1 och D2, figur 1.33, uppgår då till $3 \times 0,7 \text{ V} = 2,1 \text{ V}$ på grund av framspänningsfallen över D2 och D3. Transistorn blir tydligen ledande först då inspänningen passerar värdet

$$V_I \approx 2 \times 0,7 = 1,4 \text{ V}$$

Vid låg ingång kan tydligen inspänningen variera 1,4 V utan att utspänningen ändras. Man säger därför att man har en störmarginal vid låg ingång av storleken

$$M_L \approx 1,4 \text{ V}$$

Motsvarande störmarginal vid hög inspänning blir (vid $E = 5 \text{ V}$) cirka

$$M_H \approx 3,4 \text{ V}$$

Störmarginaler (M_L och M_H) och svinget ΔU vid omslag hos utspänningen är viktiga egenskaper hos en logikgrind. Olika logikfamiljer skiljer sig väsentligt i fråga om sving och störmarginaler. Vi skall nedan visa två logikfamiljer som förhåller sig helt olika i dessa avseenden.

4.2 HNL – en störsäker logik

frekvensen 20-30 MHz

I industriell miljö har man ofta elektriska störningar från kontaktorer och motorer. Logiksystem i sådan miljö måste ha stora störmarginaler om de skall kunna arbeta störningsfritt.

För att få stora störmarginaler måste enligt figur 1.33 matningsspänningen höjas (och därmed svinget ΔU). Om sedan omslagsspänningen för-läggs vid

$$V_I \approx \frac{1}{2} E$$

blir störmarginalerna av storleken

$$M_L \approx M_H \approx \frac{1}{2} E$$

Detta är också filosofin bakom HNL-familjen vars typgrind visas i figur 1.34. HNL betyder "high noise immunity logic", men samma familj går under många namn, bl a:

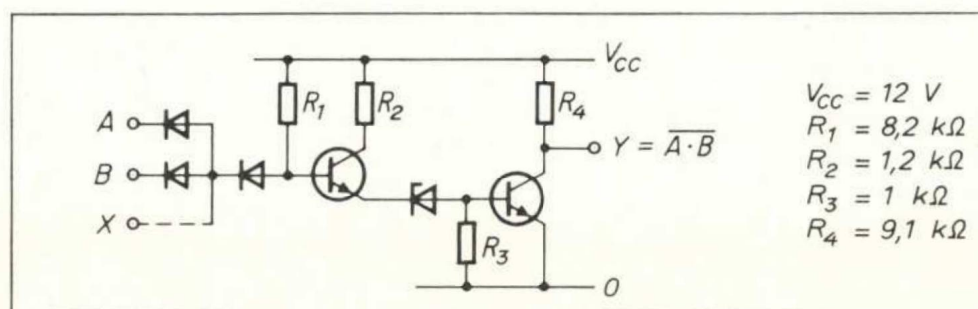
HiNIL (high noise immunity logic)

DTLZ (diode transistor logic with zener diode)

LSL (langsam störsicher Logik)

HLL (high level logic)

På svenska kallas HNL ofta "högnivålogik".



Figur 1.34
HNL-familjens typ-
grind (Teledyne
serie 300).

Typgrinden i figur 1.34 har en emitterföljare mellan diodgrinden och inverterartransistorn. Detta gör att transistorn bottnas ordentligt och därmed tål stor belastning. För att få stora störmarginaler är matningsspänningen $V_{CC} (= E) = 12 \text{ V}$. Det skulle krävas många seriekopplade dioder för att få omslagsspänningen vid $V_I \approx 1/2 E = 6 \text{ V}$.

I typgrinden har man därför ersatt DTL-logikens seriekopplade dioder med en zenerdiod Z , som måste vara vänd åt motsatt håll jämfört med dioderna i DTL-grinden.

Genom att mata HNL-grinden med hög matningsspänning och bottna transistorn hårt får man en logik som är störsäker. Men den drar å andra sidan stor effekt och blir långsam.

4.3 ECL – en snabb logik

ECL (emitter coupled logic) är optimerad för att bli en snabb logik. För att uppnå snabba omslag måste man begränsa storleken på de laddningar som deltar i omslaget. Man undviker därför bottning, som innebär stor laddningsupplagring ($q = C \cdot \Delta U$), och använder små sving.

Figur 1.35 visar typgrinden för logikfamiljen MECL 10.000. Grinden fungerar som ett differentialsteg där vänstra sidan består av två transistorer vars kollektorer är parallellkopplade. Högra sidan i differentialsteget matas med en konstant referensspänning V_{ref} .

Emitterresistorn fungerar i stort sett som en strömgenerator.

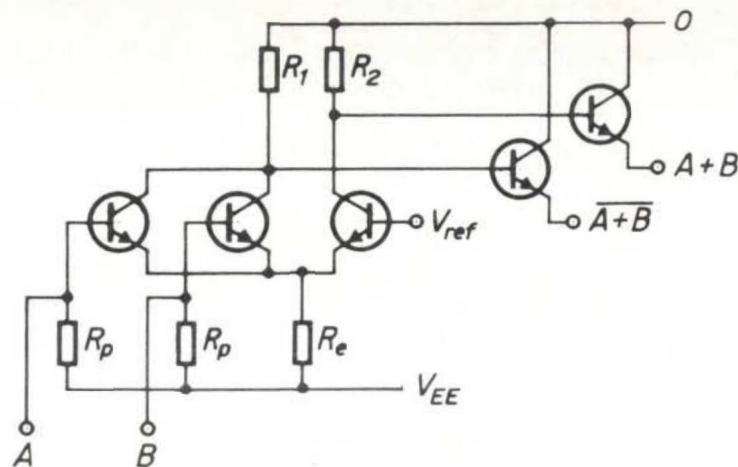
Om inspänningen på någon av de två vänstra transistorerna är hög ($H = -0,9 \text{ V}$) går totala emitterströmmen genom kollektorresistorn R_1 . Om inspänningen på båda dessa transistorer är låg ($L = -1,7 \text{ V}$) går totala emitterströmmen genom R_2 . Kollektorresistorerna R_1 och R_2 är så dimensionerade att transistorerna hela tiden arbetar i aktiva området. Svinget är, som framgår av överföringsdiagrammet i figur 1.32b, av storleken $\Delta U \approx 0,8 \text{ V}$. ECL har därför små störmarginaler.

ECL-logiken är snabb. Den har en omslagstid på ca 2 ns jämfört med 300 ns för en HNL-grind.

ECL-grinden har två utgångar. Den vänstra ger NOR-funktion på samma sätt som RTL medan den högra ger komplementet, dvs en OR-funktion.

ca $f \approx 200 \text{ MHz}$

känsliga för
självvägning



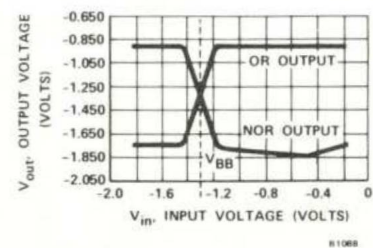
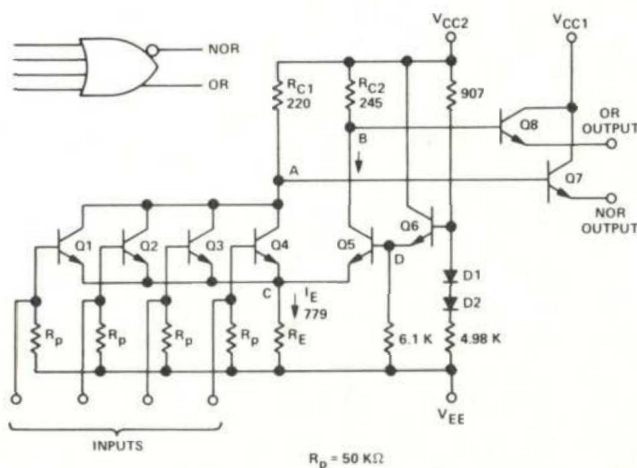
$$\begin{aligned} V_{EE} &= -5,2 \text{ V} \\ V_{ref} &= -1,3 \text{ V} \\ R_p &= 50 \text{ k}\Omega \\ R_e &= 780 \Omega \\ R_1 &= 220 \Omega \\ R_2 &= 245 \Omega \end{aligned}$$

a)

MECL 10,000 - the system designer's logic

MECL 10,000 logic is designed to satisfy the needs of the logic and systems designer who is planning next generation equipment in high speed instrumentation, communication, computer peripherals, and computer mainframes.

MECL 10,000 Basic Gate



b)

Figur 1.35
ECL-grind.

- a) Principen för en ECL-grind
b) En del av databladet för en Motorola-grind

Repetitionsfrågor

1. Rita symboler och sanningstabeller för följande funktioner: NOT, AND, OR, NAND, NOR och EXOR.
2. Vad menas med positiv och negativ logik?
3. Rita typgrindar för en DTL NAND-krets och en RTL NOR-krets (med två ingångar) samt förklara deras funktion.
4. Rita kretsschema (DTL) och logikschema för en AOI-krets.
5. Vilken logisk funktion får man vid trådning av två NAND-kretsar (med vardera två ingångar)?
6. Vad karakteriserar en logikfamilj?
7. Hur kan en halvadder realiseras med en EXOR-grind och en AND-grind?
8. Vad menas med en buss och hur många adresser kan man lägga ut på en buss med fyra ledningar?
9. Skissera kristallstrukturen för en integrerad transistor, diod och resistor.
10. Jämför HNL och ECL med avseende på sving, störmarginaler och snabbhet.

D2 Sekvenskretsar

Sekvenskretsar 30

1 Register, koder, tal 30

- 1.1 Binärkod 31
- 1.2 BCD-koder 32
- 1.3 Parallell- och serieregister 33

2 Vippor 34

- 2.1 Vippornas uppgifter i parallell- och serieregister 34
- 2.2 Asynkron och synkron triggning 36
- 2.3 Klockade RS- och JK-vippor 38
- 2.4 JK-vippan kopplad som räknare, T-vippa eller D-vippa 38
- 2.5 JK-vippans triggning 39
- 2.6 En JKMS-vippa 41
- 2.7 Krav på vippans insignaler 43

3 Skiftregister 44

4 Räknare 46

- 4.1 Rippelräknare 48
- 4.2 Synkronräknare 50
- 4.3 Skiftregister som räknare 53

Repetitionsfrågor 56

Målruta

När du läst detta kapitel skall du kunna:

- skriva decimala tal i binär, oktäl och hexadecimal form samt i BCD-kod
- rita grindscheman för klockade RS-vippor uppbyggda av NAND- och NOR-latchar
- rita sanningstabeller för klockade RS- och JK-vippor samt visa hur en JK-vippa kan kopplas som frekvensdelare, T-vippa och D-vippa
- förklara skillnaden mellan en flanktriggad vippa och en MS-vippa
- rita en $\overline{RS}(H)JK(M)$ -vippa uppbyggd av NAND-kretsar
- skissera ett fem bitars skiftregister kopplat som Johnsonräknare
- förklara skillnaden mellan rippelräknare och synkronräknare

Sekvenskretsar

Enligt Svenska akademins ordlista (1974) betyder "sekvens" ordningsföljd, svit; inom musiken upprepning av motiv. En engelsk-svensk fickordbok ger för ordet "sequence" översättningen "serie, rad, följd". SEK definierar begreppet sekvenskrets på följande sätt: (SEN 01 1603)

sekvenskrets (sequential circuit)

Logikenhet, vars utvärden vid en viss tidpunkt beror dels på dess invärden, dels på dess inre tillstånd vid denna tidpunkt och vars inre tillstånd i sin tur beror på närmast föregående invärden och inre tillstånd.

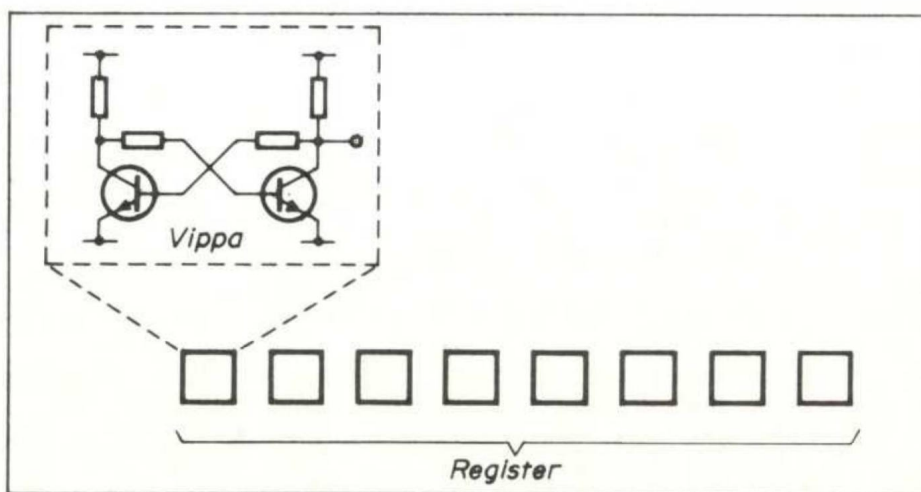
Det är tydligen inte så lätt att förklara vad en sekvenskrets är! Och i själva verket är "sekvenskrets" ett så allmänt begrepp att praktiskt taget alla digitala kretsar kan inkluderas i det, med kombinationskretsarna som ett specialfall. En dator är exempelvis en sekvenskrets.

Tydligt är att en sekvenskrets har "minne" av vad som hänt tidigare och vi skall i detta kapitel behandla de allra enklaste kretsarna med "minne". En bistabil vippa kommer ihåg hur man ställt den och är alltså en sekvenskrets. Register och räknare består av ett antal vippor. Med sekvenskretsar menar vi i detta kapitel "vippor, register och räknare".

Vi skall börja med principen för lagring av tal i register. Därefter behandlar vi registrens byggstenar - vipporna. Och slutligen skall vi studera uppbyggnad och funktion hos några vanliga register och räknare.

1 Register, koder, tal

I alla digitala system har man behov av kretsar som kan lagra tal. Sådana kretsar kallar vi här register^{*)}. Ett register består i princip av ett antal vippor och varje vippa kan lagra en binär siffra. Man anger därför registrets storlek i antal bitar. Figur 2.1 visar ett åtta bitars register och det består alltså av åtta vippor^{**)}.



Figur 2.1
Åtta bitars register.

^{*)} Inom datatekniken har "register" en bredare betydelse. Ett register kan vara information lagrad på ett magnetband, exempelvis folkbokföringsregistret.

^{**)} Vi behandlar här enbart statiska register uppbyggda av vippor. Ett dynamiskt register består i princip av kondensatorer i stället för vippor.

Åtta binära siffror kan kombineras på $2^8 = 256$ olika sätt. Vi kan alltså lagra 256 olika tal i ett åtta bitars register.

Normalt använder vi decimala tal. Vi kan gå över till andra talsystem, exempelvis till binära tal med hjälp av kodare (encoders) som kodar det decimala talet till lämplig form för lagring i det binära registret. När vi sedan vill "läsa av" innehållet i det binära registret använder vi en avkodare (decoder) som avkodar innehållet, dvs överför det binära innehållet till decimal form. Kodare och avkodare är kombinationskretsar och vi har sett exempel på sådana i samband med "adressavkodning" av en buss i kapitel 1.

1.1 Binärkod

Det binära talsystemet är ett positionssystem med basen 2 och därmed blir positionsvikterna (eller bitvikterna) potenser av 2.

	MSB							LSB
Bitnummer:	7	6	5	4	3	2	1	0
Bitvikter:	128	64	32	16	8	4	2	1
(dvs:	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0)

Figur 2.2

Bitnummer och bitvikter i ett åtta bitars binärkodat register.

I ett åtta bitars register får den lägsta positionen (LSB) vikten 2^0 och den högsta (MSB) vikten 2^7 . Se figur 2.2. Man brukar därför börja numreringen av bitarna i ett register med 0 och inte 1.

Innehållet i ett binärkodat register kan anges på flera olika sätt. Låt oss ta det decimala talet 212 som exempel. Detta tal kan skrivas som:

$$212 = 1 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 =$$

$$= 2^7 + 2^6 + 2^4 + 2^2$$

och lagrat i ett åtta bitars register ser talet därför ut på följande sätt:

	MSB							LSB
Bitnummer	7	6	5	4	3	2	1	0
Värde	1	1	0	1	0	1	0	0
Bitvikt	128	64	32	16	8	4	2	1

Talet 212 kan alltså skrivas i binär form som 11010100.

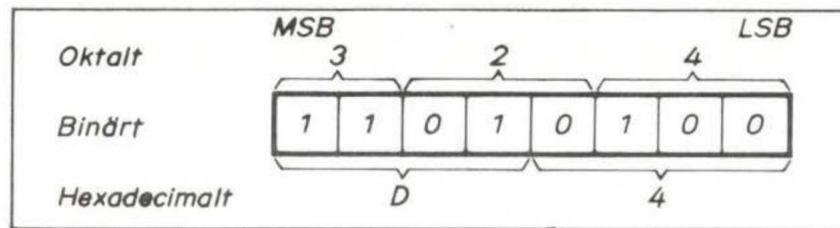
Det är ofta många bitar, dvs många binära siffror, att hålla reda på och därför brukar binära tal ofta skrivas i oktal eller hexadecimal kod.

Decimaltalet 212 kan tecknas: $212 = 3 \cdot 8^2 + 2 \cdot 8^1 + 4 \cdot 8^0$, vilket skrivet i oktal form blir 324_8 .

Man anger ofta basen i talsystemet med ett index när förväxling kan befaras. Vi kan alltså skriva $212_{10} = 324_8$ och menar därmed att talet 212 är decimalt och 324 oktalt.

I det hexadecimala talsystemet använder man bokstäverna A - F för att beteckna de decimala talen 10 - 15. Decimaltalet 212 kan därför tecknas hexadecimalt som: $212 = 13 \cdot 16^1 + 4 \cdot 16^0$, dvs $D4_{16}$.

Figur 2.3 visar ett register med innehållet 212 (decimalt) avkodat i oktal och hexadecimal form. Oktala och hexadecimala koder är mycket vanliga i samband med programmering av minidatorer.

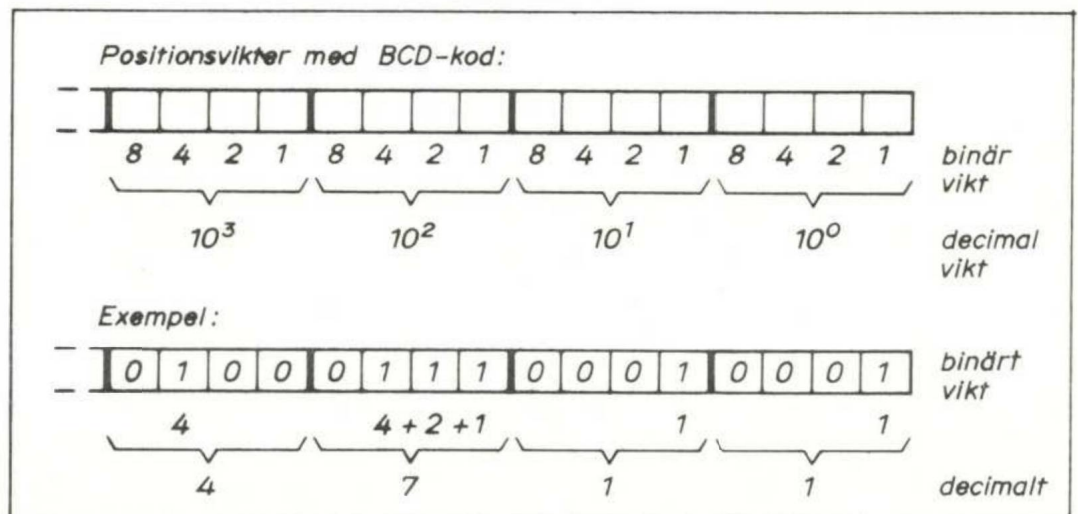


Figur 2.3

Register innehållande talet 212_{10} avkodat i oktal och hexadecimal form.

1.2 BCD-koder

BCD betyder "binary coded decimal". BCD-koder används när register, som är binärt uppbyggda, skall lagra decimala tal. Det finns många BCD-koder men vi behandlar bara en av dem här, den naturliga BCD-koden som ibland betecknas NBCD eller 1248-kod. Figur 2.4 visar ett exempel.



Figur 2.4

BCD-kodat register. Exempel: $0100011100010001_{\text{BCD}} = 4711_{10}$

I BCD-koden använder man fyra bitar för att koda en decimal siffra. Efter-som fyra bitar kan kombineras till 16 olika kombinationer (och man för en decimal siffra enbart behöver 10) utnyttjar man inte registret tillfullo vid BCD-kod.

Fördelen med BCD-kod ligger framför allt i den enkla decimala avkod-ningen och BCD-kod används därför exempelvis för fickkalkylatorer.

1.3 Parallell- och serieregister

Tal kan läsas in i eller ut från register på olika sätt. Man brukar här tala om parallell- eller serieregister. I en räknedosa skjuter vi in en ny siffra från höger i indikatorn varje gång vi trycker på en siffertangent. Det är en typisk seriematning.

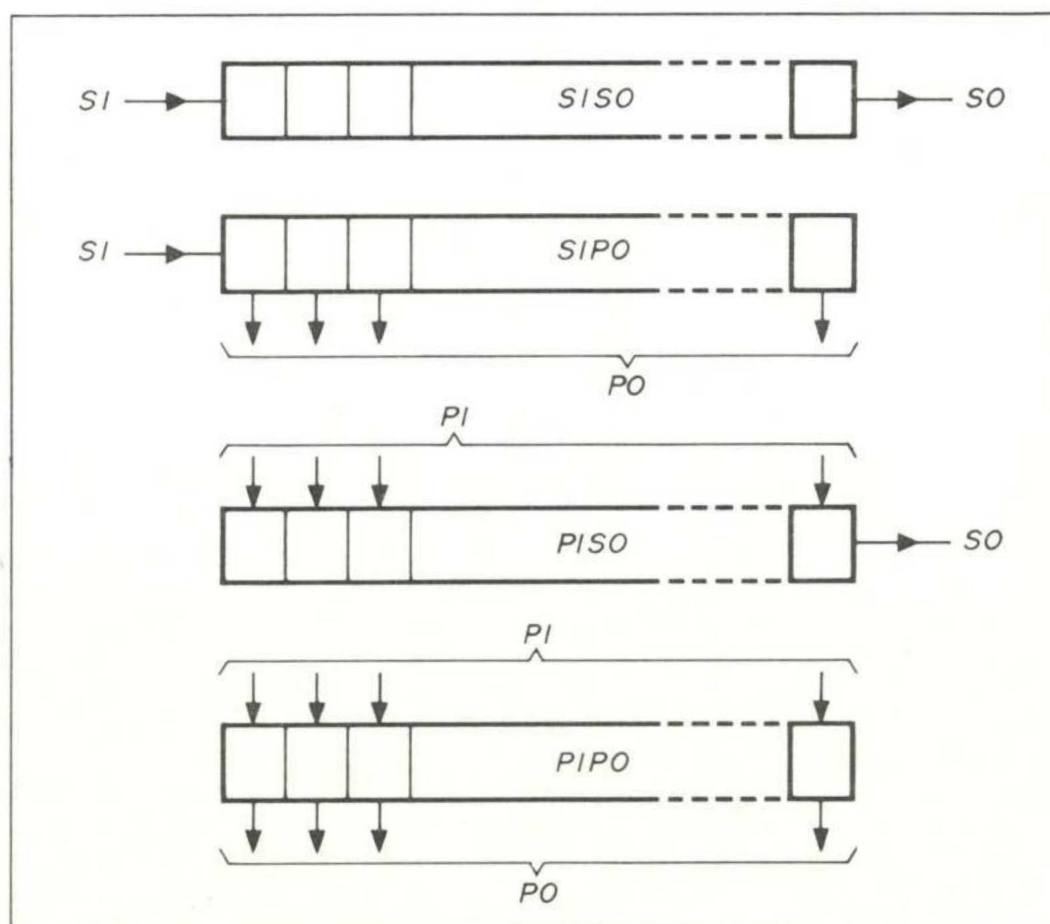
I datorer som skall arbeta snabbt har man inte tid att successivt skjuta in, eller snarare skifta in, långa tal i serieform i registren. Här laddas samtliga bitar i registret samtidigt eller "parallellt" som det heter.

I många tillämpningar laddas ett register i serieform men avläses i parallellform, exempelvis hos räknedosan. Det finns fyra olika kombinationer av in- och utläsning och de brukar betecknas på följande sätt (figur 2.5):

SISO = serie in och serie ut
SIPO = serie in och parallell ut
PISO = parallell in och serie ut
PIPO = parallell in och parallell ut

Man använder ofta termerna läsa (r = read) och skriva (w = write) i samband med inskrivning till och utläsning från ett register eller minne.

SIPO-registret utför en serie/parallell-omvandling av data och PISO en parallell/serie-omvandling. Dessa omvandlare är vanliga exempelvis i utrustningar som internt arbetar parallellt men mottar eller sänder information i serieform.



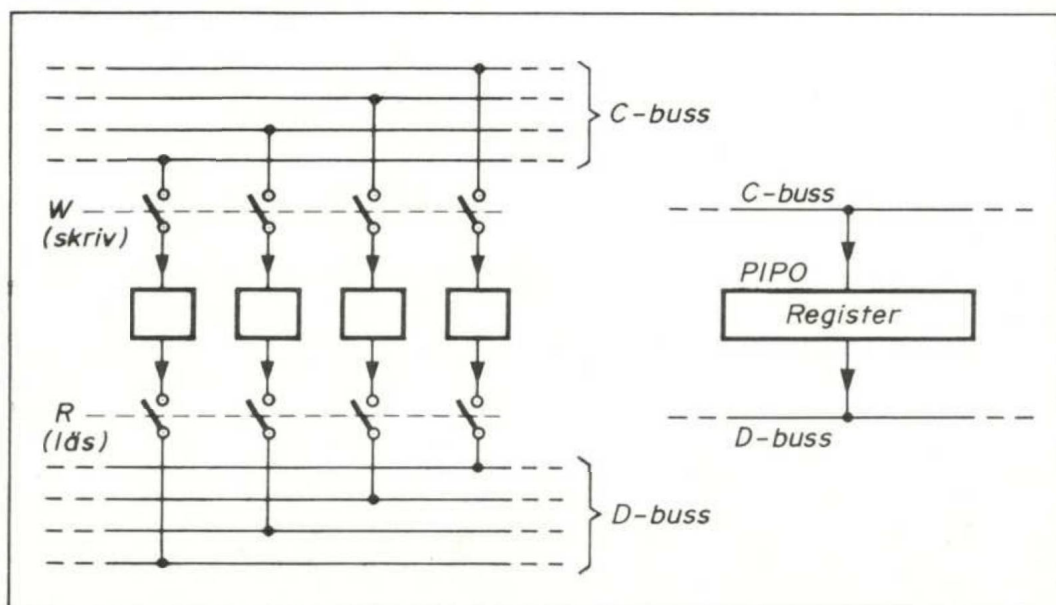
Figur 2.5
Olika sätt att skriva och läsa information i register.

2 Vippor

Vi skall i detta avsnitt bekanta oss med vipporna i olika typer av register. Först skall vi rent principiellt diskutera vippornas uppgifter i parallell- och serieregister. Därefter skall vi ge oss in på den konkreta uppbyggnaden av några vanliga vippor.

2.1 Vippornas uppgifter i parallell- och serieregister

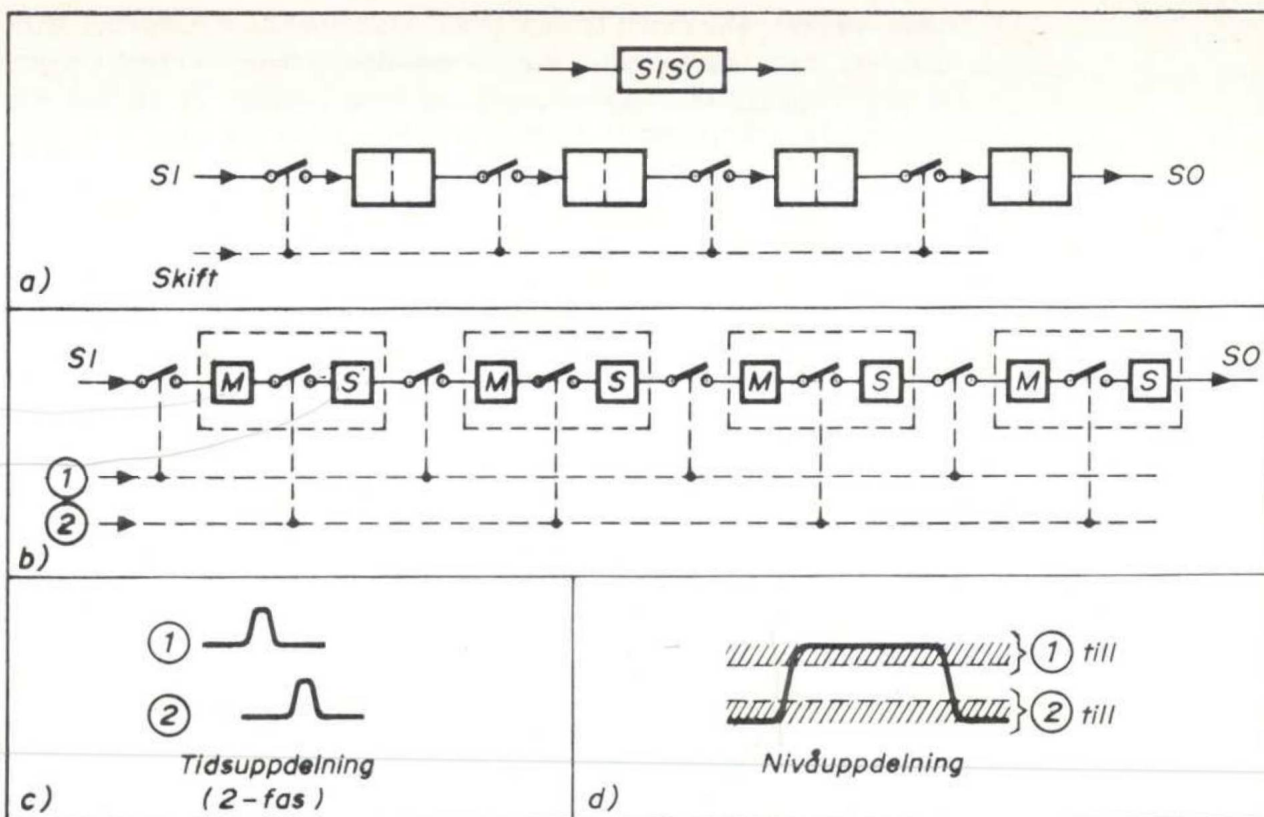
Det är, som tidigare nämnts, vipporna som utgör byggstenarna i register av olika slag. Figur 2.6 visar ett PIPO-register och vipporna är här symboliskt tecknade som fyrkanter. Register av denna typ är mycket vanliga i datorer. På kommando skrivs data in från en bussledning, C-bussen i figur 2.6. Vid ett senare tillfälle läses dessa data ut på en annan bussledning, D-bussen i figur 2.6. Ett PIPO-register är som synes mycket enkelt i sin uppbyggnad eftersom vipporna ej kommunicerar med varandra. Det är egentligen endast signalgrindarna på in- och utgångarna som kan ställa till bekymmer. Som vi strax skall se, integreras normalt signalgrindarna på ingångarna med vipporna. Signalgrindarna på utgångarna är däremot speciella grindar eftersom de matar en bussledning. (De utgörs normalt av grindar med öppen kollektor eller av s k tristate-grindar och detta för att bussen skall vara fri för andra uppgifter under de tider då vårt register ej sänder ut data.)



Figur 2.6

Principen för ett PIPO-register (med vanliga blockschemasymboler till höger).

Ett SISO-register visas i princip i figur 2.7. Här skall data successivt skiftas in i registret och detta ställer helt andra krav på vippornas uppbyggnad.



Figur 2.7
Principen för ett SISO-register.

I figur 2.7a ligger fyra vippor i serie. När signalgrindarna slås till måste var och en av dessa fyra vippor ta emot ny information från vänster och samtidigt avge den tidigare lagrade informationen till höger. Varje vippa måste alltså vid skiftningen kunna hålla två binära siffror, en på ingången och en på utgången. Den nödvändiga isoleringen mellan ingång och utgång under skiftningen är markerad med ett streck genom varje vippa.

Det finns många sätt att realisera en funktion av den beskrivna typen. När man tidigare byggde vippor med diskreta komponenter, och var mån om att göra konstruktioner med få komponenter, använde man en kondensator som "transientminne" under skiftningen. Numera använder man integrerade vippor och man bygger då samman vipporna två och två som figur 2.7b visar. Här har man nu dubbelt så många vippor som antalet bitar i registret. Man har också dubbla antalet signalgrindar. Man sluter först de switchar som markerats med ① i figur 2.7b ett kort ögonblick, medan switcharna markerade med ② står i frånläge. Informationen från tidigare steg inläses därvid till vipporna M, dvs "mästarna".

När switcharna ① därefter gått i frånläge sluts switcharna ② ett kort ögonblick. Informationen läses nu över från "mästarna" till "slavarna" som är markerade med S i figur 2.7b.

Under själva skiftoperationen, dvs mellan puls ① och ②, innehåller varje MS-kombination av vippor två värden, ett på ingången (M) och ett på utgången (S). När skiftoperationen är avslutad innehåller både mästarerna och slaven samma information.

Vissa register körs med tvåfasklocka (exempelvis dynamiska MOS-register) och man använder härvid just den princip vi visat i figur 2.7b. I bipolärt uppbyggda register använder man vanligen en nivåkänslig krets för styrningen av grindarna. Man arrangerar då styrningen så att grindarna ② ligger till i normalläge, L i figur 2.7d, medan grindarna ① ligger till vid skiftpulss, dvs H i figur 2.7d. Funktionen blir densamma. Det är i båda fallen en tidsuppdelning vid styrningen av signalgrindarna men i det senare fallet sker detta via nivåkänsliga kretsar. Man brukar kalla vippor av den senare typen för MS-vippor.

Vi skall nu lämna de principiella resonemangen och ge oss in på den konkreta uppbyggnaden av vippor avsedda för register. De flesta digitala system styrs synkront från en klockpulsgenerator. Man använder "klockade" vippor och därmed menas att grindnäten styrs av klockpulsgeneratoren. Vi skall börja med att beskriva funktionen hos några vanliga klockade vippor.

2.2 Asynkron och synkron trigging

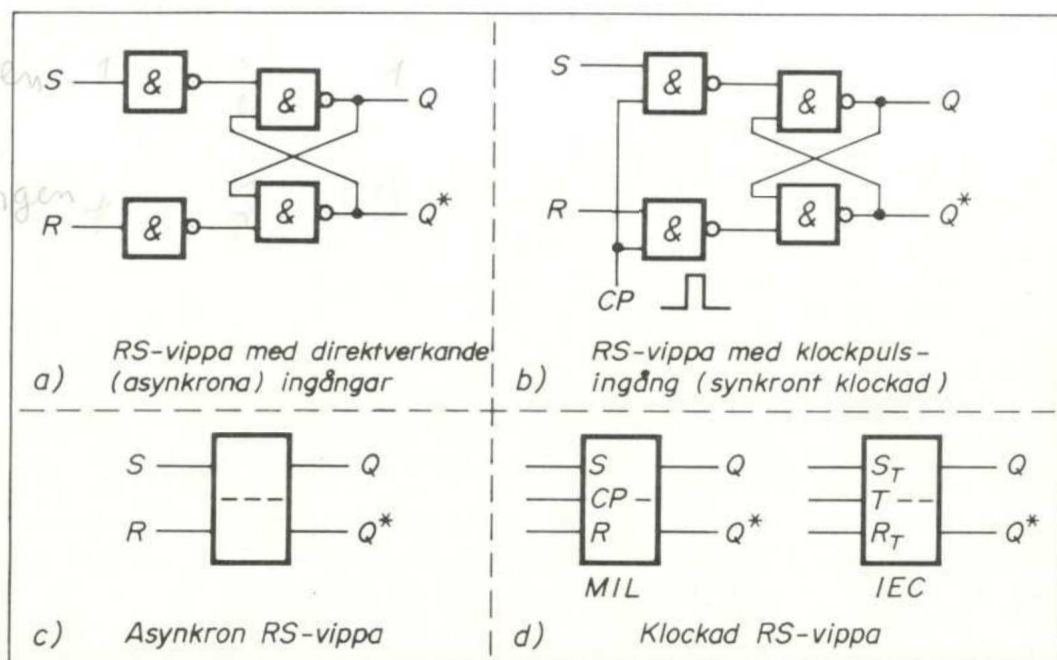
Om vi kompletterar den tidigare visade latchen i figur 3.10 i "Pulskretsar" med två inverterare får vi RS-vippan i figur 2.8a. Hög S-ingång ettställer vippan ($Q = H$). Hög R-ingång nollställer vippan. Omställningen av vippan sker omedelbart vid signal på R- eller S-ingångarna. Dessa ingångar kallas därför "direktverkande" eller "asynkrona".

Om båda ingångarna är höga samtidigt kommer både Q och Q^* -utgångarna att bli höga. Detta är ett icke önskvärt tillstånd som kallas "pseudostabilt" enligt IEC:s terminologi. Det skall normalt inte användas. För att markera RS-vippans pseudostabila tillstånd ($Q = Q^* = H$) kan enligt IEC vippan i figur 2.8a benämnas:

RS(H)-vipa

Vill vi synkronisera vippans omslag med en, för ett helt digitalsystem gemensam klockpulssignal, kan detta enkelt utföras enligt figur 2.8b.

när det är en 1:a på S-ingången ska det vara en 1:a på Q utgången



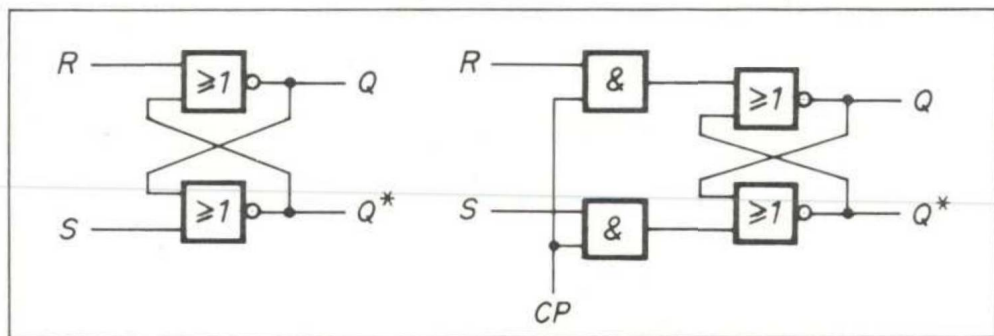
Figur 2.8
Asynkron och synkron trigging av en RS-vippa.

När klockpulssignalen, "klockan", är låg är båda ingångsgrindarna stängda och de öppnas enbart den tid då klockan ligger hög.

Den grafiska symbolen för en klockad vippa måste ha en klockpulsingång och en rad olika beteckningar förekommer för denna, bl a CK (central clock), CP (clock pulse), T (toggle) eller G (gate).

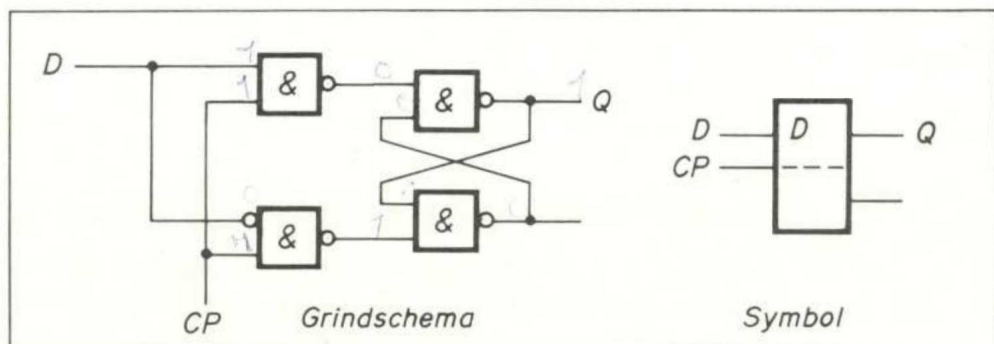
Figur 2.8d visar två vanligt förekommande symboler, Mil-standard från USA samt IEC-standard. Indexen T i R_T och S_T i IEC-symbolen anger att dessa ingångar endast är verksamma vid klockpuls. I vårt exempel gäller detta då klockan är hög. Vippor som triggas vid låg klocka förses med en inverterarrang på klockpulsingången.

I figur 2.9 ser vi motsvarande krets med NOR-grindar. Funktionen blir densamma, om vi bortser från det pseudostabila tillståndet som hos NOR-vippan blir $Q = Q^* = L$.



Figur 2.9
NOR-latch och klockad NOR-vippa av RS-typ.

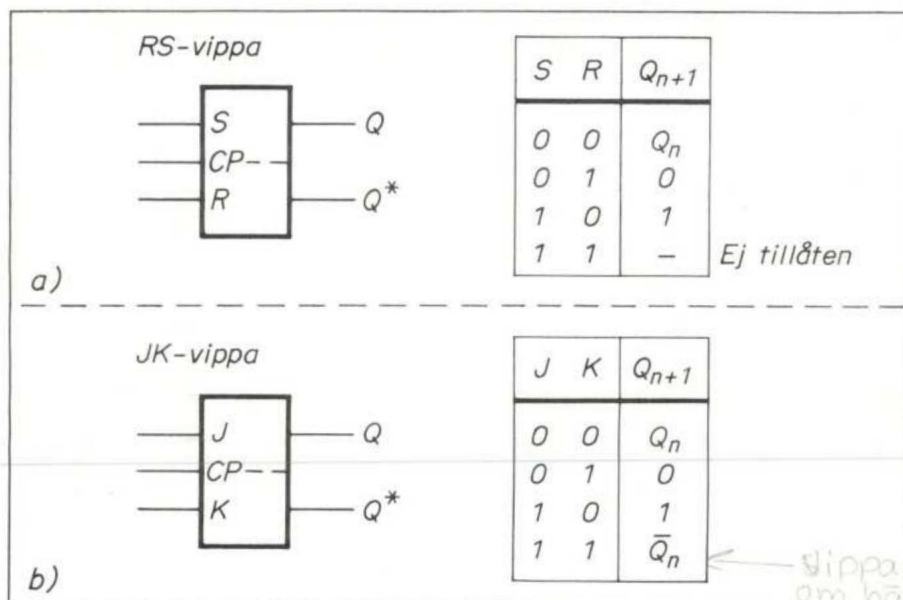
I digitala system har man ofta behov av att lagra (frysa) inställningen hos en buss vid ett visst ögonblick. Detta kan göras med hjälp av D-latchar av den typ som visas i figur 2.10. D-latchen utgörs av en RS-vippa, figur 2.8b, som kompletteras med en inverterare på ingången. Då klockan är hög följer utgången Q dataingången D, men i det ögonblick klockan går låg låses utgångsvärdet Q. Vi har därmed gjort en "synkroniserad" avläsning av inställningen på dataingången, exempelvis från en databuss som i figur 2.6.



Figur 2.10
D-latch.

2.3 Klockade RS- och JK-vippor

En klockad RS-vippa har en sanningstabell som framgår av figur 2.11. Beteckningarna Q_n och Q_{n+1} i sanningstabellen avser utgångens ställning före klockpuls (Q_n) och efter klockpuls (Q_{n+1}). En viss kombination av insignaler ($S = R = 1$) är förbjuden eftersom den kan leda till ett obestämt tillstånd. Man utnyttjar alltså inte ingångarna till fullo i en RS-vippa.



Figur 2.11

Sanningstabeller för klockade RS- och JK-vippor (definitioner).

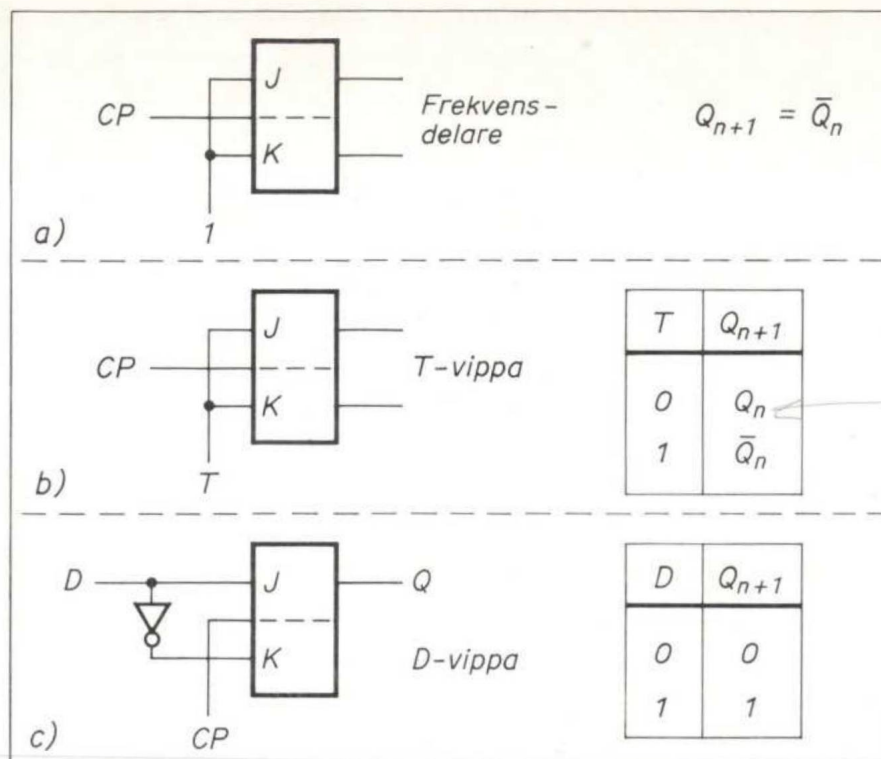
I figur 2.11b har vi definierat funktionen hos en JK-vippa. Den enda skillnaden mellan en RS- och en JK-vippa ligger i att RS-vippans förbjudna inkomination ($S = R = 1$) i JK-vippan har ersatts av en tillåten insignal ($J = K = 1$) som gör att JK-vippan slår om (toggle) för varje klockpuls. För övrigt är JK-vippans J-ingång en direkt motsvarighet till RS-vippans S-ingång (liksom K motsvarar R). Som komponent är en JK-vippa avsevärt mera användbar än en RS-vippa. Detta är orsaken till att JK-vippor är helt dominerande på marknaden.

Man frågar sig kanske om bokstäverna J och K har någon speciell betydelse i likhet med R (reset) och S (set). Bokstäverna J och K har ingen sådan betydelse - de lär ha valts för att de är två bokstäver efter varandra i alfabetet.

2.4 JK-vippan kopplad som räknare, T-vippa eller D-vippa

JK-vippan är en generellt användbar vippa. J- och K-ingångarna förbereder vippans omställning och denna omställning verkställs sedan synkront med klockpulsen. J- och K-ingångarna kallas därför preparatoriska ingångar (preparatory inputs), dataingångar eller sätt-ingångar.

En JK-vippa kan med enkla yttre kopplingar användas som frekvensdelare, T-vippa eller D-vippa, figur 2.12. Som nämnts tidigare avser beteckningen Q_n vippans läge före triggpuls och Q_{n+1} läget efter triggpuls.



Figur 2.12

JK-vippa kopplad som: a) Frekvensdelare, b) T-vippa, c) D-vippa.

I figur 2.12a är $J = K = 1$ och då slår vippan om för varje klockpuls. Om klockpulssignalen är en fyrkantvåg kommer tydligen även utsignalen att bli en fyrkantvåg men med halva infrekvensen. Kaskadkoppling av sådana steg kan som vi snart skall se användas för räkning av klockpulser. Kretsen i figur 2.12a kallas därför ofta för binärräknare eller Modulo 2-räknare.

I figur 2.12b har J- och K-ingångarna hopkopplats till en gemensam T-ingång. När $T = 1$ räknas inkommande klockpulser precis som i frekvensdelaren ovan, men när $T = 0$ påverkas ej vippan. T-vippan har fått sitt namn av engelska "toggle". Olika fabrikanter och litteraturkällor använder bokstaven T på olika sätt. I IEC-standard betecknas ofta klockpulseringången med T.

Figur 2.12c visar en JK-vippa kopplad som D-vippa. D-vippans funktion har vi ju tidigare diskuterat i anslutning till figur 2.10.

2.5 JK-vippans trigging

I diskret uppbyggda JK-vippor använde man tidigare kondensatorer som "transientminnen" under skiftoperationen. Triggpulsingången var då AC-kopplad och krav ställdes på pulsflankens form. Figur 2.13 visar en sådan vippa. Kondensatorerna C_1 fungerar här som transientminnen.

I moderna integrerade vippor används vanligen DC-kopplade triggnät. Man skiljer härvid på två typer av DC-triggade vippor:

- Flanktriggad vippa (transition operated, edge triggered)
- MS-vippa (master slave)

J	K	Q
0	0	$\overline{Q_n}$
0	1	0
1	0	1
1	1	1

T_2 ströms
 T_1 betätnad
- diodens backspänns
det händer
klockpulsen kommer

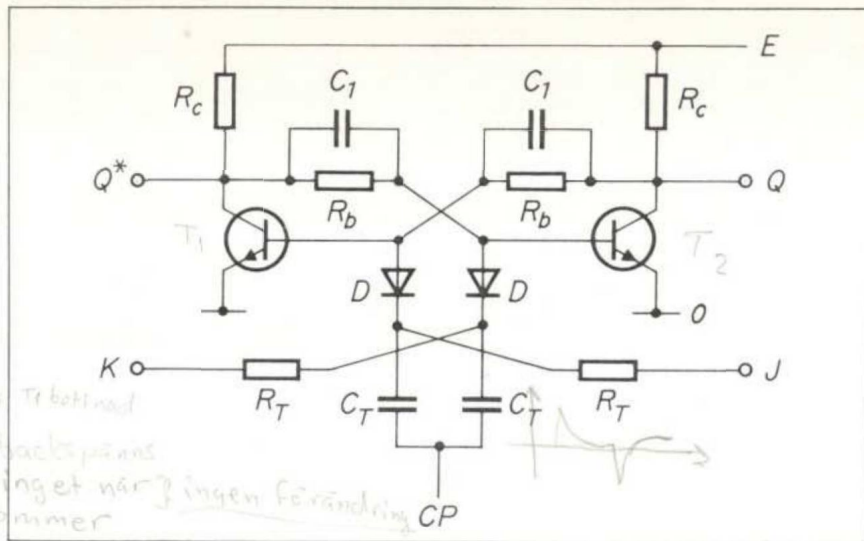
Q_n
↑
Hälsändet
som varit före
klockpulsen

Här står den alltid
om på bak kanten av
klockpulsen

∴ Negativ trigging

Om en inverterare
läggs före själva
vippan ∴ den inverterar
klockpulsen

∴ Positiv trigging



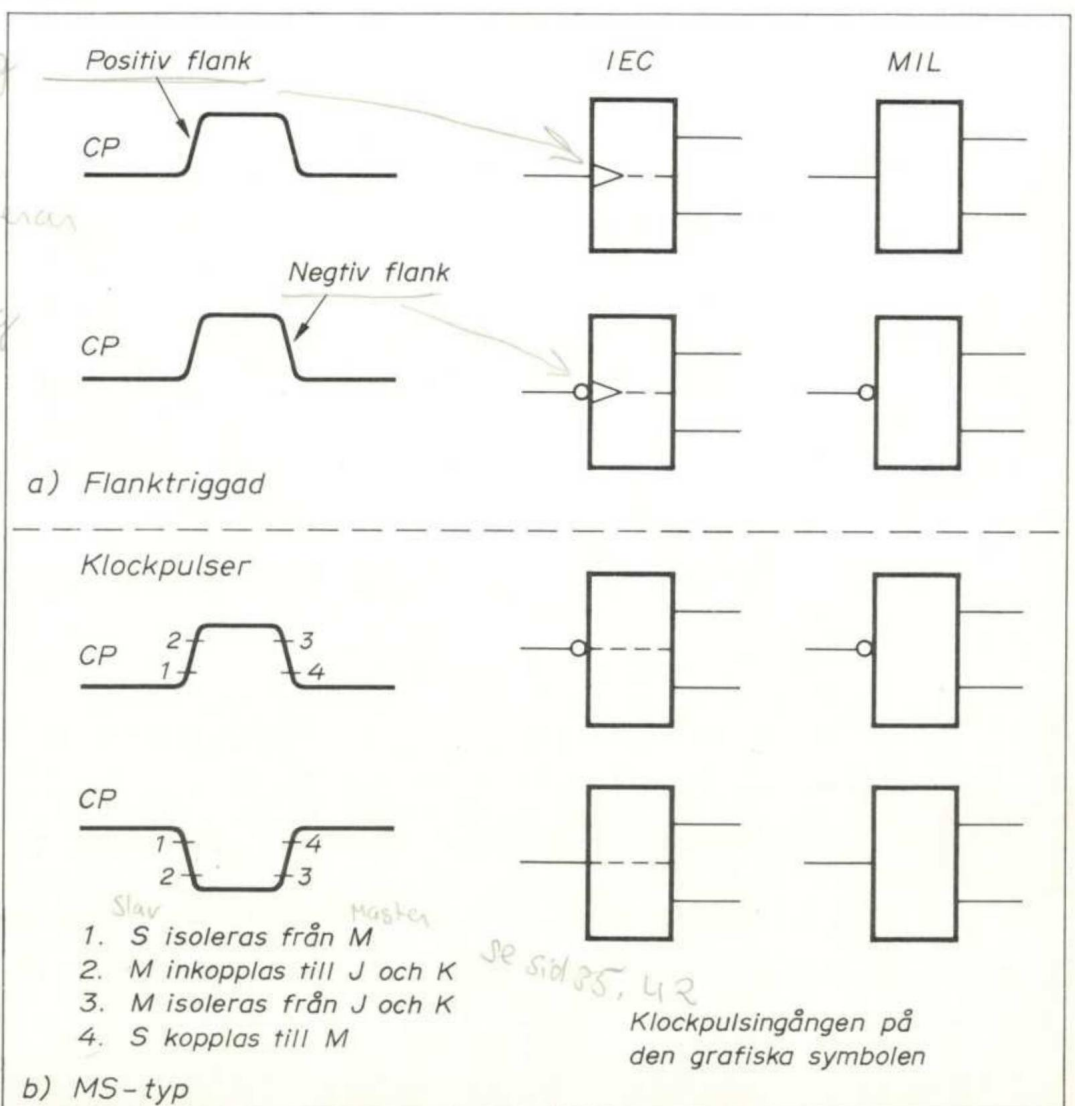
Figur 2.13

Flanktriggad (AC-kopplad) JK-vippa.

Dimensioneringsexempel:

$E = 6\text{ V}$, $R_C = 2,2\text{ k}\Omega$, $R_b = 10\text{ k}\Omega$, $R_T = 10\text{ k}\Omega$, $C_1 = 100\text{ pF}$, BSX20 och 1N4148.

(Trigga ej vippan från mekanisk kontakt med kontaktstudsar !)



Figur 2.14

Trigging av flanktriggad vippa och MS-vippa.

I en flanktriggad vippa av DC-kopplad typ sker omslaget vid en viss på klockpulsens flank, figur 2. 14a. Man talar här om klockpulsens positiva flank (positive going edge) eller negativa flank (negative going edge). Omedelbart efter triggningen blockeras ingångsgrindarna. På grund av stegfördröjningen blir in- och ut signaler åtskilda och därmed kan flanktriggade vippor kopplas som räknare eller skiftregister.

I en MS-vippa sker, som vi sett tidigare, separationen av in- och utstörhet med hjälp av två vippor, mästaren och slaven. Under klockpulsen läses instörheten in till mästaren, medan slaven kvarhåller tidigare värde, och först vid klockpulsens slut slussas instörheten vidare till slaven, figur 2. 14b. MS-vippan liknar i mycket en sluss där in- och utgång är skilda åt genom att endast en slussport i taget är öppen.

Vissa MS-vippor triggar på positiv klockpuls och andra på negativ.

Det finns en rad olika sätt att markera klockpulsingångens funktion i den grafiska symbolen. IEC använder en pil för att markera flanktriggning (transition operated) och ingen pil vid sk nivåtriggning (level operated). De flesta fabrikanter markerar triggnings- och utgång på negativ flank med en inverterings-ring på klockpulsingången. Den amerikanska Mil-standarden gör ingen skillnad på flank och nivåtriggning i den grafiska symbolen. Data-bladen använder övervägande Mil-symboler och vi kommer därför att använda dessa i fortsättningen.

MS-vippor som triggas av en positiv klockpuls brukar ha klockpulsingången försedd med ring eftersom utgången hos en sådan MS-vippa ställs om först på klockpulsens negativa flank (bakkanten).

2.6 En JKMS-vippa

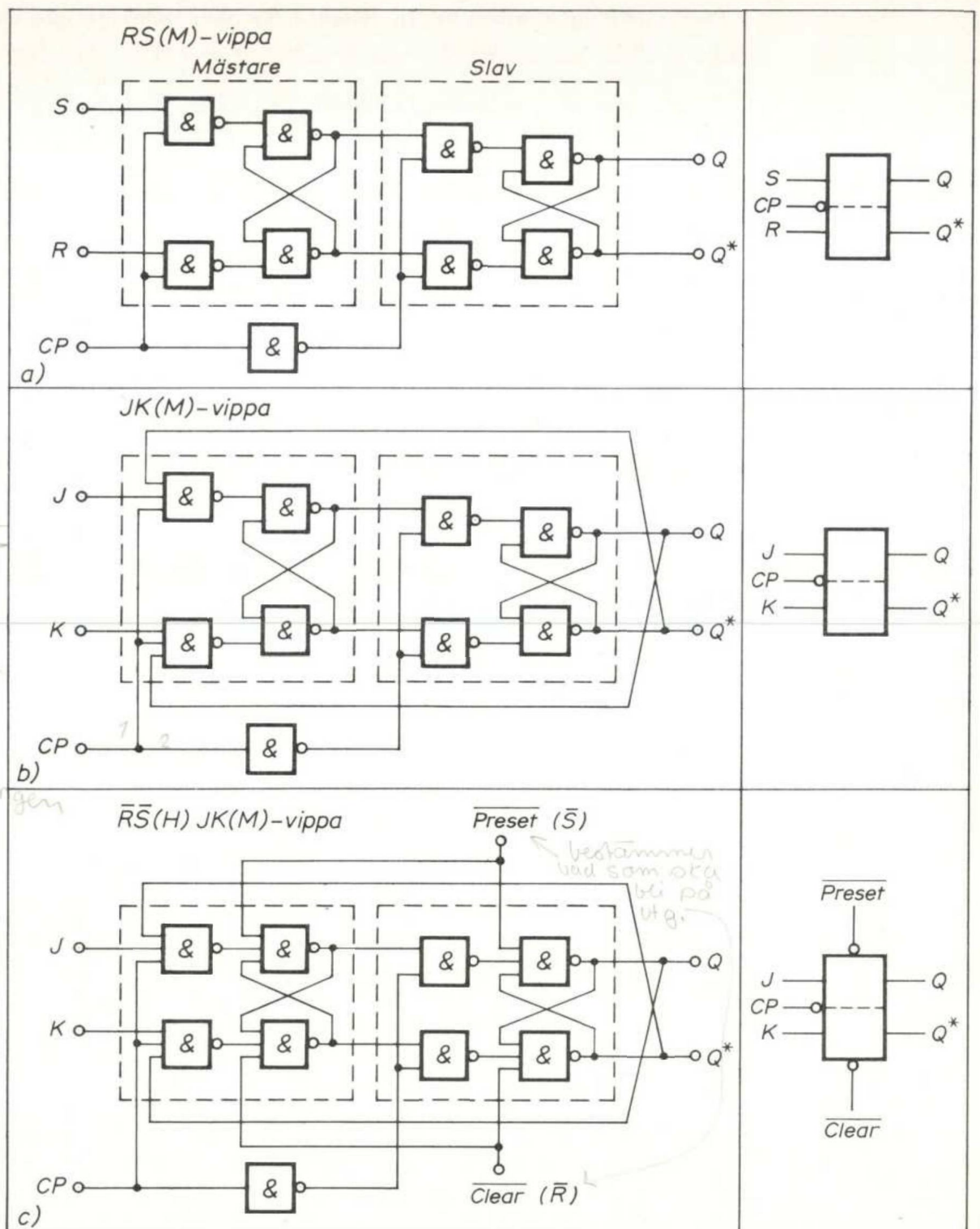
JKMS-vippan är i dag den utan jämförelse viktigaste typen av klockade vippor. Vi har redan tidigare gjort den första bekantskapen med MS-vippan och dess funktion. Vi skall nu visa en av de många varianter av JKMS-vippor som förekommer. För att konkretisera vår diskussion förutsätter vi nedan att både mästaren och slaven utgörs av grindade RS-vippor av den typ som vi tidigare mött i figur 2. 8b. Figur 2. 15a visar en MS-vippa av RS-typ uppbyggd med NAND-vippor.

Genom att styra både mästaren och slaven från samma klocka men invertera klockpulssignalen till slaven kan man få den nivåseparering som vi tidigare beskrivit, figur 2. 7d. Detta förutsätter givetvis att både inverteraren och vippornas klockpulsingångar ges lämpliga omslagsnivåer.

I den vanliga MS-vippan ligger in- och utgång alltid separerade. Då klockan är låg är ingången till mästaren M blockerad medan M och S ligger sammankopplade. Då klockan går hög håller S kvar sin ställning samtidigt som den frikopplas från M. Då ingångarna öppnas vid en något högre nivå kan M ställas om utan att störa S. Då klockpulsen åter går låg överförs M:s inställning till S.

Genom att återkoppla RS-vippans utgångar till J- och K-ingångarna får man en JKMS-vippa, eller med IEC-terminologi JK(M)-vippa. Figur 2. 15b.

Ofta behöver man kunna ettställa (preset) eller nollställa (clear) en vippa utan att klocka den. Man behöver därför direktverkande, asynkrona ingångar. Kompletterar vi vår JKMS-vippa med sådana ingångar får vi en



Figur 2.15

MS-vippor av RS- och JK-typ.

a) $RS(M)$ -vippa, b) $JK(M)$ -vippa, c) $\overline{RS}(H)JK(M)$ -vippa.

vippa som med IEC-terminologi kan tecknas $\overline{RS}(H)JK(M)$ -vippa, figur 2.15c.

Ingångarna $\overline{R}$ och $\overline{S}$ är asynkrona och aktiva i läge låg. Normalt betecknas de i datablad: preset (= $\overline{S}$) och clear (= $\overline{R}$). Ringarna på ingångarna anger att de är aktiva i läge låg och ibland används också ett inverteringsstreck över beteckningarna (preset och clear) på samma sätt som vi tidigare använt för $\overline{R}$ och $\overline{S}$. Ordet reset är synonym till clear.

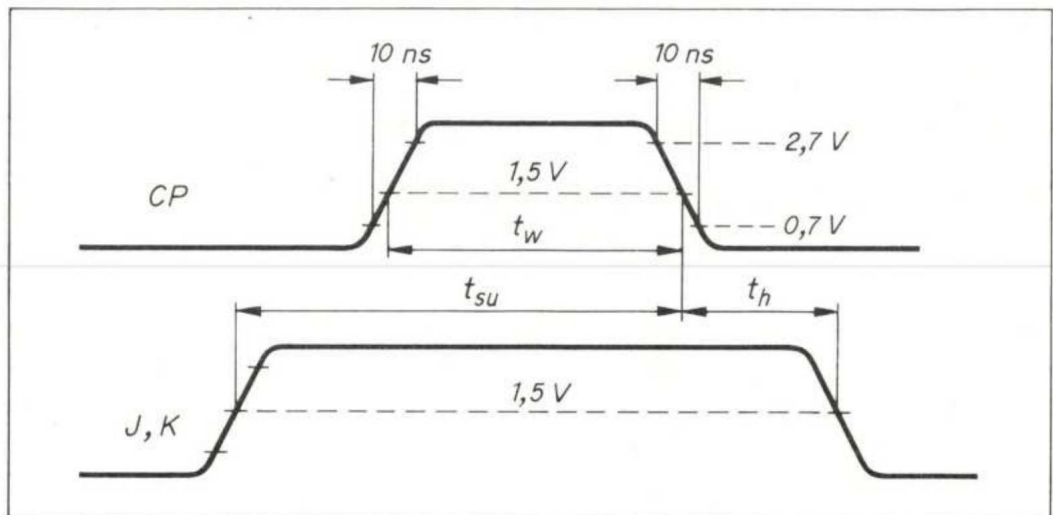
2.7 Krav på vippans insignaler

inte
så nogg

Om en klockad vippa skall fungera tillfredsställande ställs det vissa krav på insignalerna. Dataingångarna (J och K) och direktingångarna (preset och clear) måste ha legat på konstanta spänningar så länge att vippan är redo att motta klockpulsen.

För vippor brukar man därför specificera två tider:
sätt-tid (set-up time)
håll-tid (hold time)

Dessa tider kan inte mätas upp direkt utan är att betrakta som ett krav på vippans insignaler för att den skall kunna fungera tillfredsställande. Figur 2.16 visar hur dessa tider specificeras för TTL-vippan 7473.



Figur 2.16

Specifikationer av sättid och hålltid för en JK-vippa i TTL-familjen.
 t_{su} = sättid, t_h = hålltid.

Det finns även andra liknande restriktioner på insignalernas form, exempelvis krav på viss minsta längd hos klockpuls och nollställningspuls.

Tabellen i figur 2.17 ger exempel på minsta tillåtna värden för dessa tider för TTL-vippan 7473.

Recommended operation, minvärden för vippan 7473.			
sättid	(t_{su})	$t_{wclock} \downarrow$	ns
hålltid	(t_h)	$0 \downarrow$	ns
klockpulslängd	(t_{wclock})	20	ns
clearpulslängd	(t_{wclear})	25	ns
(pilen avser referensflank på klockpulsen)			

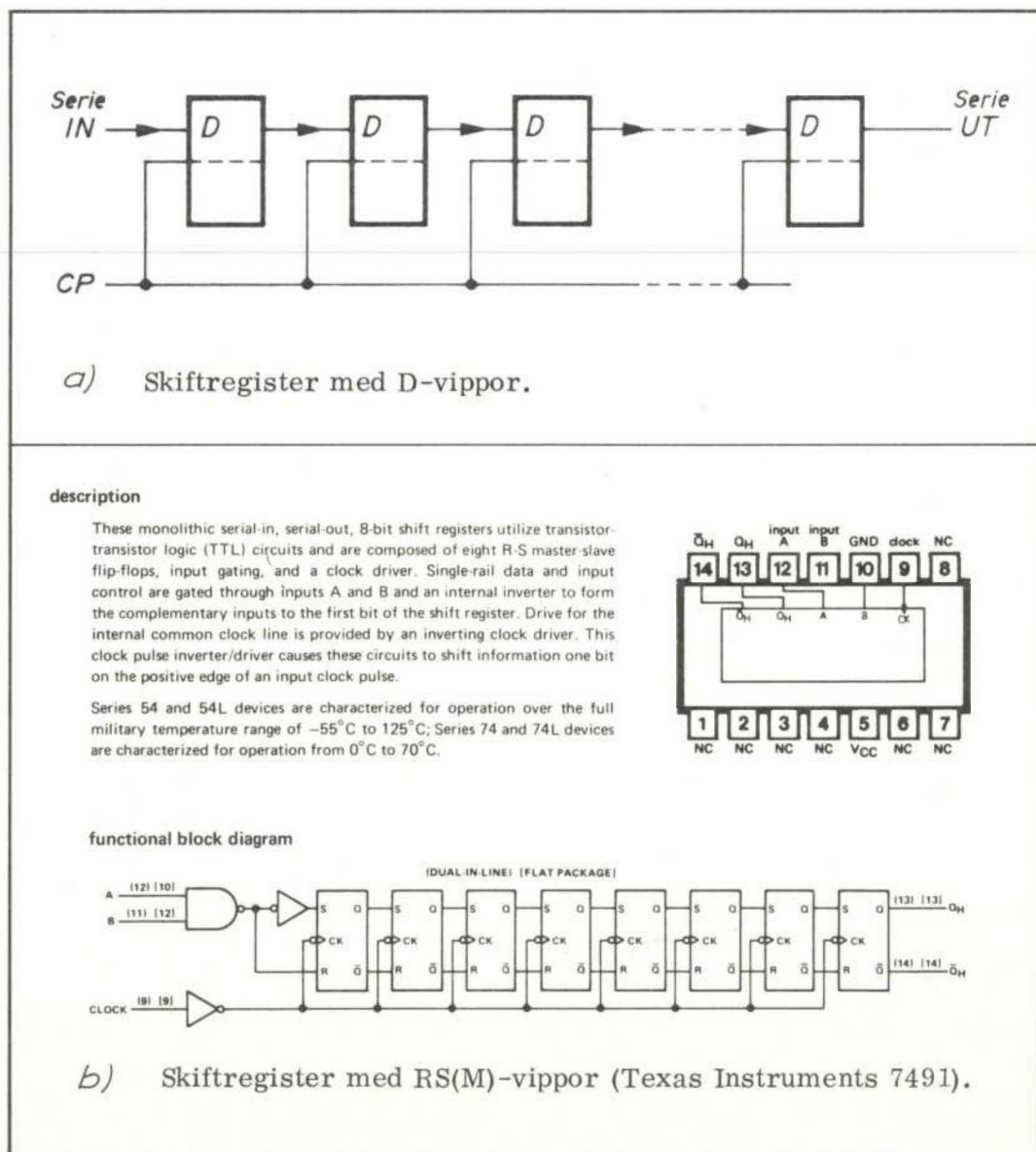
Figur 2.17

Specificerade krav på insignaler till TTL-vippan 7473.

3 Skiftregister

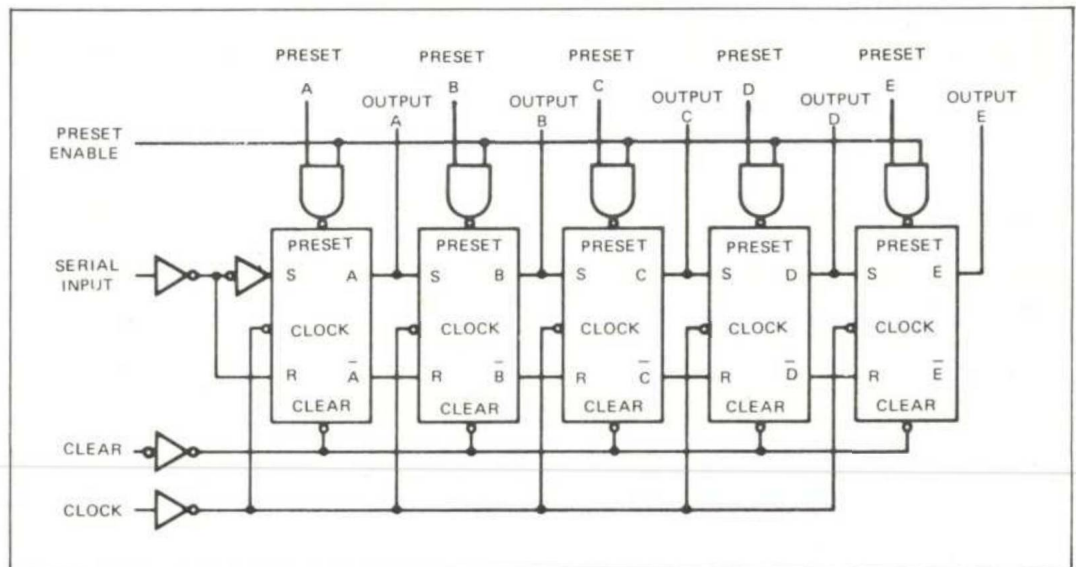
Register används för datalagring och förflyttning av data. Register är uppbyggda av klockade vippor av den typ vi studerat i föregående avsnitt. Eftersom det numera finns ett mycket stort antal integrerade kretsar innehållande register av olika typer behöver man inte själv koppla samman vippor till register.

Ett skiftregister byggs normalt som ett antal kaskadkopplade vippor av JK-typ, RS-typ eller D-typ. Figur 2.18a visar principen för ett skiftregister med D-vippor, dvs JKMS-vippor kopplade enligt figur 2.12. Figur 2.18b visar ett 8 bitars skiftregister uppbyggt av MS-vippor (7491). Registren i figur 2.18 är serieregister (SISO-typ).



Figur 2.18
Exempel på skiftregister med D-vippor och RS-vippor.

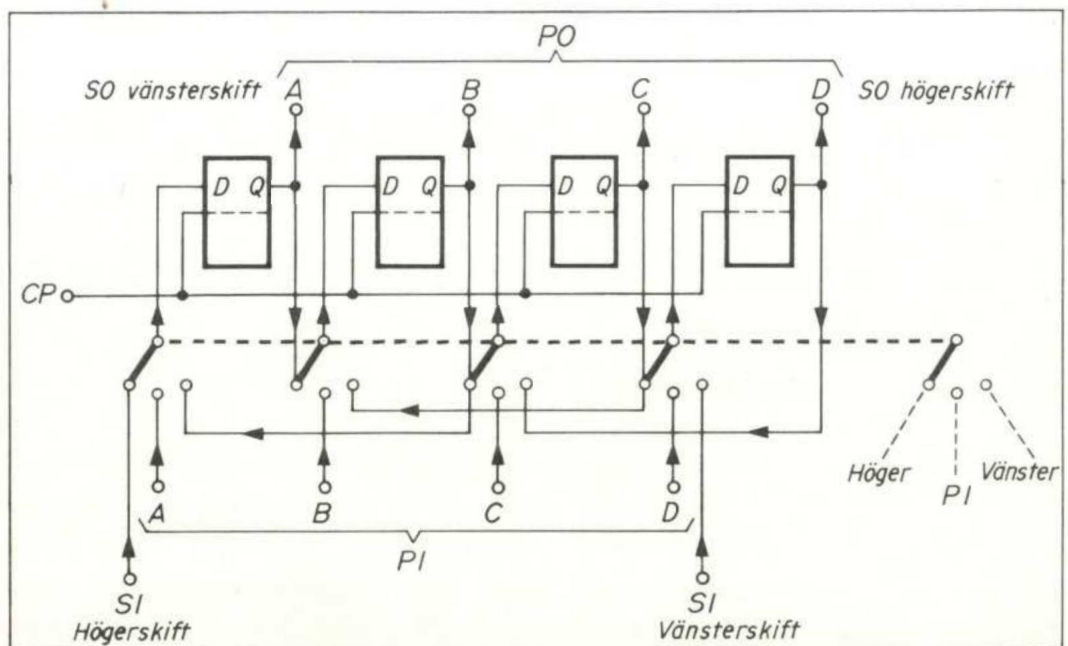
Vi har tidigare i figur 2.5 visat olika kombinationer av serie- och parallellmatning av register och hur dessa kan användas som serie/parallellomvandlare. Det finns generellt användbara skiftregister som kan utföra alla dessa omvandlingar. TTL-kretsen 7496 är ett exempel, figur 2.19. Den har såväl serieingång (serial input) som parallella ingångar (preset: A, B, C, D, E). Den har både serieutgång (output E) och parallella utgångar (output: A, B, C, D, E). 7496 är uppbyggd av RS(M)-vippor.



Figur 2.19

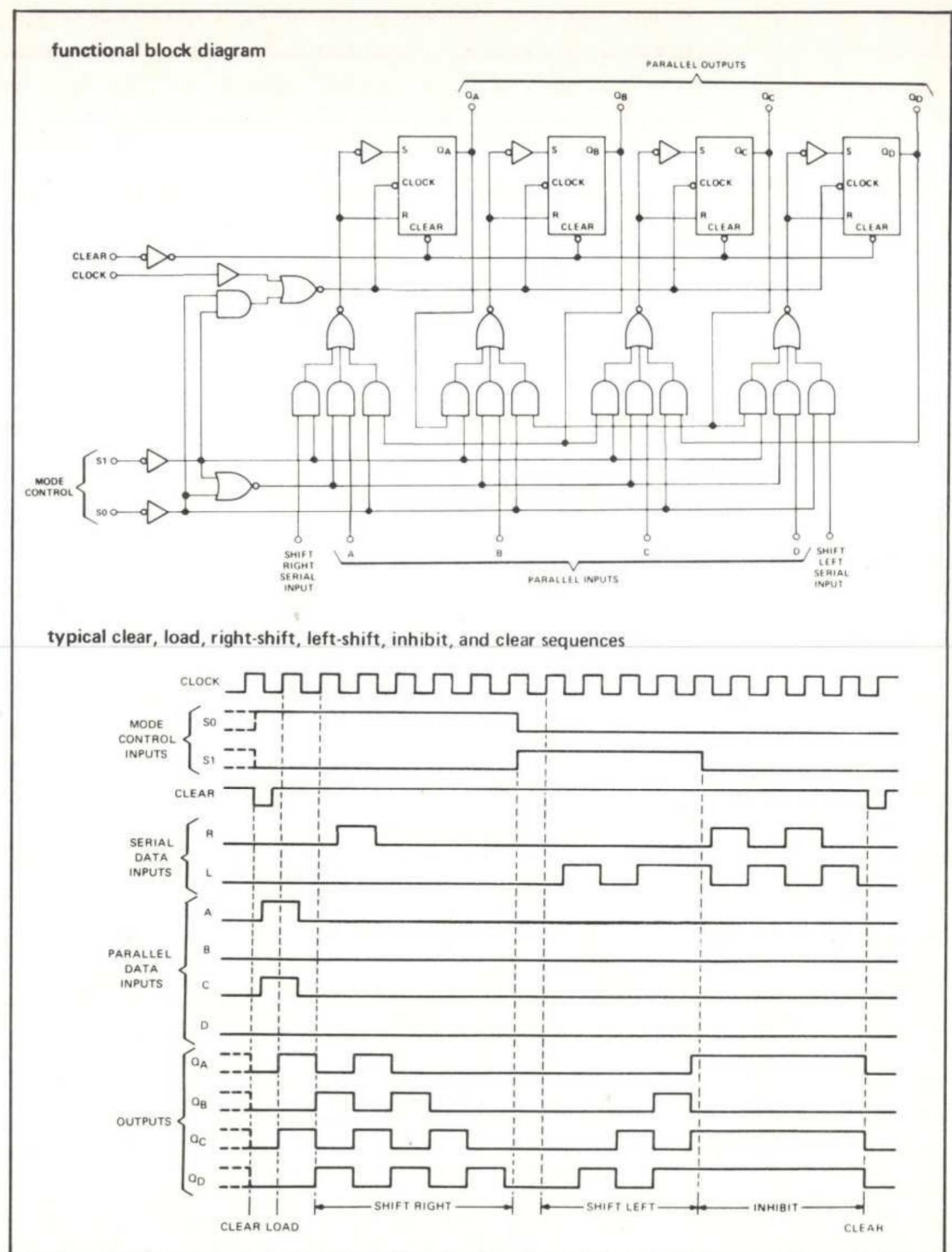
Exempel på ett generellt skiftregister som kan användas för samtliga funktioner enligt figur 2.5 (Texas Instruments, 7496).

I många tillämpningar är det väsentligt att kunna skifta både vänster och höger. Principen är enkel och framgår av figur 2.20. Omkopplingen utförs givetvis i praktiska kretsar av grindnät. Figur 2.21 visar ett generellt användbart fyra bitars skiftregister (74194) med vänster- och högerskift.



Figur 2.20

Principen för generellt register med höger/vänster-skift.



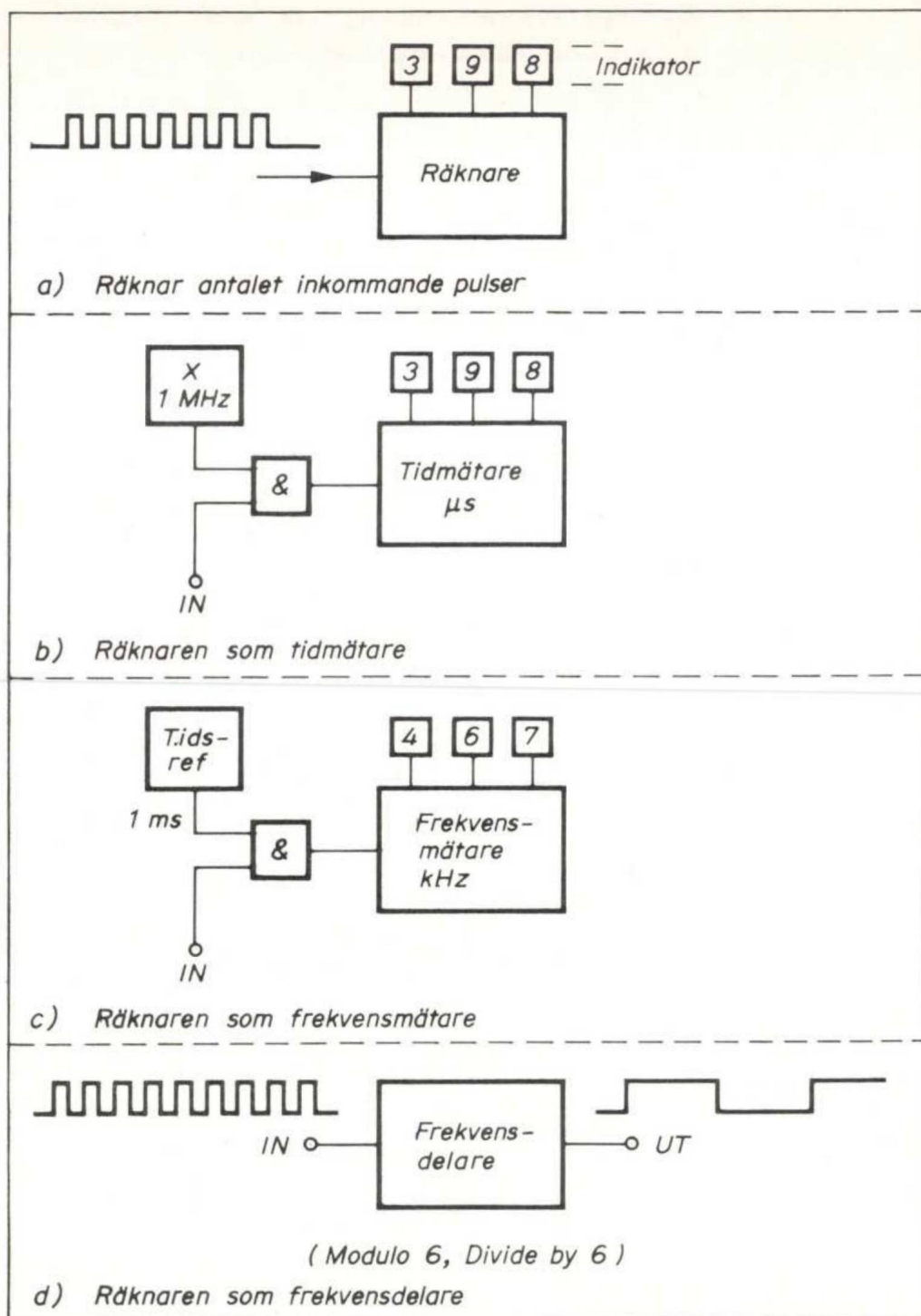
Figur 2.21
4-bitars skiftregister med vänster- och högerskift (74194 Texas Instruments).

4 Räknare

Räknare är viktiga byggblock i digitala system. Figur 2.22a visar principen för en räknare som räknar antalet inkommande pulser. I figuren har vi försett räknaren med en indikator (och avkodare).

Pulser från en stabil kristallosillator (X) kan tillföras räknaren via en OCH-grind. Vi får på detta sätt en tidmätare, figur 2.22b.

Om grinden styrs från en tidsreferens (i princip en monovippa) kan räknaren användas som frekvensräknare, figur 2.22c.



Figur 2.22

Principen för räknarens funktion som tidmätare, frekvensmätare och frekvensdelare. (I praktiken måste man tillgripa mer komplexa grindnät än enbart grinden G i figur b och c.)

I många fall önskar man en utsignal för varje gång ett givet antal impulser (N) tillförts utgången. Det krävs i detta fall en lämplig utgång på räknaren och den fungerar därvid som frekvensdelare (divide by N, modulo N-räknare).

Det ställs givetvis olika krav på räknare i olika tillämpningar. I ett digitalt armbandsur räknas frekvensen från en kristalloscillator ner till sekunder, minuter och timmar. I en sådan tillämpning vill man ha en enkel och effektsnål frekvensdelare med givna värden på N. I frekvensmätare

för höga frekvenser önskar man snabba räknare men här spelar effekt-konsumtion och storlek mindre roll.

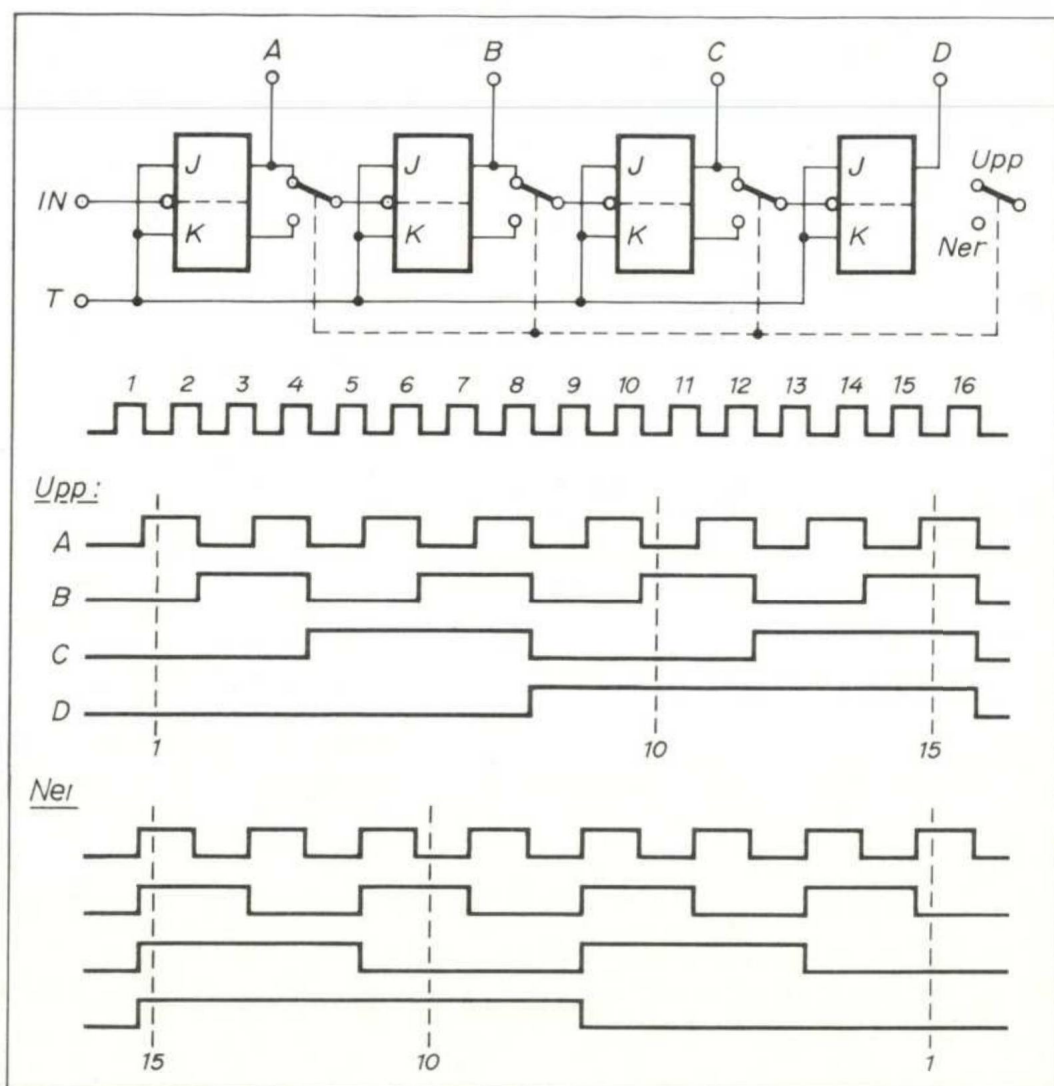
Vi skall nedan enbart syssla med principerna för de vanligaste typerna av räknare: rippelräknare och synkronräknare. Vi skall också se hur skiftregister kan användas som räknare.

Engelska termer: counter (räknare), ripple counter, asynchronous counter, synchronous counter, divide by N counter.

4.1 Rippelräknare

Rippelräknare arbetar asynkront och kallas därför även asynkrona räknare.

I figur 2.23 har vi kopplat fyra JK-vippor i serie. Mellan vipporna sitter omkopplare så att efterföljande vippas klockpulsingång kan anslutas till endera av utgångarna Q eller Q* på föregående steg.



Figur 2.23
Principen för en rippelräknare (vid $T = 1$).

T-ingången är förbunden med samtliga J- och K-ingångar. När T-ingången är hög kommer varje steg att fungera som en frekvenshalverare (modulo-2) och utsignalen på D-utgången kommer alltså att ha en frekvens som är 2^4 ggr lägre än insignalen. Detta gäller oberoende av i vilket läge omkopplarna mellan stegen befinner sig.

Om räknaren är nollställd när T-ingången läggs hög kommer spänningarna på utgångarna A - D att ange räknarens innehåll. Med omkopplarna i läge "UPP" kan antalet inräknade pulser tecknas:

$$S_{UPP} = A \cdot 2^0 + B \cdot 2^1 + C \cdot 2^2 + D \cdot 2^3$$

där A...D kan anta värdena 0 eller 1.

Med omkopplarna i läge "NER" kan antalet pulser tecknas:

$$S_{NER} = 16 - S_{UPP}$$

Tidsdiagrammen i figur 2.23 är spegelvända. Det övre räknar upp ... 2, 3, 4, 5, 6 ... medan det nedre räknar ner ... 6, 5, 4, 3 ...

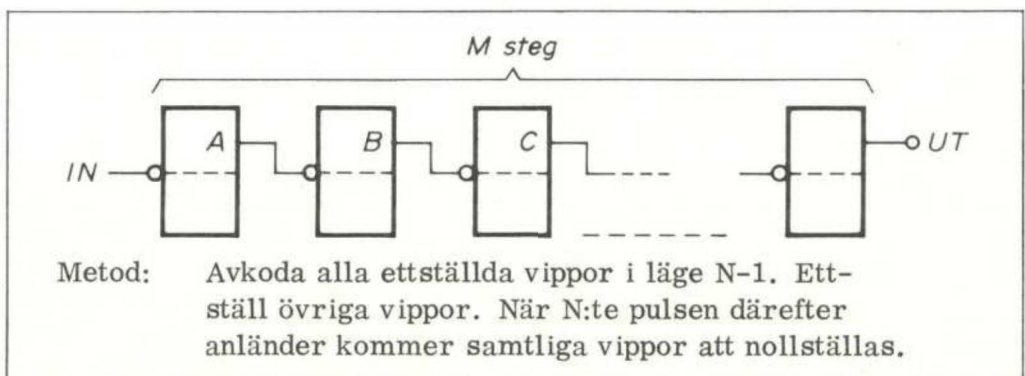
Räknaren i figur 2.23 skulle kunna kallas UPP/NER-räknare (UP/DOWN-counter). Pulser måste i tur och ordning passera alla fyra vipporna likt en krusning (ripple) som breder ut sig genom hela kedjan. Räknare av den typ som visas i figur 2.23 kallas därför rippel-räknare. De är enkelt uppbyggda men också långsamma.

UPP/NER-omkopplingen utförs givetvis i praktiska kretsar av grindar (exempelvis AND/OR-kretsar).

Räknare av den typ vi studerat i figur 2.23 kan byggas ut till ett godtyckligt antal steg (M). Räkncykeln blir alltid en potens av 2 (2^M).

Vill man konstruera en modulo N-räknare där N inte är en potens av 2 kan man göra på följande sätt:

Koppla först upp en rippelräknare med M steg där $2^{M-1} < N < 2^M$. Denna räknare räknar nu "för långt" och måste modifieras så att den återgår till noll efter N klockpulser. Modifieringen kan utföras enligt den metod som visas i figur 2.24.



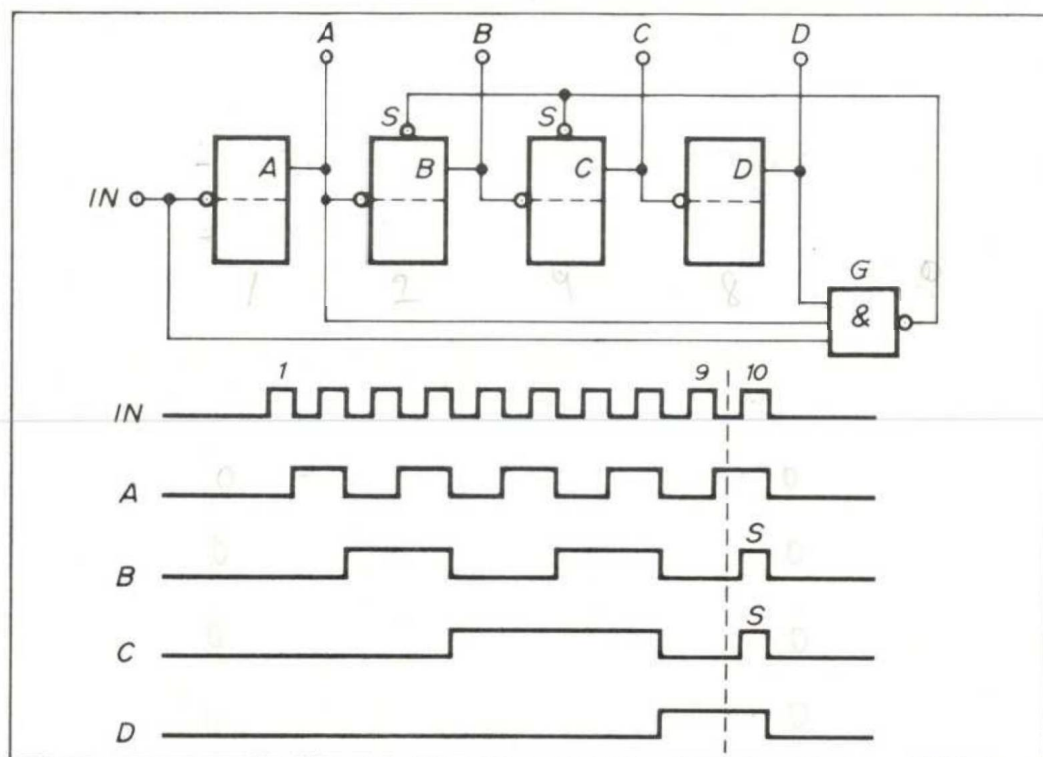
Figur 2.24

Metod för uppbyggnad av en modulo N-räknare av rippeltyp.

Figur 2.25 visar en uppbyggnad enligt figur 2.24. I läge (N - 1), dvs 9, är utgångarna A och D höga:

$$\begin{array}{cccc} A & B & C & D \\ \hline 9 = 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^3 \end{array}$$

Grinden G avkodar när detta läge inträffar samtidigt med hög klocka, dvs när klockpuls 10 anländer. Härvid ettställs övriga vippor, B och C. När klockan efter puls 10 åter går låg kommer samtliga vippor att nollställas.



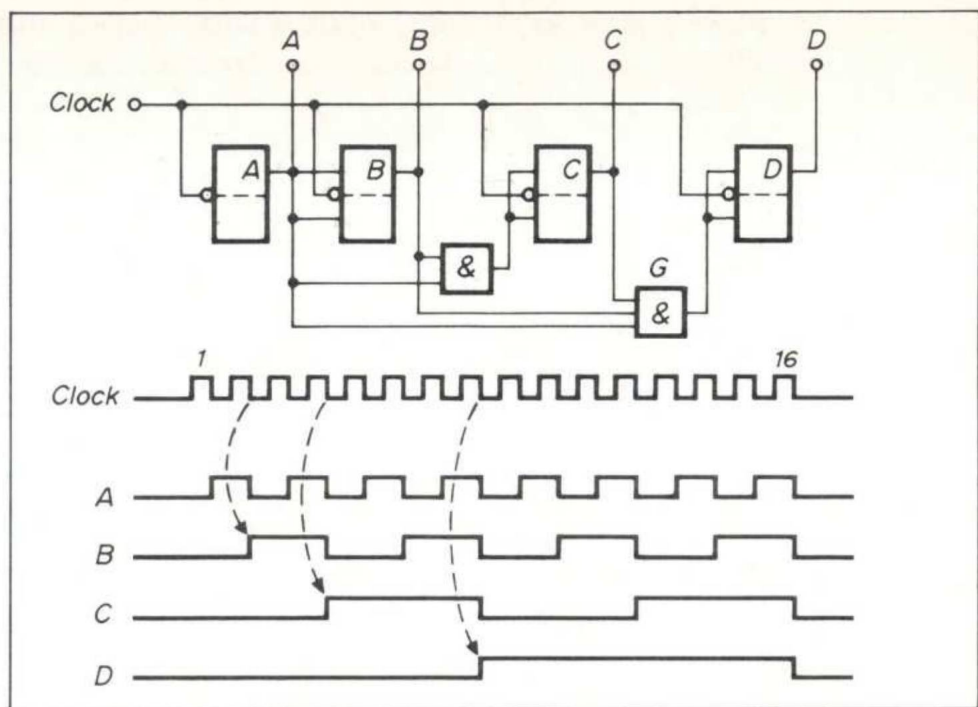
Figur 2.25
Dekadräknare av rippeltyp.

4.2 Synkronräknare

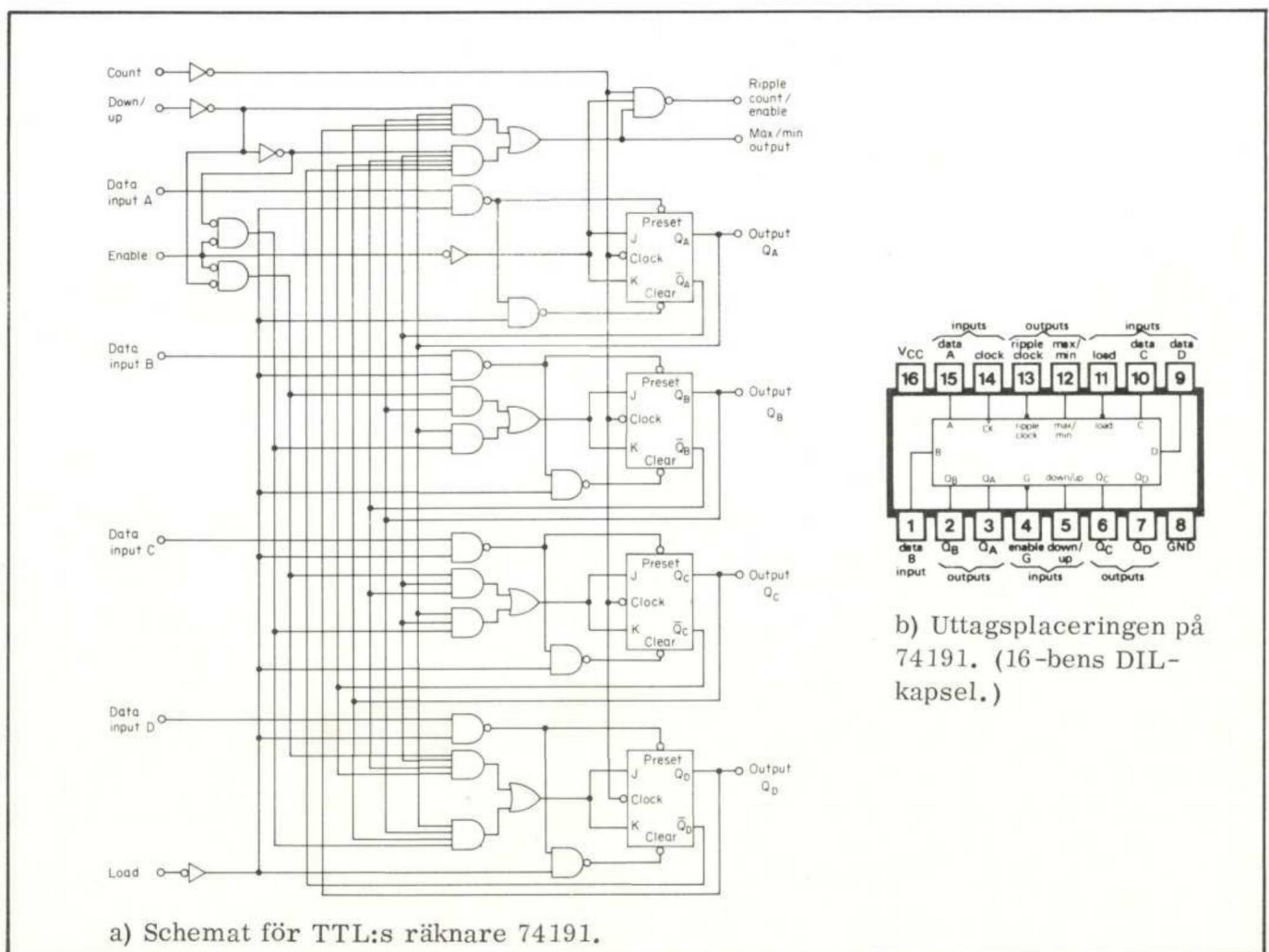
I en rippelräknare får man en fördröjning som blir direkt proportionell mot antalet vippor. Vipporna slår inte om synkront utan signalen "ripp-lar" fram genom räknaren.

Man har därför i stor utsträckning frångått rippelräknare och använder i stället räknare där samtliga vippor ställs synkront av klockpulsen och där vippornas inställning bestäms av grindnät. En synkron räknare får därmed en fördröjning av samma storlek som fördröjningen hos en vippa och en grind. Den större snabbheten får man till priset av komplexa grindnät. (Med IC-kretsar utgör emellertid komplexa grindnät inget hinder.)

Figur 2.26 visar principen för en synkron binärräknare. För att få samma tidsdiagram som för UPP-räknaren i figur 2.23 måste vippa B i figur 2.26 triggas enbart när A ligger hög, vippa C enbart när både A och B ligger höga och vippa D slutligen enbart när samtliga föregående vippor, A, B och C, ligger höga. Alla dessa villkor kan som framgår av figur 2.26 klaras av med hjälp av två OCH-grindar.



Figur 2.26
Principen för en synkron binärräknare uppbyggd av JK-vippor.

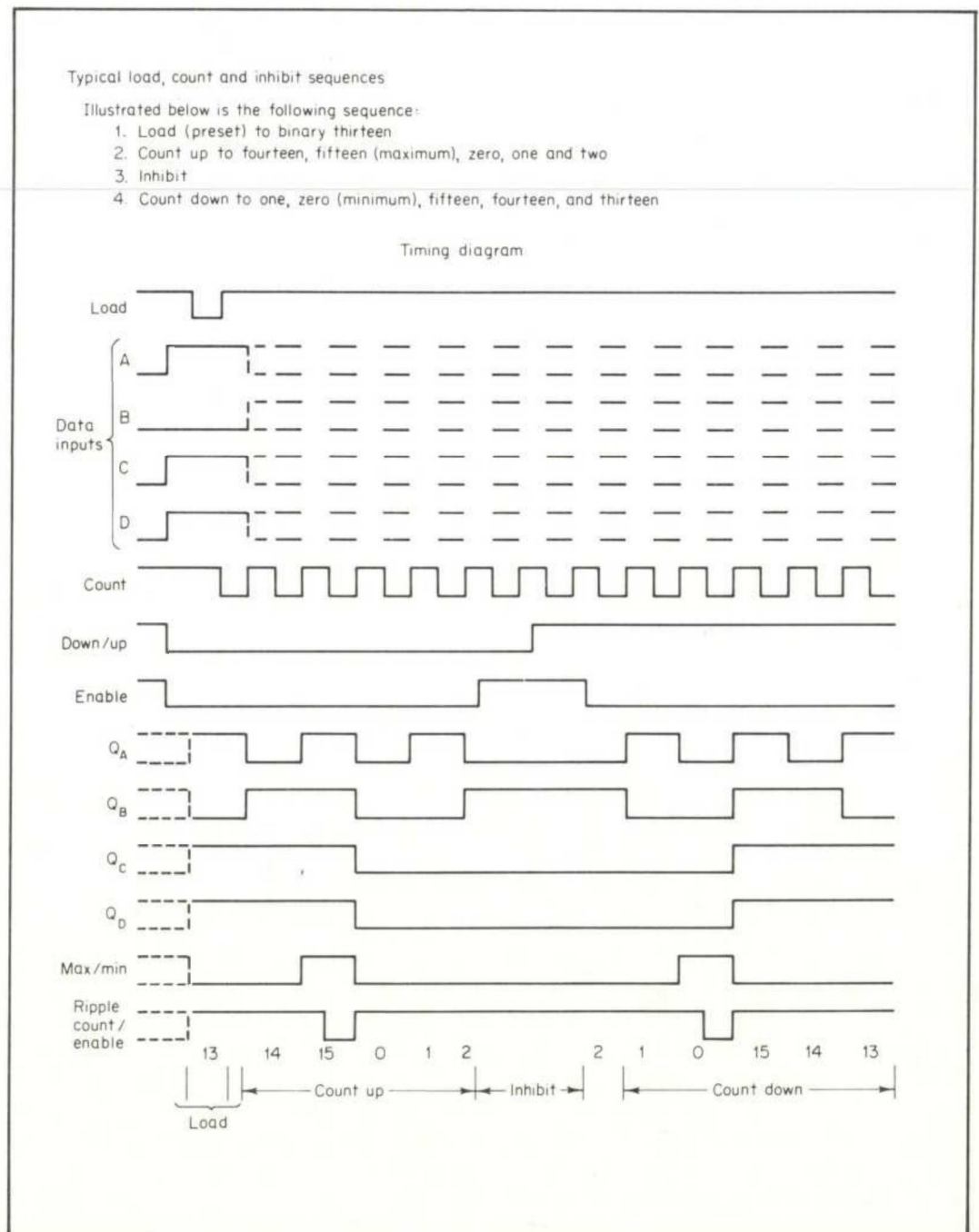


Figur 2.27
Fyra bitars binär synkronräknare. (Texas Instruments.)

Figur 2.27 visar en vanlig 4 bitars binär synkron UPP/NER-räknare i TTL-serien (74191). Räknaren har fyra dataingångar som aktiveras med "load"-ingången. De pulser som skall räknas tillförs räkneingången (count) och "down/up"-ingången bestämmer räkneriktning. Ingången "enable" måste ligga låg för att räknaren ska fungera, precis som T-ingången i figur 2.23 måste ligga hög.

Räknaren har fyra parallella utgångar från vipporna A - D. Dessutom finns en "max/min-output" som vid UPP-räkning ger utsignal när alla vippor ligger höga och vid NER-räkning när alla vippor ligger låga. Rip-pelutgången ger utpuls synkront med räknepulsen (count) när "max/min" ligger hög.

Figur 2.28 visar exempel på tidsdiagram. Enable kallas på svenska fri-signal och inhibit (eller disable) spärrsignal.



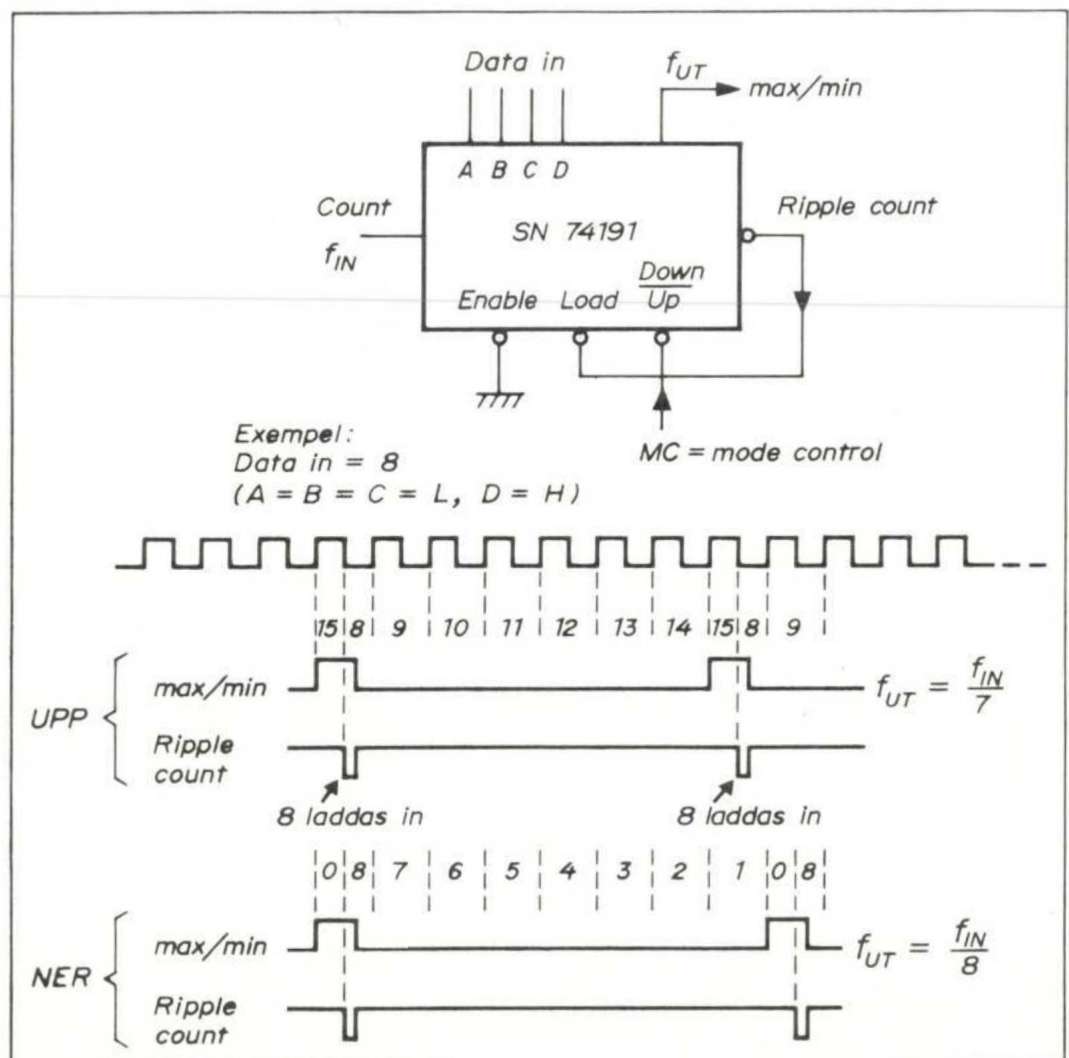
Figur 2.28
Tidsdiagram för 74191. (Texas Instruments.)

Figur 2.29 visar hur elegant man kan utnyttja räknaren 74191 som frekvensdelare. Om dataingångarna A - D ges inställningen X och serieutgången (ripple count) får styra laddningskontrollen (load) kommer max/min-utgången att ge utfrekvenserna

$$f_{\text{UPP}} = \frac{f_{\text{IN}}}{15 - X} \quad (\text{vid UPP-räkning})$$

$$f_{\text{NER}} = \frac{f_{\text{IN}}}{X} \quad (\text{vid NER-räkning})$$

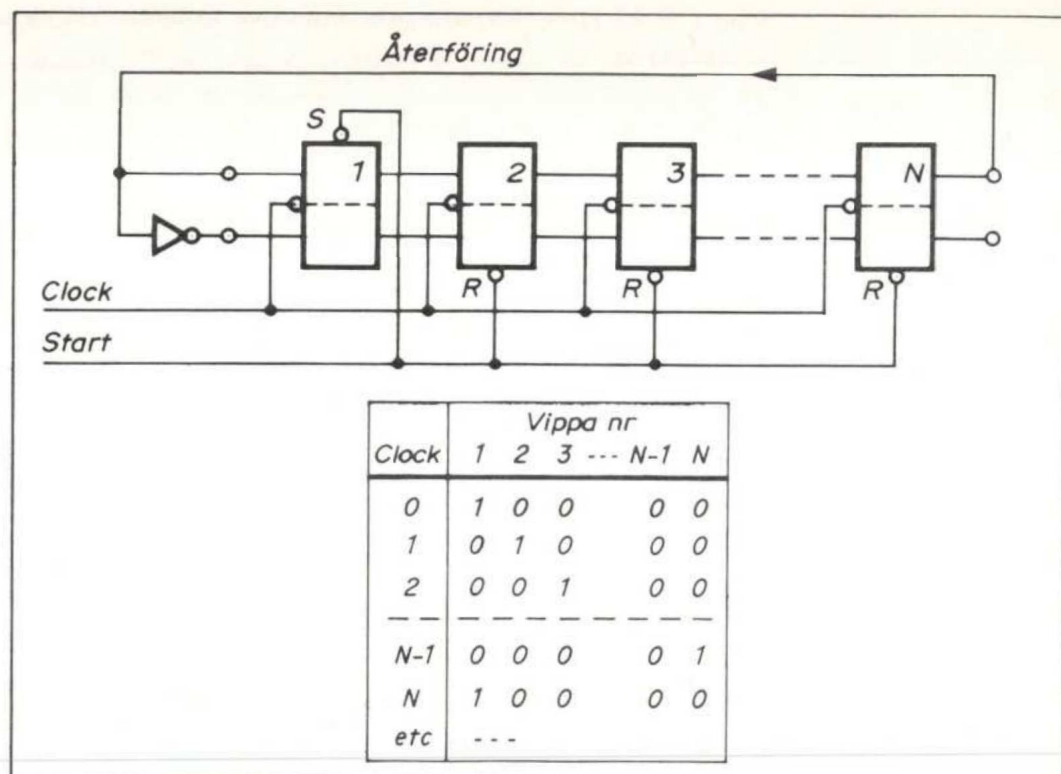
där f_{IN} är ingångsfrekvensen (count).



Figur 2.29
Räknaren 74191 som frekvensdelare. (Data in = 8.)

4.3 Skiftregister som räknare

Om man kopplar samman utgång och ingång på ett skiftregister kommer innehållet i registret att skiftas runt. Om registret innehåller N vippor återkommer man till utgångspunkten efter N skiftpulser. Ett sådant återkopplat skiftregister kan tydligen användas som räknare eller frekvensdelare. Det kallas vanligen ringräknare (ring counter eller walking ring counter).



Figur 2.30
Ringräknare (i princip).

lätt att avkoda

Figur 2.30 visar principen för en ringräknare. Om man som utgångsläge ettställer första vippan men nollställer övriga blir räknarens ställning enkel att avkoda vid inräkning av klockpulser dvs skiftpulser.

Jämfört med de räknare vi tidigare behandlat kräver ringräknaren fler vippor än binärräknaren. Antalet tillstånd hos en räknare med N steg blir:

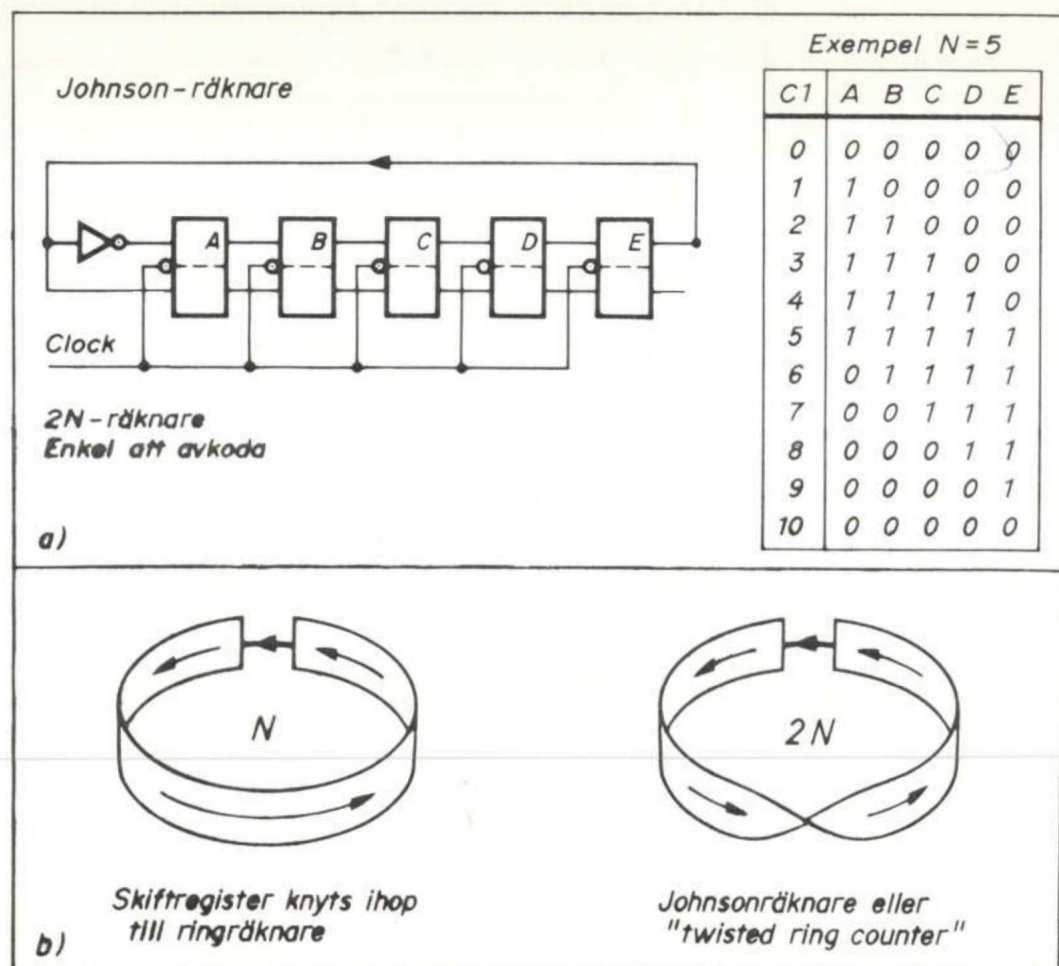
2^N hos binärräknaren
N hos ringräknaren

Ringräknare är alltså lämpliga vid låga N-värden och där krav ställs på enkel avkodning.

Det kan uppstå problem med ringräknare om de kommer in i icke önskade räknecykler. Figur 2.31a visar den önskade räknecykeln hos en fyra bitars ringräknare. Vippornas tillstånd i de fyra stegen i cykeln är tecknade i ett tillståndsdiagram. Siffrorna i varje ring avser utgångarna på vipporna A, B, C och D. Det är som synes endast fyra av registrets 16 möjliga kombinationer som utnyttjas vid den önskade och enkla räknecykeln.

Figur 2.31b visar de övriga tolv tillstånden. Pilarna anger ordningsföljden vid trigging. Två tillstånd 0000 och 1111 är tydligen helt låsta. Kommer man i tillståndet 0101 (eller 1010) får man en räknecykel på 2. De övriga två tillstånden har båda räknecykler med fyra steg.

I praktiken tvingar man normalt in ringräknare i ett önskat starttillstånd så som visats i figur 2.30. Med hjälp av en grind enligt figur 2.32 kan man få ringräknaren självstartande.



Figur 2.33
Johnson-räknare i
princip.

Figur 2.33 visar principen. Genom att invertera den signal som återförs från utgång till ingång i en ringräknare får man räknecykeln $2N$, jämfört med N hos den vanliga ringräknaren.

Samtliga tillstånd hos en Johnson-räknare kan avkodas med enkla OCH-grindar med två ingångar.

Repetitionsfrågor

1. Hur ser talet 213_{10} ut om det skrivs i binär, oktal eller hexadecimal form eller i BCD-kod?
2. Vad är innebörden i förkortningarna SISO, SIPO, PIPO och PISO?
3. Varför ställs det olika krav på vipporna i ett PIPO- och ett SISO-register?
4. Vad karakteriserar en "asynkron RS(H)-vippa"?
5. Vilka två huvudtyper av trigging förekommer i IC-vippor?
6. Vad är skillnaden mellan en RS- och en JK-vippa?
7. Vad är typiskt för en JK-vippa och för en MS-vippa?
8. Vad menas med sättid och hålltid?
9. Skissera ett fyra bitars skiftregister av SISO-typ uppbyggt av D-vippor.
10. Hur kan man modifiera en fyra bitars rippelräknare så att den blir en dekadräknare?

D3 TTL-kretsar

TTL-kretsar 58

1 Varför TTL 58

- 1.1 Sink och source 58
- 1.2 DTL och TTL 59

2 Typgrinden 7400 64

- 2.1 Överföringsdiagrammet 64
- 2.2 Strömspikar 66
- 2.3 Förlusteffekten 68
- 2.4 Stegfördröjningen 69

3 Typgrindens specifikation 70

- 3.1 TTL-familjens nivåer 70
- 3.2 TTL-kretsens belastbarhet 72
- 3.3 TTL-kretsens störmarginaler 73

4 TTL-familjens stamträd 75

- 4.1 High speed och low power 77
- 4.2 Schottky och low power Schottky 78
- 4.3 Antalet typer i TTL-familjen 82

5 Buss- och linjekretsar 83

- 5.1 Krav på grindar för bussmatning 83
- 5.2 Öppen kollektor 84
- 5.3 TRI-STATE 85
- 5.4 Line drivers och line receivers 86

Repetitionsfrågor 87

Målruta

När du läst detta kapitel skall du kunna:

- ange 7400-grindens specificerade spänningsnivåer på ut- och ingång
- ange 7400-grindens belastbarhet, strömsänkingsförmåga och garanterade störmarginaler
- förklara sambandet mellan stegfördröjning och förlusteffekt hos 74, 74H och 74L
- förklara varför 74S är snabbare än 74
- beskriva TTL-familjens stamträd
- beräkna förlusteffekten vid given frekvens och given kapacitiv belastning hos standard TTL-krets

TTL-kretsar

Bland bipolära digitala kretsar är TTL-kretsarna de utan jämförelse vanligaste. Vi skall börja detta kapitel med att förklara bakgrunden till TTL-kretsarnas popularitet.

Därefter skall vi mer i detalj studera funktionen hos typgrinden 7400 i TTL-serien. TTL-kretsarna har mycket väl specificerade data och vi skall gå igenom typgrindens viktigaste data.

För att möta olika krav på snabbhet och effektkonsumtion har TTL-familjen utvidgats till ett stamträd som nu omfattar flera grenar och vi skall bekanta oss med några av dessa.

Utvecklingen går mot integrering av allt större enheter och dessa enheter kommunicerar ofta med omvärlden via bussledningar. Linjekretsar ("line drivers" och "line receivers") blir därvid viktiga byggstenar. Vi skall därför avsluta detta kapitel med en liten översikt över de vanligaste typerna av linjekretsar.

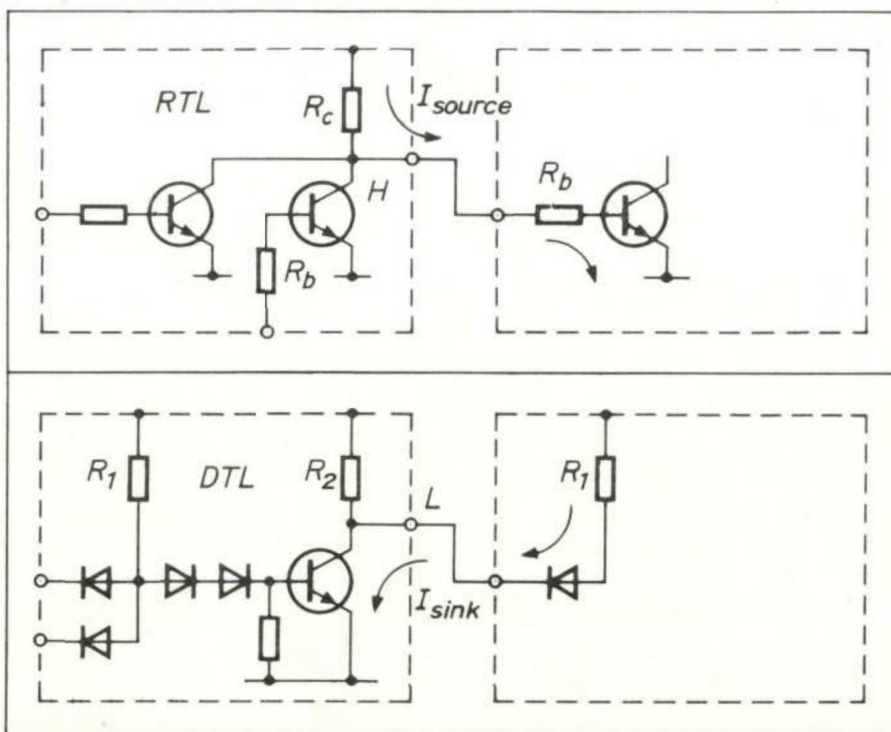
1 Varför TTL

Det är många faktorer som bidragit till TTL-familjens stora popularitet. Ingenjörerna var vana vid DTL-logik i diskret uppbyggnad och TTL-logik fungerar (utifrån sett) helt likartat.

Sänka

1.1 Sink och source = *make*

Man säger att TTL liksom DTL är en strömsänkande logik (sinking logic) medan exempelvis RTL är en strömmatande logik (sourcing logic) figur 3.1.



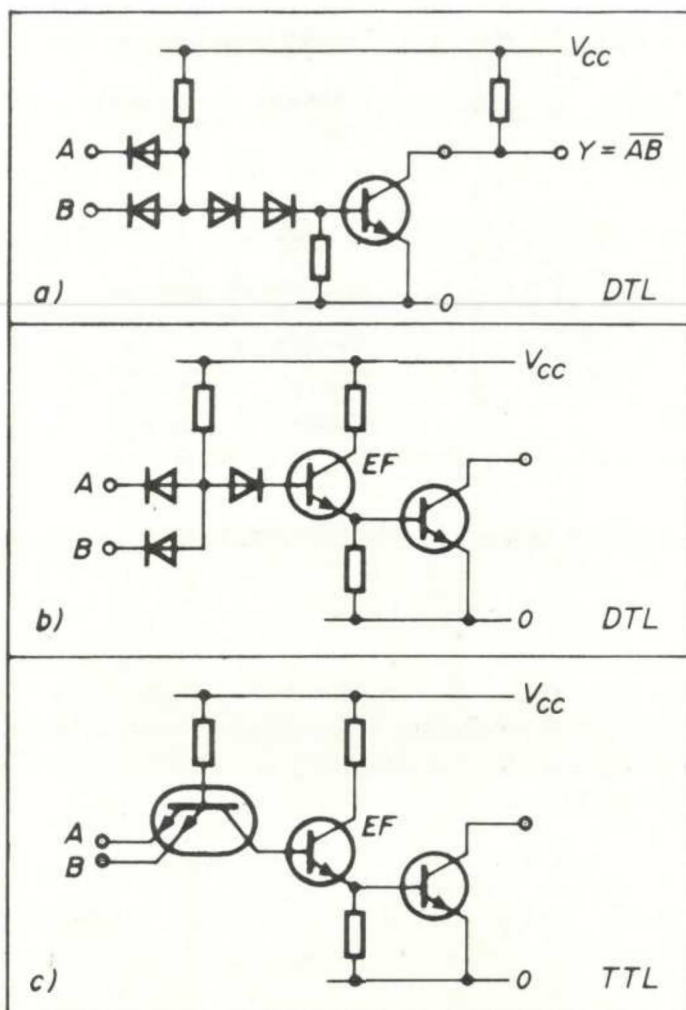
Figur 3.1

Exempel på "source" (RTL) och "sink" (DTL).

DTL-kretsen känner den hårdaste belastningen när utgången är låg och inverterartransistorn måste då "sänka" strömmen från belastande kretsar. RTL-kretsen känner däremot den hårdaste belastningen när utgången är hög och den måste då mata ("source") belastande kretsar.

1.2 DTL och TTL

DTL och TTL är mycket närbesläktade. Figur 3.2a visar den DTL-grind som vi tidigare diskuterat, figur 1.8 och figur 1.33. Grinden i figur 3.2a är utan kollektorresistor (open collector). Resistorn som utritats mellan V_{CC} och utgången är här en yttre resistor.

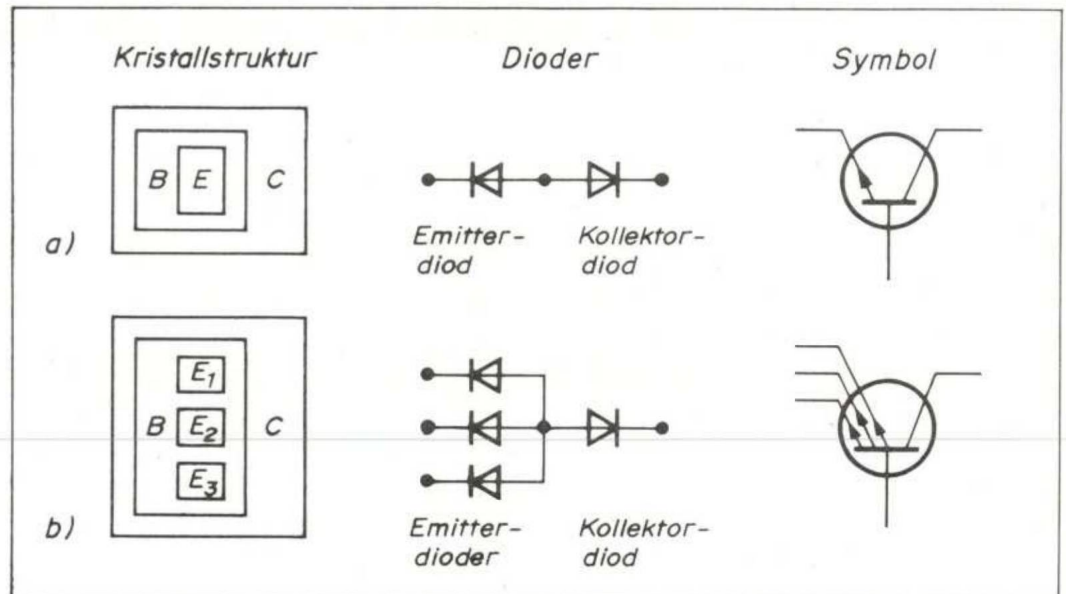


Figur 3.2

- a) DTL-grind (open collector).
- b) DTL-grind med emitterföljare (EF).
- c) TTL-grind (open collector).

För att få bättre drivning till inverterartransistorn, dvs för att kunna "sänka" större ström (eller tåla större belastning) har man i kretsen i figur 3.2b ersatt en av dioderna med en emitterföljare EF. Överföringskurvan blir fortfarande densamma, figur 1.33, eftersom U_{BE} hos transistoren EF är lika stor som framspänningsfallet hos den tidigare dioden. Om EF har strömförstärkningsfaktorn B blir belastbarheten (fan out, output loading capability) eller strömsänkingsförmågan hos grinden i figur 3.2b bortåt B gånger större än hos grinden i figur 3.2a.

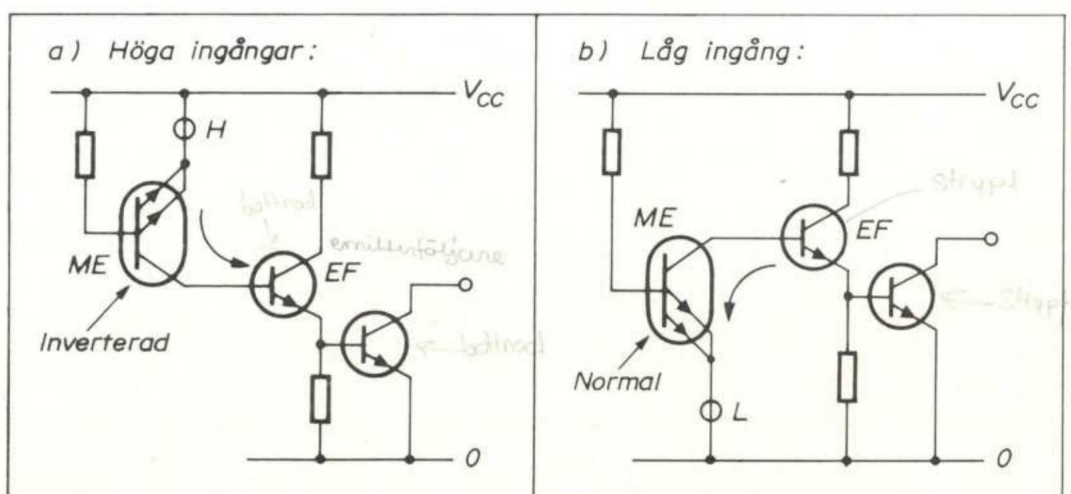
I figur 3.2c har vi ersatt de resterande tre dioderna med en multiemittertransistor. Figur 3.2b och c ser kanske vid första ögonkastet helt olika ut men i princip är skillnaden inte stor. En transistor innehåller ju två dioder, en emitterdiod och en kollektordiod. Genom att diffundera flera emitterar i samma basområde får man en multiemittertransistor, figur 3.3. Vi har för övrigt redan i figur 1.22 sett en kristallstruktur för en transistor med två emitterar som indiffunderats i samma basområde.



Figur 3.3

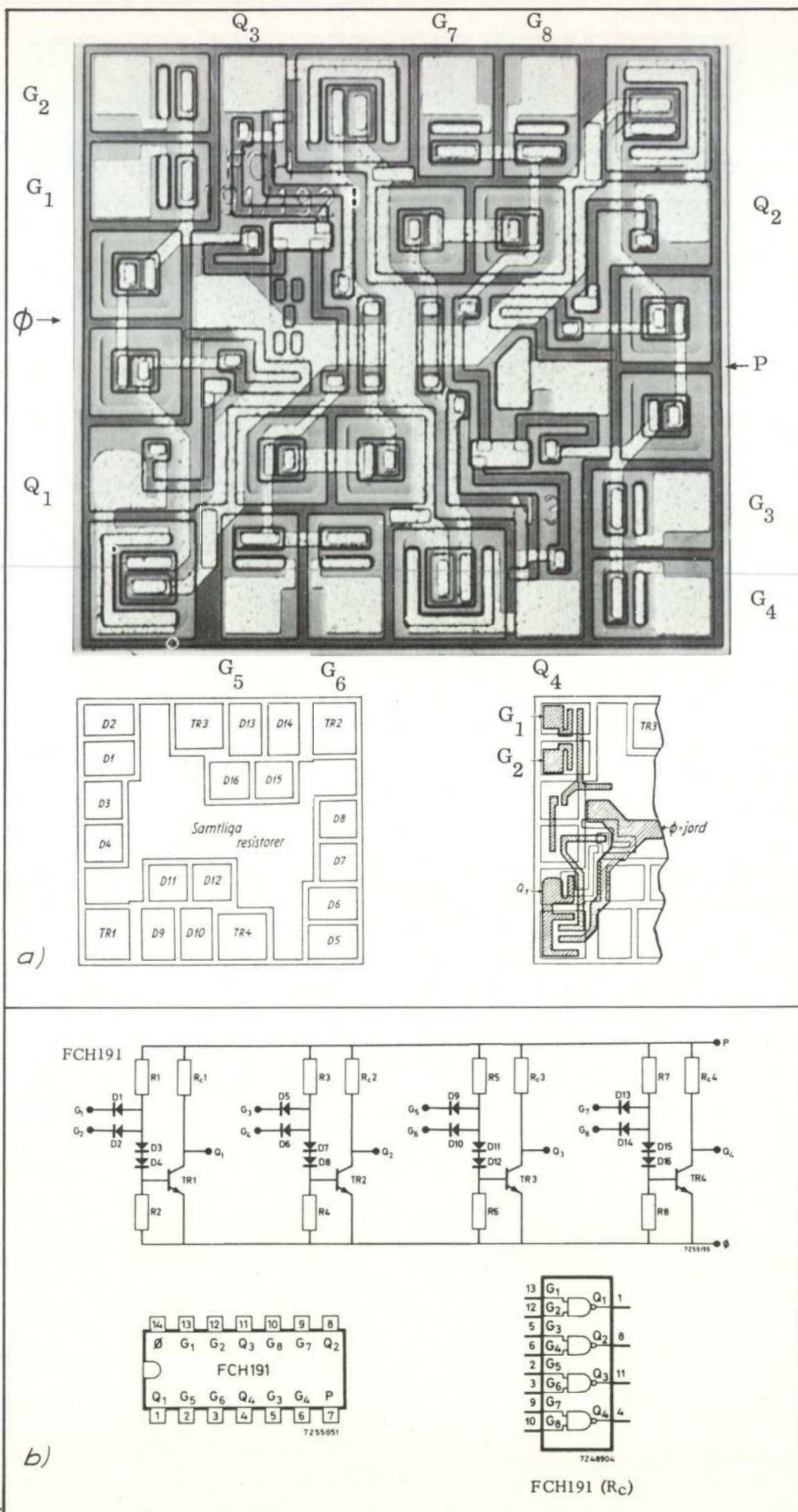
a) Transistorns emitter- och kollektordioder. b) Multiemittertransistorn.

Det är flera fördelar med multiemittertransistorn. Figur 3.4a visar, med schemat omritat, hur TTL-grindens multiemittertransistor fungerar när samtliga ingångar är höga (H) och figur 3.4b när minst en av ingångarna är låg (L).



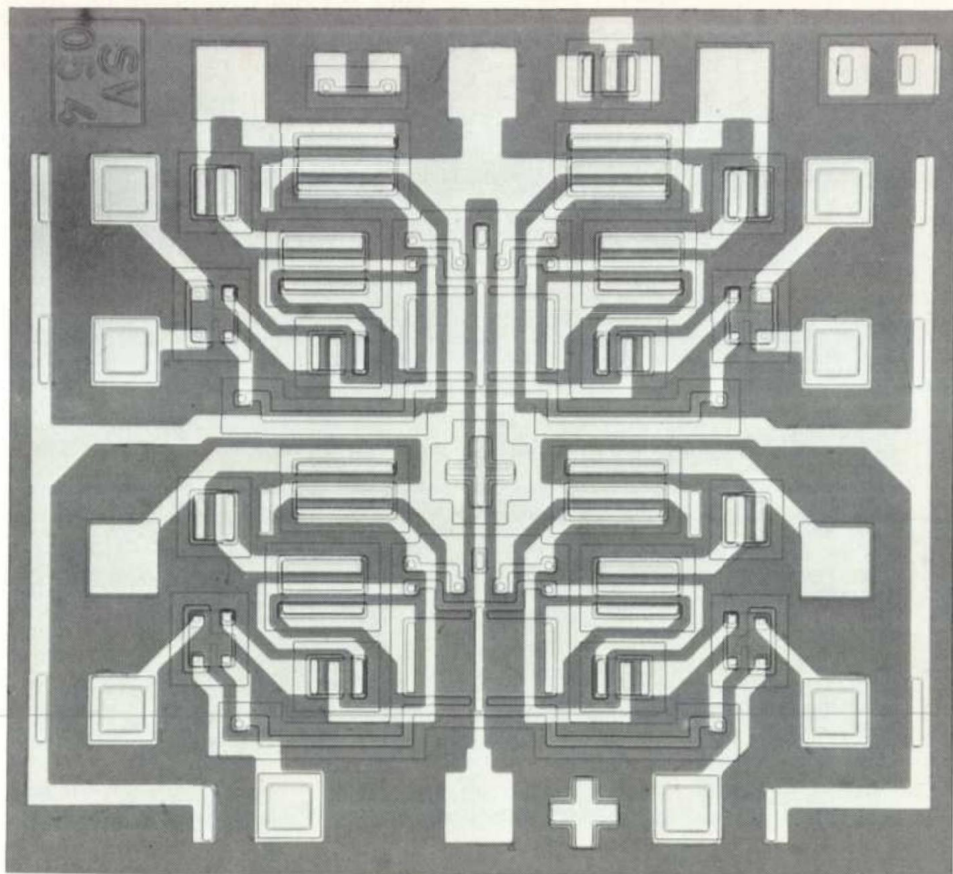
Figur 3.4

Multiemittertransistorns funktion vid hög (H) och låg (L) ingång.

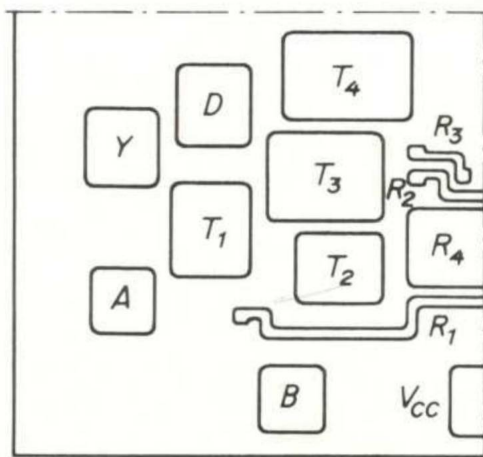


Figur 3.5
a) Kristallfoto på kretsen FCH 191 (Philips).
b) FCH 191 och dess uttagsschema (Philips).

a)

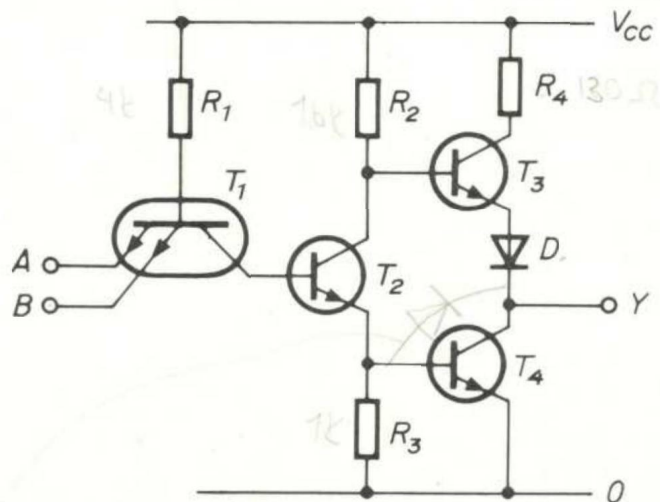
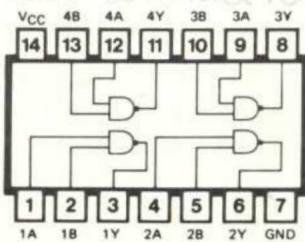


tolempålen
gör att omslaget
sker snabbare



här kan en diod läggas
för att T_4 ej ska ligga
värt bottnack, dvs. omslaget
blir snabbare.

b)



Genom att T_3 leder blir R_{tot}
mindre $\Rightarrow$ den driver och
sänker V_{in} .

D Dioden med för att T_3 ska vara
säkert släppt

Figur 3.6

a) Kristallfoto av 7400 (Siemens).

b) Komponenterna i en av grindarna samt kretsschema.

Med samtliga ingångar höga fungerar multiemittertransistorn som en inverterad transistor, dvs med emittern fungerande som kollektor och kollektorn som emitter. Den inverterade multiemittertransistorn ME ger därvid stor ström och snabbt omslag av emitterföljaren EF, som går från strypt till ledande tillstånd.

Med en ingång låg (L) fungerar multiemittertransistorn som en vanlig transistorswitch som bottnas och därvid snabbt stryper EF. De emitterar som eventuellt ligger höga får här backspänning och påverkar inte förloppet.

Multiemittertransistorns funktion vid till- och frånslag gör att TTL-logiken blir mer än dubbelt så snabb som DTL-logiken.

År 1964 när TTL-logiken introducerades var kanske snabbheten inte lika väsentlig som kristallarean. Liten area på kristallen betydde bättre "yield", dvs större andel fungerande kretsar vid produktionen och därmed lägre pris. Figurerna 3.5 och 3.6 visar kristallstrukturer för typiska DTL- och TTL-kretsar. Man ser här att grinddioderna på ingången till DTL-kretsen tar stort utrymme i anspråk jämfört med TTL-logikens multiemittertransistor.

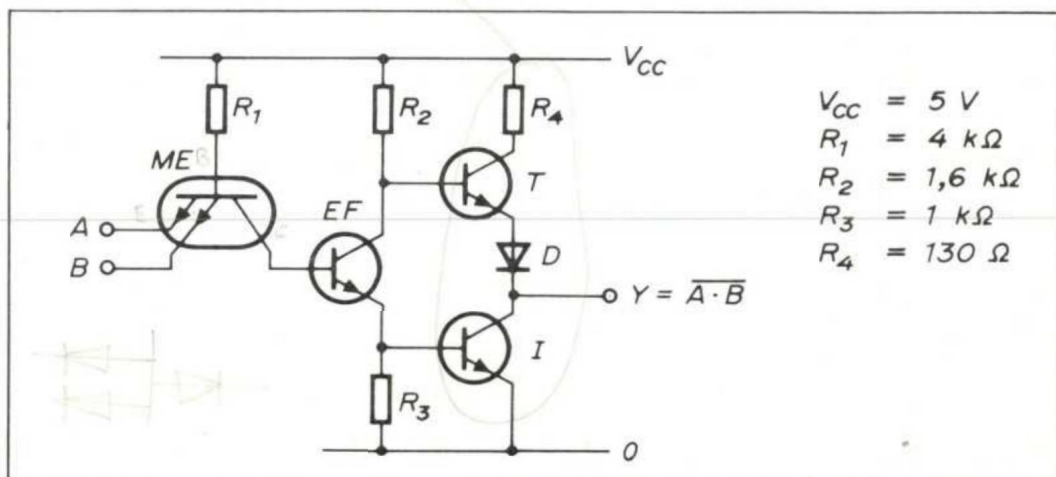
Ytterligare en bidragande faktor till TTL-familjens stora framgång är det enhetliga och praktiska sätt på vilket TTL-kretsarna specificeras. Men till TTL-grindens specifikation skall vi återkomma senare.

Snabbhet, liten area, lågt pris och bra specifikationer är alltså några av orsakerna till TTL-familjens framgångar.

2 Typgrinden 7400

Figur 3.7 visar kretsschemat för typgrinden 7400. Förutom multiemittertransistorn ME, emitterföljaren EF och inverterartransistorn I innehåller typgrinden även en transistor T och en diod D. De bildar tillsammans med I en sk totempåle (totem-pole output). Då utgången är hög fungerar T som en emitterföljare och D leder i framriktning. Vid hög utgång kan alltså T mata ström (source) till belastningen. På detta sätt kan utgången både sänka och driva ström och man får snabba omslag i båda riktningarna.

Grindar med totempåle på utgången kan inte trådas (wired AND) och därför finns det även en variant med öppen kollektor, dvs utan T och D i figur 3.7. Den heter 7401.



Figur 3.7
Typgrinden 7400.

2.1 Överföringsdiagrammet

Man kan beskriva en grinds funktion med hjälp av olika diagram. Överföringsdiagrammet är härvid ett av de viktigaste. Figur 3.8 visar överföringsdiagrammet för typgrinden 7400 i TTL-familjen.

Det är tre områden i V_O/V_I -diagrammet som är av intresse. När V_I är låg är utgången V_O hög (H). När V_I är hög är utgången V_O låg (L). Mellan dessa områden ligger omslagsområdet.

2.1.1 Låg ingång ($V_I = L$)

Vid låg ingång bottnar ME i normal arbetsriktning och EF stryps. Inverterartransistorn får därmed ingen basdrivning och stryps, figur 3.9a. Spänningen V_{OH} (O = output, H = hög) kan uppskattas på följande sätt:

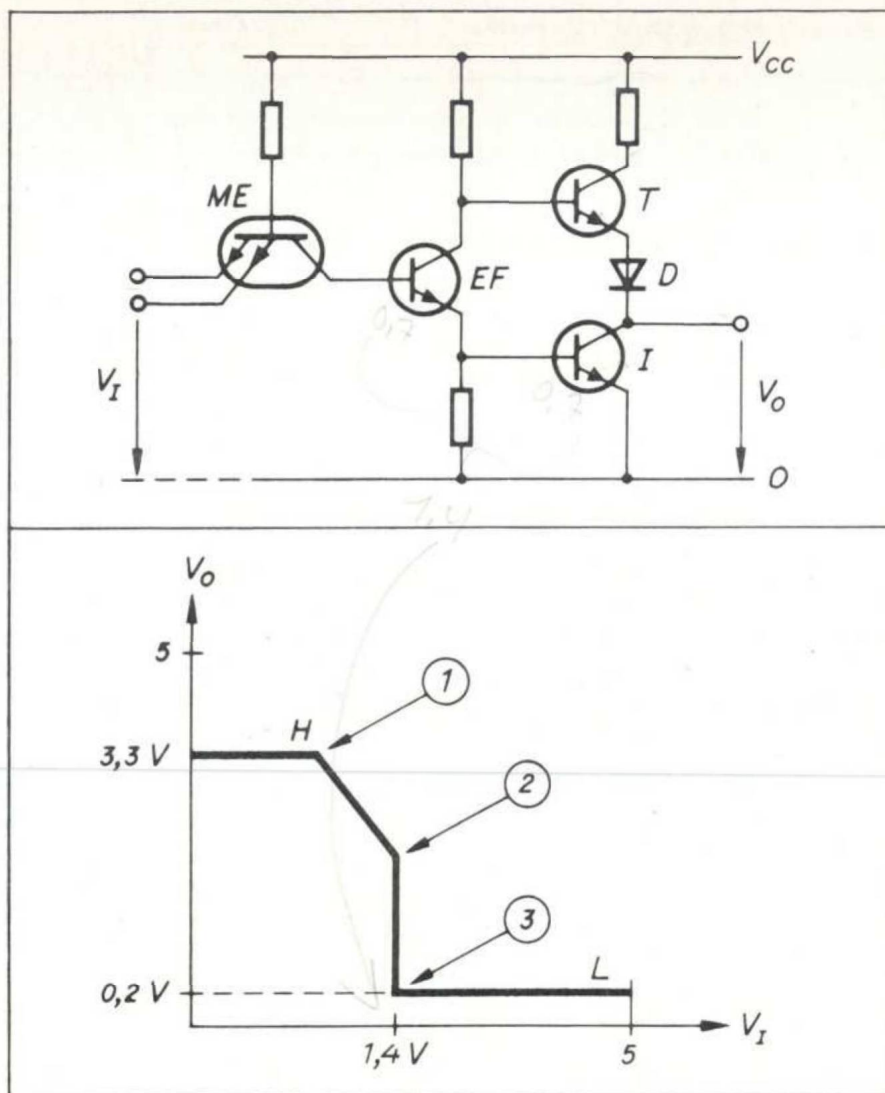
$$V_{OH} = V_{CC} - U_D - U_{BE} - U_R \approx (5 - 0,7 - 0,7 - 0,3) \text{ V} = 3,3 \text{ V}$$

där U_D = spänningsfall över D

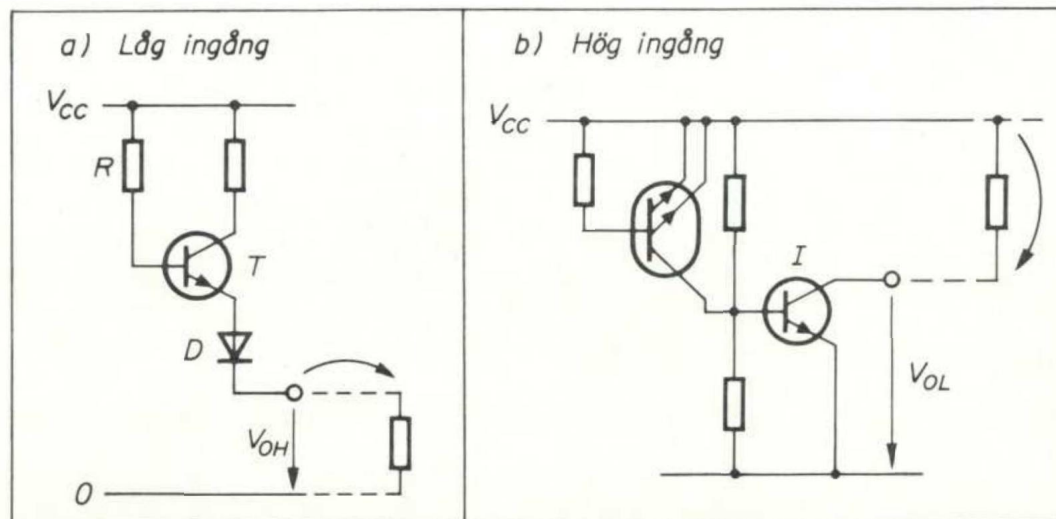
U_{BE} = bas/emitter-spänning hos T

U_R = spänningsfall över R

$V_{CC} = 5 \text{ V}$ = nominell matningsspänning hos TTL-familjen.



Figur 3.8
Överföringsdiagrammet för TTL-kretsen 7400.



Figur 3.9
TTL-grindens utspänning vid låg och hög ingång.

2.1.2 Hög ingång ($V_I = H$)

Vid hög ingång på alla ingångar kommer ME att arbeta inverterad och driva EF i bottning. Inverterartransistorn bottnar medan T stryps. (Dioden D utgör härvid en marginal för säker strypning av T.) Utspänningen V_{OL} blir i detta fall bottenpotentialen U_{CEsat} hos inverterartransistorn I.

$$V_{OL} = U_{CEsat} \approx 0,2 \text{ V}$$

2.1.3 Omslagsområdet

Omslagsområdet har tre intressanta punkter. Om vi ökar inspänningen V_I från ett lågt värde ligger grinden med hög utgång ända tills V_I når punkten ①.

Här börjar EF leda (i aktiva området) och T följer nu dess utspänning som en vanlig emitterföljare. EF, R_2 och T fungerar här som ett förstärkarsteg med förstärkningen

$$\frac{\Delta V_O}{\Delta V_I} \approx -\frac{R_2}{R_3} = -1,6 \text{ ggr}$$

Om vi ökar inspänningen uppnår vi snart punkten ② där I börjar leda. Mellan punkterna ② och ③ är både EF och T i aktiva området. Kretsen har hög förstärkning och kurvan faller brant.

Kretsen är inte stabil när inspänningen befinner sig på värden mellan punkterna ① och ③. En grind skall därför aldrig matas med för långsamma flanker på V_I . Normala värden på stig- och falltider rör sig om ca 10 ns.

Själva omslaget, ② till ③, sker när inspänningen ungefär har samma storlek som framspänningen hos två seriekopplade dioder, dvs $V_I \approx 1,4 \text{ V}$.

2.2 Strömspikar

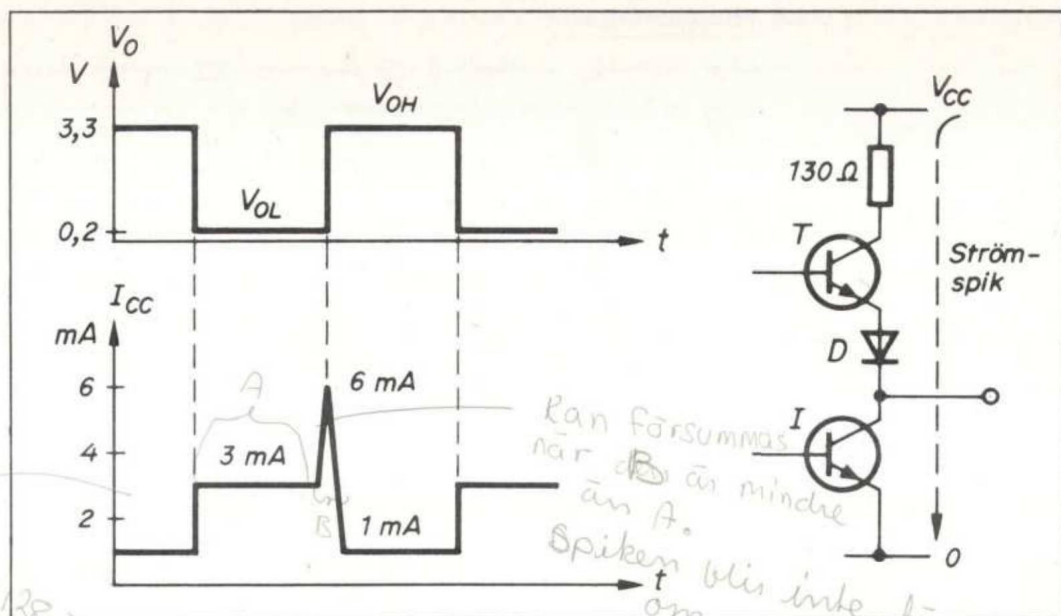
Karakteristiskt för TTL-familjen är den strömspik som uppstår vid omslag, ② till ③ i figur 3.8. Under ett kort ögonblick leder både inverteraren I och transistorn T, figur 3.10. Strömspikens amplitud begränsas till ett rimligt värde av serieresistorn.

Området där strömspiken inträffar är inte stabilt och det är övervägande dynamiska fenomen som orsakar strömspiken. Det är enbart när utgången går hög som strömspiken inträffar och detta beror på att inverterartransistorn ligger hårt bottnad och därför blir långsam. Den leder (efter ledning) ett kort ögonblick efter det att transistorn T snabbt börjat leda. Det är snarare en laddningsspik än en strömspik. Hos typgrinden 7400 uppgår strömspikens amplitud till ca 3 mA under 15 ns, medan den snabba Schottkygrinden drar 13 mA under 5 ns. Det är alltså frågan om en laddning av storleksordningen

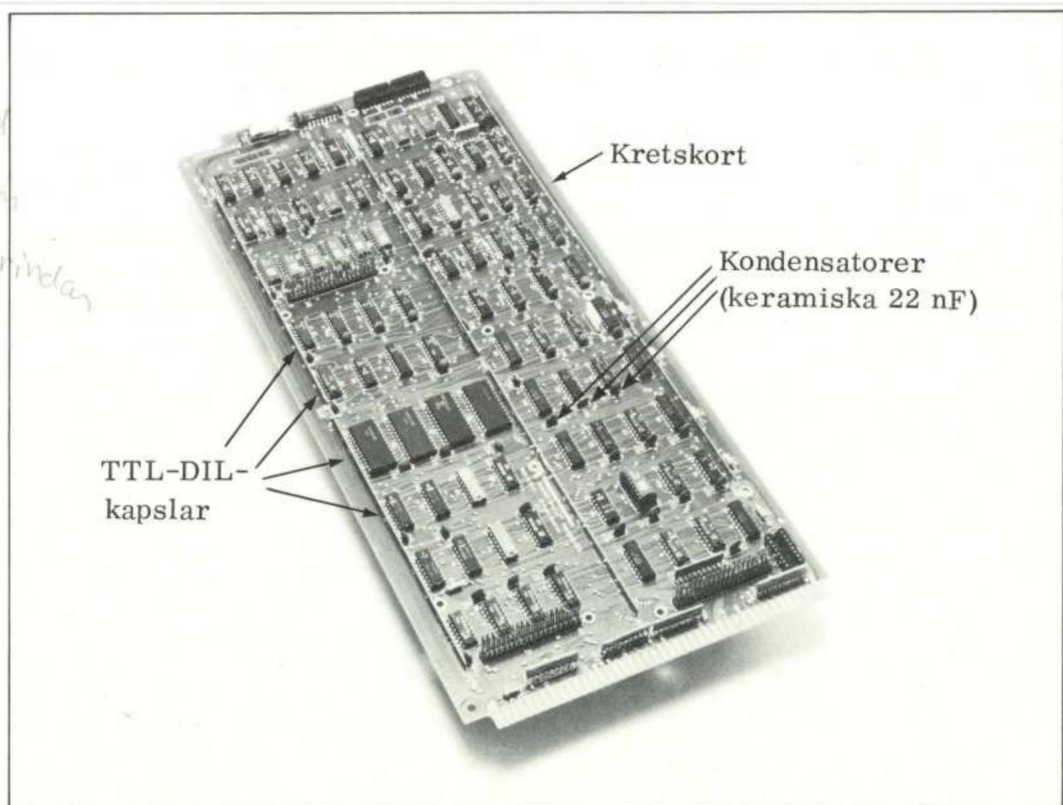
$$Q = I \cdot t \approx 50 \text{ pC}$$

som vid varje omslag (i riktning $V_{OL} \rightarrow V_{OH}$) pumpas igenom grinden.

Strömspikarna i en TTL-grind kan förorsaka störningar och därför avkopplas normalt varje krets med en kondensator av högfrekvenstyp, exempelvis med en keramisk plattkondensator på 22 nF. Figur 3.11 visar ett typiskt kretskort med TTL-kretsar och deras avkopplingskondensatorer.



Figur 3.10
Strömspik vid omslag hos TTL-kretsen 7400.



Figur 3.11
Avkoppling av strömspikar på kretskort med TTL-kretsar (Scandia-Metric).

En laddning $Q (= I \cdot t)$ som pumpas in i en krets från en generator med spänningen U tillför kretsen en energi

$$W = Q \cdot U = U \cdot I \cdot t \text{ Js}$$

Spänningsspikarna medför alltså inte enbart risk för störningar utan dessutom en förlusteffekt som vid frekvensen f uppgår till

$$P_f = Q \cdot U \cdot f$$

2.3 Förlusteffekten

Som framgår av figur 3.10 drar en TTL-grind (7400) strömmen 3 mA vid V_{OL} och strömmen 1 mA vid V_{OH} .

Förlusteffekten i en kapsel innehållande fyra grindar (7400) kan alltså variera mellan

$$P_{fL} = 4 \times 3 \times 5 = 60 \text{ mW}$$

om alla grindarna har låg utgång och

$$P_{fH} = 4 \times 1 \times 5 = 20 \text{ mW}$$

om alla grindarna har hög utgång.

Om alla grindarna i en 7400-krets matas med fyrkantvåg med frekvensen f Hz kommer kretsen att obelastad dra förlusteffekten

$$P_f = \frac{P_{fL} + P_{fH}}{2} + 4 \cdot V_{CC} \cdot Q \cdot f$$

medel- förlusteffekt av
effekt strömspikar i
fyra grindar

Med $f = 10 \text{ MHz}$ och $Q = 50 \text{ pC}$ får vi siffervärdena

$$P_f = \left(\frac{60 + 20}{2} \cdot 10^{-3} + 4 \cdot 5 \cdot 50 \cdot 10^{-12} \cdot 10 \cdot 10^6 \right) \text{ W} = 40 + 10 \text{ mW}$$

Vid höga frekvenser måste vi alltså ta hänsyn även till den förlusteffekt som strömspikarna förorsakar vid omslag. Här tillkommer också den förlusteffekt som belastningen medför. Speciellt medför kapacitiv belastning en icke försumbar förlusteffekt.

En kapacitiv belastning av storleken C upptar vid uppladdning med spänningen ΔU laddningen $Q = \Delta U \cdot C$. Laddningen kan tecknas

$$Q = \Delta U \cdot C$$

där $Q = I \cdot t$.

Den från matningsspänningen tillförda energin blir därvid

$$W = V_{CC} \cdot \Delta U \cdot C = V_{CC} \cdot Q = V_{CC} \cdot I \cdot t$$

och förlusteffekten i en grind belastad med kapacitansen C kan därmed tecknas

$$P_f = V_{CC} \cdot \Delta U \cdot C \cdot f$$

Eftersom utgångsspänningen ändras från 0,3 till 3,3 V (approximativt) blir $\Delta U = 3 \text{ V}$.

En TTL-kapsel innehållande fyra grindar som vardera är belastade med $C = 40 \text{ pF}$ och arbetar med frekvensen $f = 10 \text{ MHz}$ får tydligen en extra förlusteffekt av den kapacitiva lasten av storleken

$$P_f = 4 \cdot V_{CC} \cdot \Delta U \cdot C \cdot f = 24 \text{ mW}$$

Förlusteffekten i en TTL-kapsel beror alltså av flera faktorer, bl a kapacitiv last och den frekvens vid vilken kretsen arbetar. I vårt fall kan tydligen den totala förlusteffekten (för fyra grindar i en kapsel) tecknas

$$P_f = \underbrace{\frac{P_{fL} + P_{fH}}{2}}_{\text{DC-förlust}} + \underbrace{4 \cdot V_{CC} \cdot Q \cdot f}_{\text{spik-förlust}} + \underbrace{4 \cdot V_{CC} \cdot \Delta U \cdot C \cdot f}_{\text{kapacitiv-belastning}}$$

där $\Delta U = 3 \text{ V}$.

proportionell mot förlustfrekvensen för att det kommer vara

2.4 Stegfördröjningen

Till- och frånslag hos en transistorswitch tar, som vi sett tidigare, viss tid i anspråk. Laddningen i transistorer, dioder och anslutna kretsar måste omfördelas innan omslaget sker och snabbheten begränsas därmed av storleken hos de strömmar med vilka omladdningen sker.

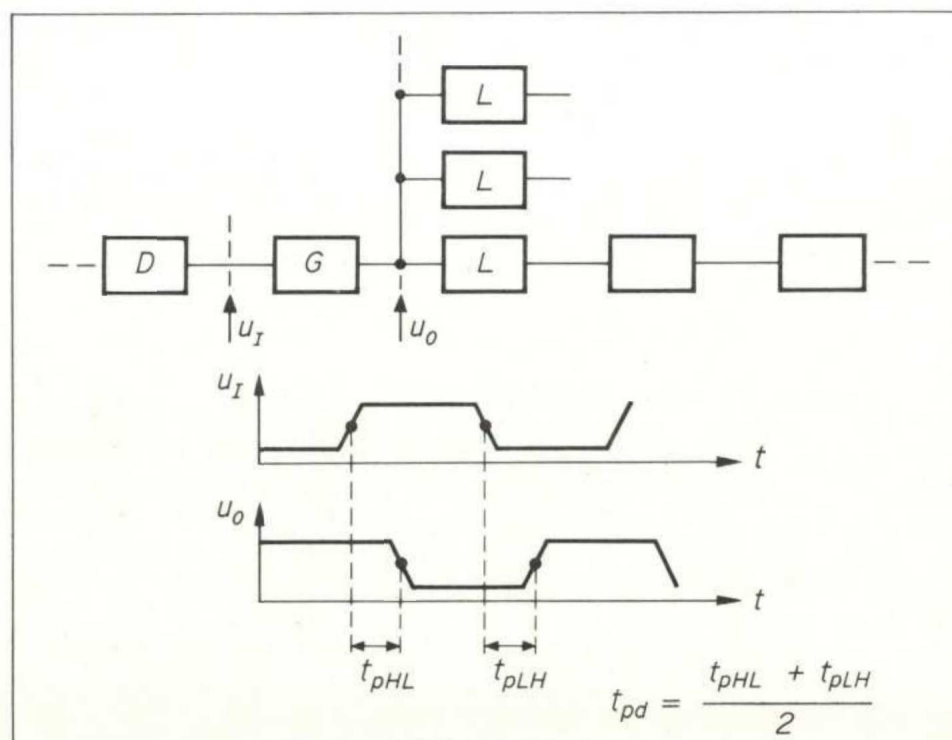
En grind som dimensionerats för att dra liten förlusteffekt måste ha höga resistansvärden (dvs små strömmar) och blir därför långsam. En krets som dimensionerats för att vara snabb måste ha höga omladdningsströmmar (dvs låga resistansvärden) och den drar därför som regel hög effekt.

TTL-grinden 7400 är dimensionerad för en förlusteffekt på ca 10 mW per grind och omslaget tar därvid ca 10 ns.

Grindens snabbhet beror på belastningen. I figur 3.12 matas grinden G från grinden D och belastas av grindarna L (D = drive, L = load). Vi ser i figuren hur vi får en viss fördröjning t_{pLH} när G:s utgång går hög och en annan fördröjning t_{pHL} när G:s utgång går låg.

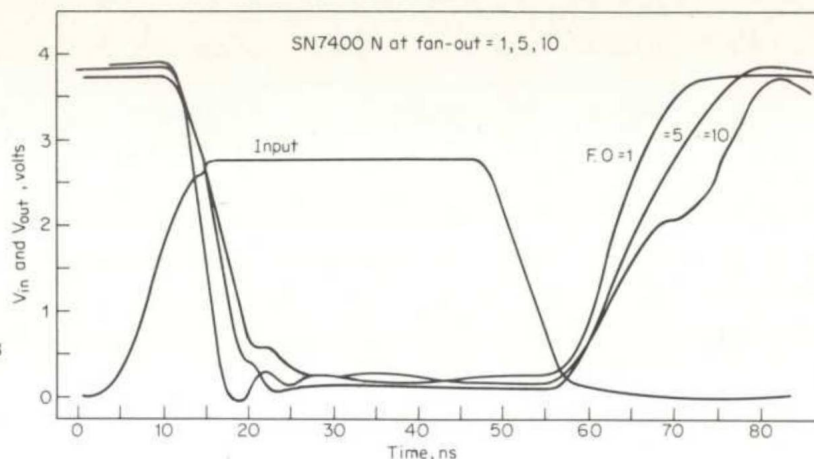
Medelvärdet av dessa fördröjningar brukar kallas "stegfördröjning" eller "propagation delay" och betecknas t_{pd} .

$$t_{pd} = \frac{t_{pLH} + t_{pHL}}{2}$$



Figur 3.12
Stegfördröjning hos
grind.

Figur 3.13
Typiska kurvformer hos
TTL-krets (vid 25 °C).
(Texas Instruments.)



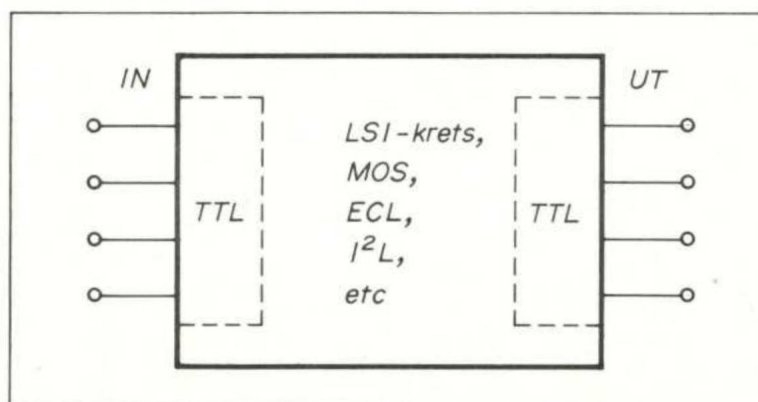
Typiska kurvformer för omslagsförloppen i en TTL-grind som belastas med en, fem eller tio likadana grindar visas i figur 3.13.

3 Typgrindens specifikation

En av orsakerna till TTL-familjens stora framgångar är det sätt på vilket kretsarna specificeras. Vi skall nedan bekanta oss med TTL-grindens specifikation, speciellt vad gäller nivåer, störmarginaler, belastbarhet och databladets uppställning.

De som arbetar med digitala kretsar stöter ständigt på TTL-kretsarnas specifikation - även vid arbeten med andra logikfamiljer. De flesta högintegrerade kretsar, dvs MSI- och LSI-kretsar*) görs i dag "TTL-kompatibla". Därmed avser man att ingångar och utgångar är anpassade för att drivas från respektive mata TTL-kretsar. Figur 3.14.

Figur 3.14
LSI-krets med TTL-kompatibla in- och utgångar.



3.1 TTL-familjens nivåer

Till det viktigaste i familjens specifikation hör spänningsnivåerna. Vi har tidigare bekantat oss med typiska utspänningsnivåer från en TTL-grind.

$$\begin{aligned} V_{OH} &= 3,3 \text{ V} & (\text{OH} = \text{output high}) \\ V_{OL} &= 0,2 \text{ V} & (\text{OL} = \text{output low}) \end{aligned}$$

Detta gäller alltså en typisk TTL-grind vid temperaturen 25 °C och vid matningsspänningen $V_{CC} = 5,0 \text{ V}$. Nu blir givetvis inte alla grindar exakt lika vid tillverkningen och därför varierar V_{OH} och V_{OL} något från exemplar till exemplar.

*) MSI = medium scale integration, dvs kapsel innehållande 10 - 100 grindar.
LSI = large scale integration, dvs kapsel innehållande mer än 100 grindar.

Både framspänningsfall (U_{BE}), strömförstärkningsfaktor (B) och resistansvärden varierar med temperaturen och det inverkar på V_{OH} och V_{OL} .

Om grinden belastas på olika sätt kommer V_{OH} och V_{OL} också att variera.

Slutligen måste man ju även tillåta en liten variation hos matningsspänningen V_{CC} .

Det är alltså många orsaker som gör att V_{OH} och V_{OL} inte kan förväntas ligga vid exakt 3,3 V respektive 0,2 V.

Vid den ursprungliga specifikationen av TTL-familjen angav man den "miljö" som typgrinden 7400 måste kunna arbeta vid på följande sätt:

Matningsspänning:	$4,75 \text{ V} \leq V_{CC} \leq 5,25 \text{ V}$
Temperatur:	$0^\circ \leq T \leq 70^\circ \text{C}$
Belastning:	max tio lika grindar

Inom detta område (och med godtyckliga kombinationer) garanterar fabrikanten att utspänningsnivåerna från typgrinden skall hålla sig inom följande värden:

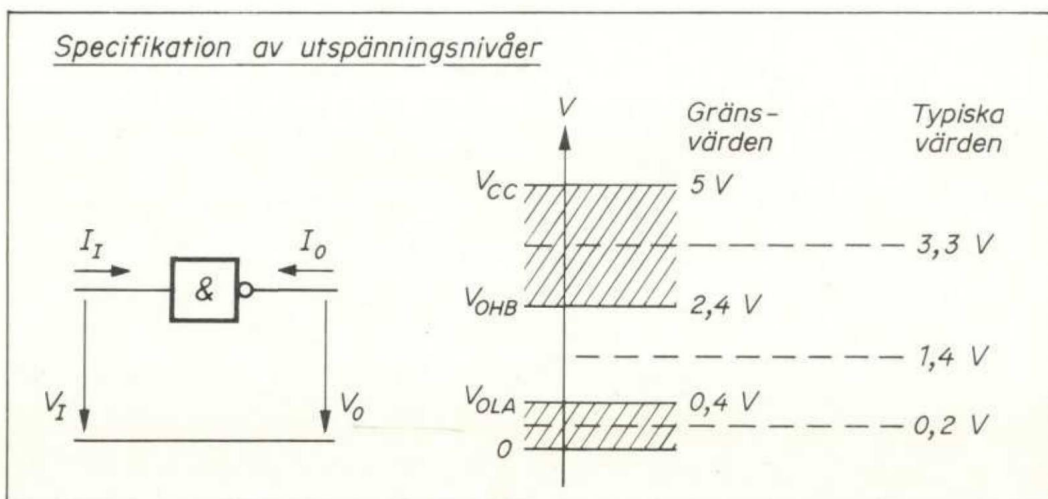
Utgångsnivåer:

$$\begin{aligned} V_{OH} &\geq 2,4 \text{ V om } V_{IL} \leq 0,8 \text{ V} \\ V_{OL} &\leq 0,4 \text{ V om } V_{IH} \geq 2,0 \text{ V} \end{aligned}$$

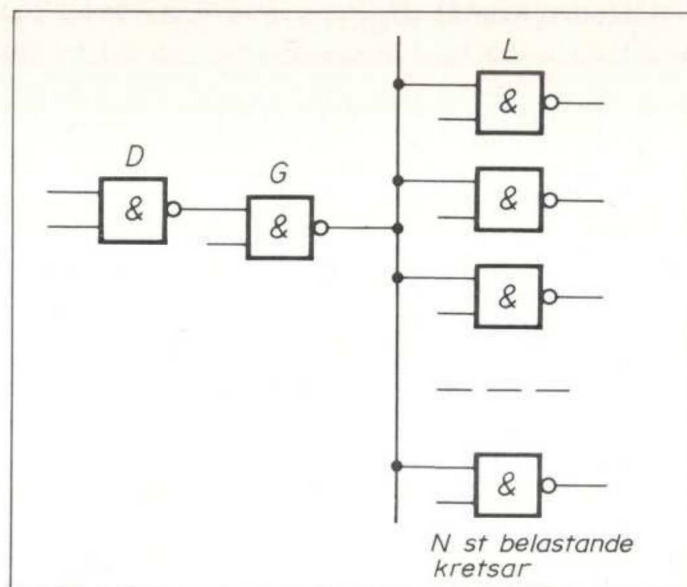
Man talar om "värsta fallet" (worst case) eftersom man får kombinera de svåraste påkänningarna och ändå skall nivåspecifikationerna hålla. Exempelvis måste typgrinden kunna sänka inströmmen till tio belastande (likadana) grindar även om temperaturen stiger till 70°C och matningsspänningen samtidigt faller till 4,75 V. Det är verkligen ett "värsta fall" sett ur TTL-grindens synvinkel.

IEC rekommenderar index A (above) för övre gräns och B (below) för nedre gräns och garantivärdena för spänningsnivåerna kan då tecknas, se figur 3.15

$$\begin{cases} V_{OHB} = 2,4 \text{ V} \\ V_{OLA} = 0,4 \text{ V} \end{cases}$$



Figur 3.15
TTL-familjens utspänningsnivåer.



Figur 3.16
Belastbarhet.

3.2 TTL-kretsens belastbarhet

TTL-kretsen skulle enligt ovan kunna belastas med maximalt $N = 10$ likadana kretsar. Man säger därvid att grindens belastbarhet (fan out, output loading capability) är $N = 10$.

Man använder begreppet referenslast i stället för "likadana kretsar", och tydligt är att denna referenslast måste specificeras. Referenslasten är ingångssidan på typgrinden. Ofta betecknas den UL (unit load) och för TTL-familjen specificeras referenslasten på följande sätt:

Referenslast (1 UL):

$$\begin{cases} I_{IH} = 40 \mu A & \text{vid } V_{OH} = 2,4 V \\ I_{IL} = -1,6 \text{ mA} & \text{vid } V_{OL} = 0,4 V \end{cases}$$

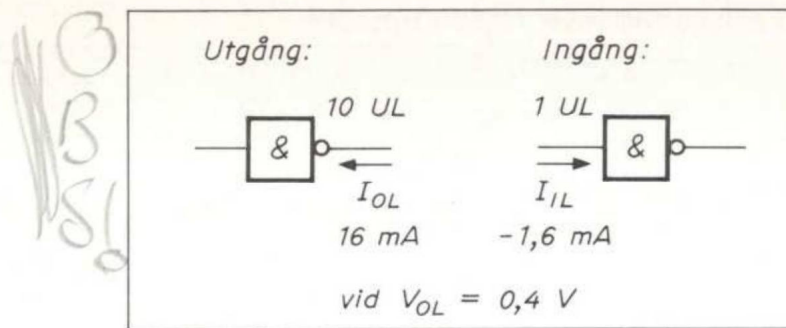
För att kunna driva 10 UL kan tydligen typgrindens belastbarhet på utgången specificeras (antingen som $N = 10$ eller, vilket är exakt samma sak) med nedanstående strömvärden:

TTL-grindens belastbarhet:

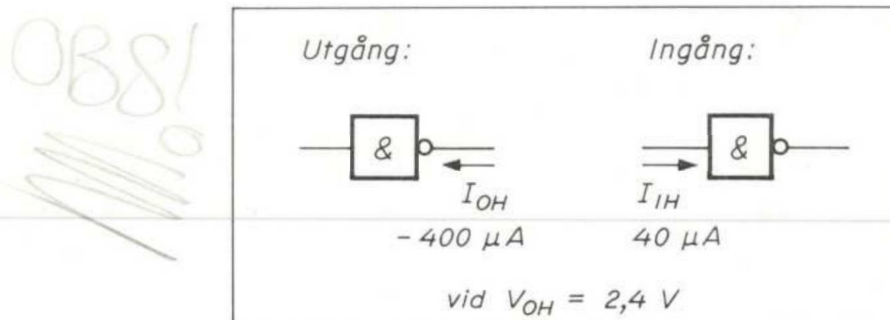
$$\begin{array}{ll} N = 10 & I_{OL} = 16 \text{ mA} \quad \text{för } V_{OL} = 0,4 V \\ & I_{OH} = -400 \mu A \quad \text{för } V_{OH} = 2,4 V \end{array}$$

Strömmar definieras enligt figur 3.15 positiva in mot grinden och därför får I_{OH} ett negativt värde.

I figurerna 3.17 och 3.18 sammanfattas TTL-grindens strömspecifikationer.



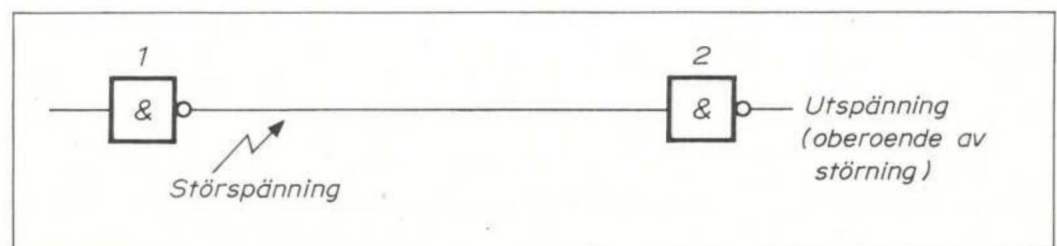
Figur 3.17
 TTL-kretsen i läge L.



Figur 3.18
 TTL-kretsen i läge H.

3.3 TTL-kretsens störmarginaler

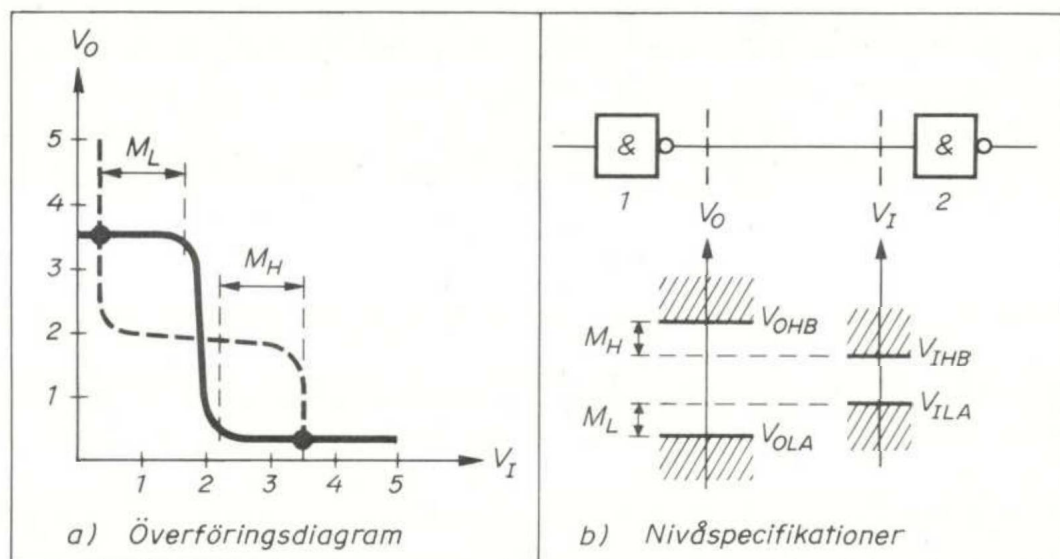
Ett digitalt system är tämligen okänsligt för störningar jämfört med ett analogt system. Figur 3.19 visar en grind som via en ledning matar en annan grind. I ledningar induceras ofta störspänningar. Vi frågar oss nu hur stor den i ledningen inducerade störspänningen kan tillåtas bli utan att störningen ger sig till känna på utgången av grind 2.



Figur 3.19
 Störspänning i digitalt system.

3.3.1 Typiska och garanterade störmarginaler

Vi kan analysera problemet med störspänningen på flera sätt. Figurerna 3.20 a och b ger två lösningar.



Figur 3.20
Störmarginalerna M_H och M_L .

Störmarginalen ca 0,4 V

Figur 3.20a visar överföringsdiagrammet för grind 2 i figur 3.19. Normalt ligger grinden i något av de två lägen som är markerade med prickar i diagrammet. Observera nu att inspänningen V_I till grind 2 kan variera utan att detta inverkar på utspänningen V_O . I läge "låg" på ingången kan V_I ökas med värdet M_L och i läge "hög" kan den minskas med värdet M_H innan utspänningen märkbart ändras. Man har alltså en marginal till godo på inspänningen - man kan tillåta en störning på ledningen av storleken M_L eller M_H utan att utspänningen påverkas. Man kallar därför M_L och M_H för störmarginaler (M_L i läge "låg" på ingången och M_H i läge "hög" på ingången). Engelska termer: noise margin, noise immunity margin, noise immunity.

I figur 3.20b har vi angripit vårt problem med störningen från en annan utgångspunkt. Vi har markerat de garanterade utnivåerna från grind 1 med V_{OHB} och V_{OLA} och de tillåtna innivåerna till grind 2 med V_{IHB} och V_{ILA} .

När vi kopplar ihop grindarna med ledningen ser vi att det är en marginal mellan den garanterade utspänningen och den tillåtna inspänningen både i läge H och läge L.

I figur 3.20b har vi markerat de specificerade störmarginalerna och de kan tydligen tecknas:

$$\begin{cases} M_H = V_{OHB} - V_{IHB} \\ M_L = V_{ILA} - V_{OLA} \end{cases}$$

B = undre gräns

A = övre gräns

I figur 3.20a har vi ritat en typisk överföringskurva. När fabrikanten specificerar sina nivåer måste han ta hänsyn till värsta fallet (kurvan varierar med temperatur, matningsspänning och belastning och den spri-

der vid tillverkningen) och därför blir de "typiska" störmarginaler vi skisserat i figur 3.20a avsevärt större än de "garanterade störmarginaler" som angivits i figur 3.20b.

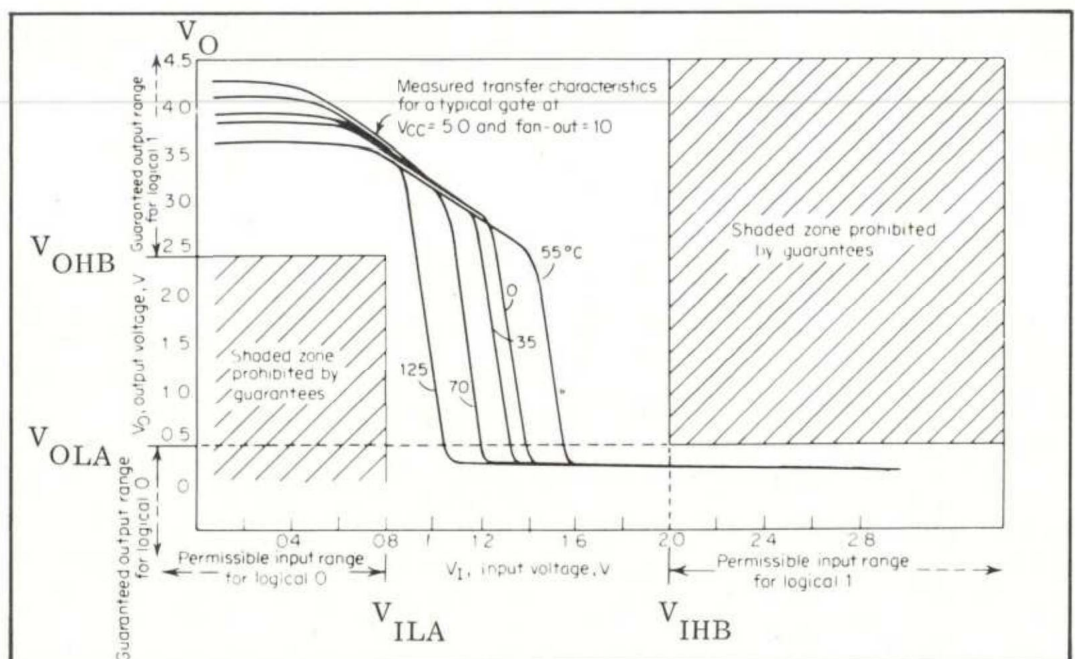
OBS! För TTL-kretsar kan typiska värden på båda störmarginalerna röra sig om ca 1 V medan de garanterade störmarginalerna i båda lägena är 0,4 V.

Att TTL blivit en så populär logik beror bl a på att originalfabrikanten, Texas Instruments, lade fast nivåerna på ett optimalt sätt och att han härvid inkluderade spridning i data vid tillverkningen, temperaturvariationer och variationer i matningsspänning. För standard TTL, dvs 74-serien, är de tillåtna variationerna i temperatur och matningsspänning satta på följande sätt:

Tillåtet temperaturområde: $0^{\circ}\text{C} - 75^{\circ}\text{C}$

Tillåten matningsspänning: $4,75\text{ V} - 5,25\text{ V}$

Figur 3.21 visar Texas Instruments' specifikation av nivåerna V_{ILA} , V_{IHB} , V_{OLA} och V_{OHB} .



Figur 3.21

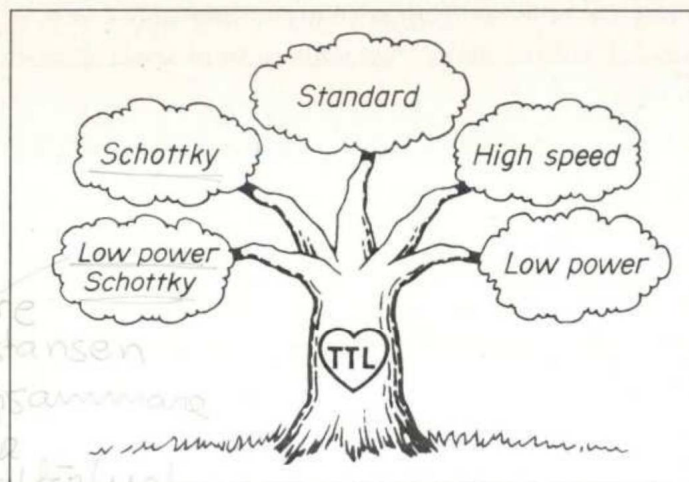
Specifikation av nivåer för standard TTL-logik.

Som merläsning rekommenderas Texas Instruments' Application report: TTL: From the beginning, B 124. I denna genomgång typgrindens datablad utförligt.

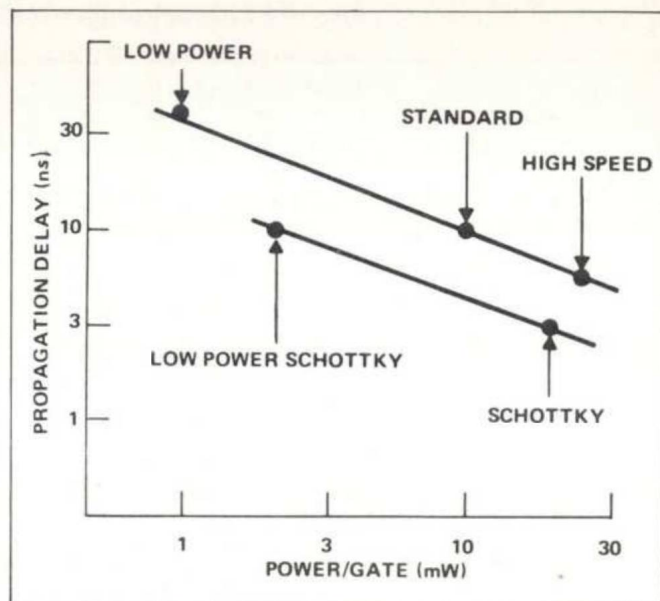
4 TTL-familjens stamträd

I vissa tillämpningar har man behov av stor snabbhet medan man i andra kräver låg effektförbrukning. Som tidigare nämnts måste man betala snabbheten med högre effekt och därför kan inte kraven på hög snabbhet och låg effekt uppfyllas samtidigt.

Man gjorde därför tidigt två varianter av TTL-familjen. Den ena gjordes med lägre resistansvärden och blev därmed snabbare men också mer effektkrävande. Den kallas "high speed TTL" eller kort och gott 74H-serien.



Figur 3.22
TTL-familjens stamträd.



Figur 3.23
TTL-familjens fem grenar har optimerats för olika krav på snabbhet och effektkonsumtion. (Texas Instruments.)

Den andra serien gav man större resistansvärden och man kunde därmed minska förlusteffekten per grind från 10 mW till 2 mW, men givetvis till priset av mindre snabbhet. Denna serie kallas "low power TTL" eller 74L-serien.

Under 1970-talet har ytterligare två varianter tillkommit, nämligen "Schottky TTL" och nu sist "low power Schottky". TTL-familjens stamträd omfattar därför fem grenar enligt figur 3.22. De tre förstnämnda är gjorda i samma "teknologi" och därför ligger de på en rät linje i effekt/tid-diagrammet i figur 3.23. Schottky-kretsarna utnyttjar en annorlunda teknologi och ligger därför på en annan linje.

Man brukar ofta säga att en viss teknologi har en viss "speed-power product", dvs ett typiskt och konstant värde på produkten av stegfördröjning och förlusteffekt. Vill man minska stegfördröjningen så ökar förlusteffekten och vice versa, men produkten är i viss mån konstant.

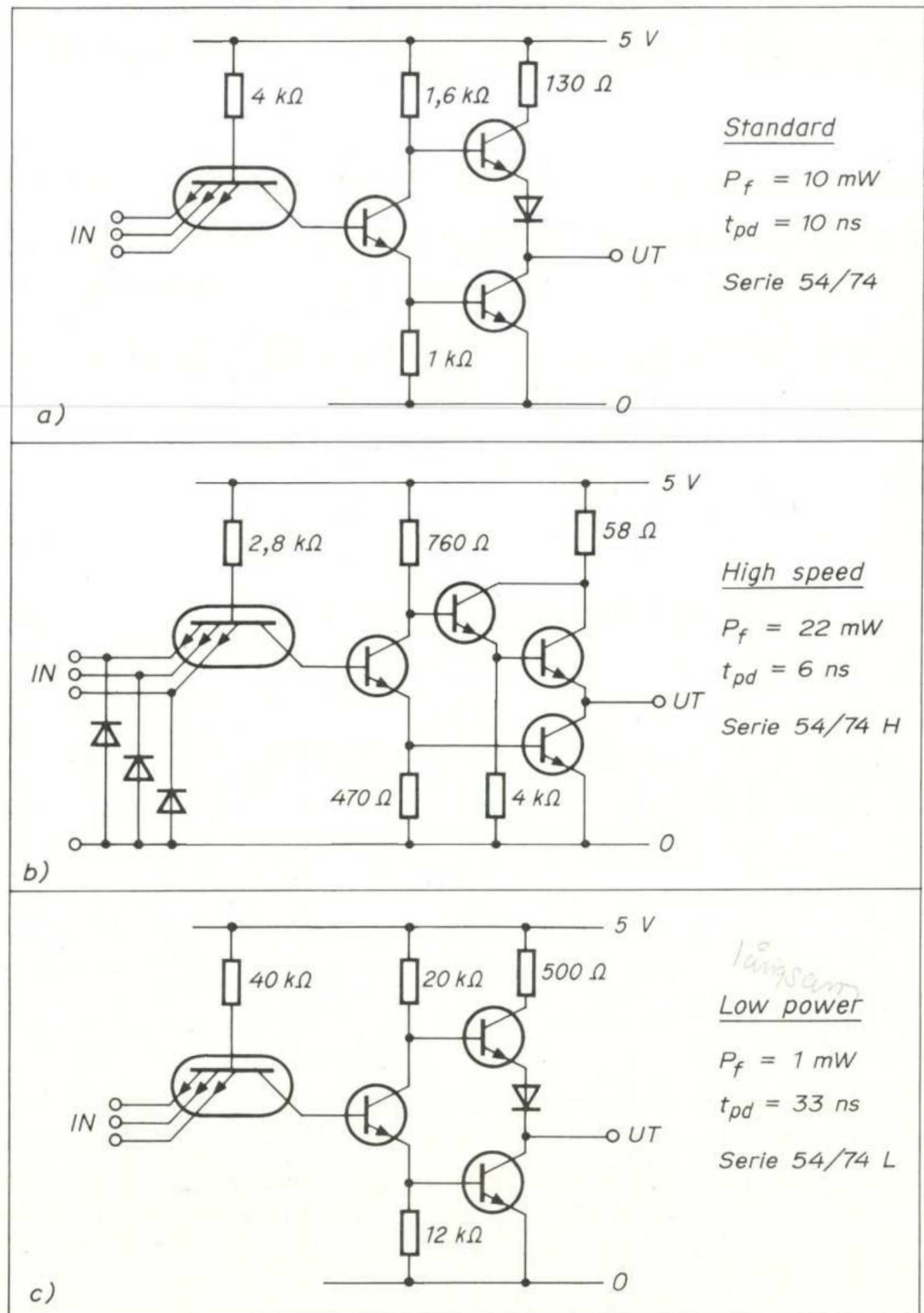
Termen "speed-power product" är språkligt missvisande eftersom man vanligen avser en "tid-effekt produkt". Tid-effekt-produkten brukar anges i pJ. Figur 3.24 visar typiska värden för stegfördröjning, förlusteffekt och speed-power produkt för TTL-familjens fem grenar.

Type Of TTL	Speed ns	Power mW	Product pJ
Low Power	33	1	33
Low Power Schottky	10	2	20
Standard	10	10	100
High Speed	6	23	138
Schottky	3	19	57

Figur 3.24
Typiska värden för speed-power produkten hos TTL-familjens fem grenar. (Texas Instruments.)

4.1 High speed och low power

I figur 3.25 jämförs typgrindarna hos 74H och 74L med "standard" 74. Man ser ofta beteckningen 54/74 och det beror på att man tillverkar en serie, TTL serie 54, som tål ett större temperaturområde och ett större matningsspänningsområde men för övrigt har samma specifikation som serie 74.



Figur 3.25
TTL-familjen, standard, high speed och low power.

	Temperatur	V_{CC}
serie 74	$0^{\circ} - +70^{\circ}\text{C}$	4,75 V - 5,25 V
serie 54	$-55^{\circ} - +125^{\circ}\text{C}$	4,5 V - 5,5 V

I figur 3.25 b ser vi att 54/74H-typgrinden har lägre resistansvärden medan 54/74L-grinden i figur 3.25c har högre. Båda är specificerade med en belastbarhet på 10. Detta gäller vid belastning med likadana grindar. Spänningsnivåerna på in- och utgångar är lika och därmed kan man blanda grindar och använda snabba grindar på de ställen i ett system där detta är nödvändigt, men spara effekt genom att använda "low power" på andra ställen.

Vid blandning av grindar måste man se upp med det antal grindar av annan typ som utgör maximal belastning. En standardgrind kan exempelvis driva 40 stycken 74L men enbart 5 stycken 74H-grindar. En 74L-grind kan driva 1 st 74 + 2 st 74L men enbart 1 st 74H.

I serierna ingår ett antal vippor av olika typer. Typiska räknehastigheter kan vara av storleksordningen:

74 standard	35 MHz
74 H	50 MHz
74 L	3 MHz

4.2 Schottky och low power Schottky

Vi har tidigare konstaterat att det normalt är hård bottning som mest begränsar transistorswitchens snabbhet. För att få snabba omslag, dvs liten stegfördröjning, måste vi alltså undvika att bottena transistorswitchar. ECL är ett exempel på detta.

4.2.1 Schottky-transistorn

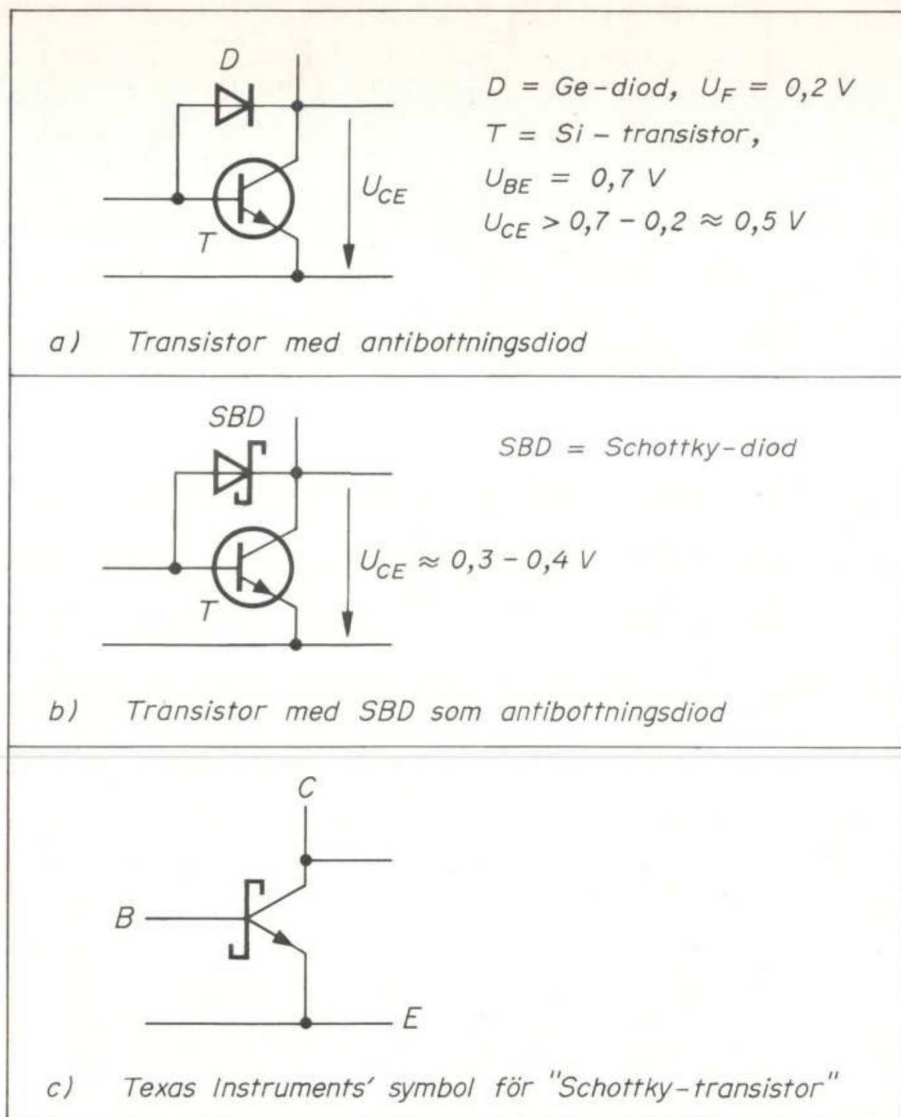
Man kan undvika att bottena en transistorswitch genom att koppla in en diod som i figur 3.26a. Om transistorn är en kiseltransistor ($U_{BE} = 0,7\text{ V}$) och dioden en germaniumdiod ($U_F = 0,2\text{ V}$) så kan tydligen spänningen U_{CE} över transistorn aldrig understiga värdet:

$$U_{CE} = 0,7 - 0,2\text{ V} = 0,5\text{ V}$$

Dioden hindrar därmed transistorn från att bottena och en diod som inkopplas på detta sätt kallades förr "antibottningsdiod".

Normalt kör vi in mer basdrivström än vad som behövs för bottning. Med antibottningsdiod avleds denna överskottsdrivström via antibottningsdioden och transistorn (kollektor till emitter). Transistorn får på detta sätt automatiskt så mycket basström att den i till-läge håller $U_{CE} = 0,5\text{ V}$ utan att bottena.

Om man lägger ett aluminiumskikt på ett svagt dopat N-skikt uppstår en likriktarverkan mellan metall och halvledare. Dioder uppbyggda på detta sätt kallas Schottky-dioder (SBD = surface barrier diode) och har ett framspänningsfall av ca 0,4 V. Schottky-dioder är utomordentligt snabba till skillnad från vanliga PN-dioder.



OBS!

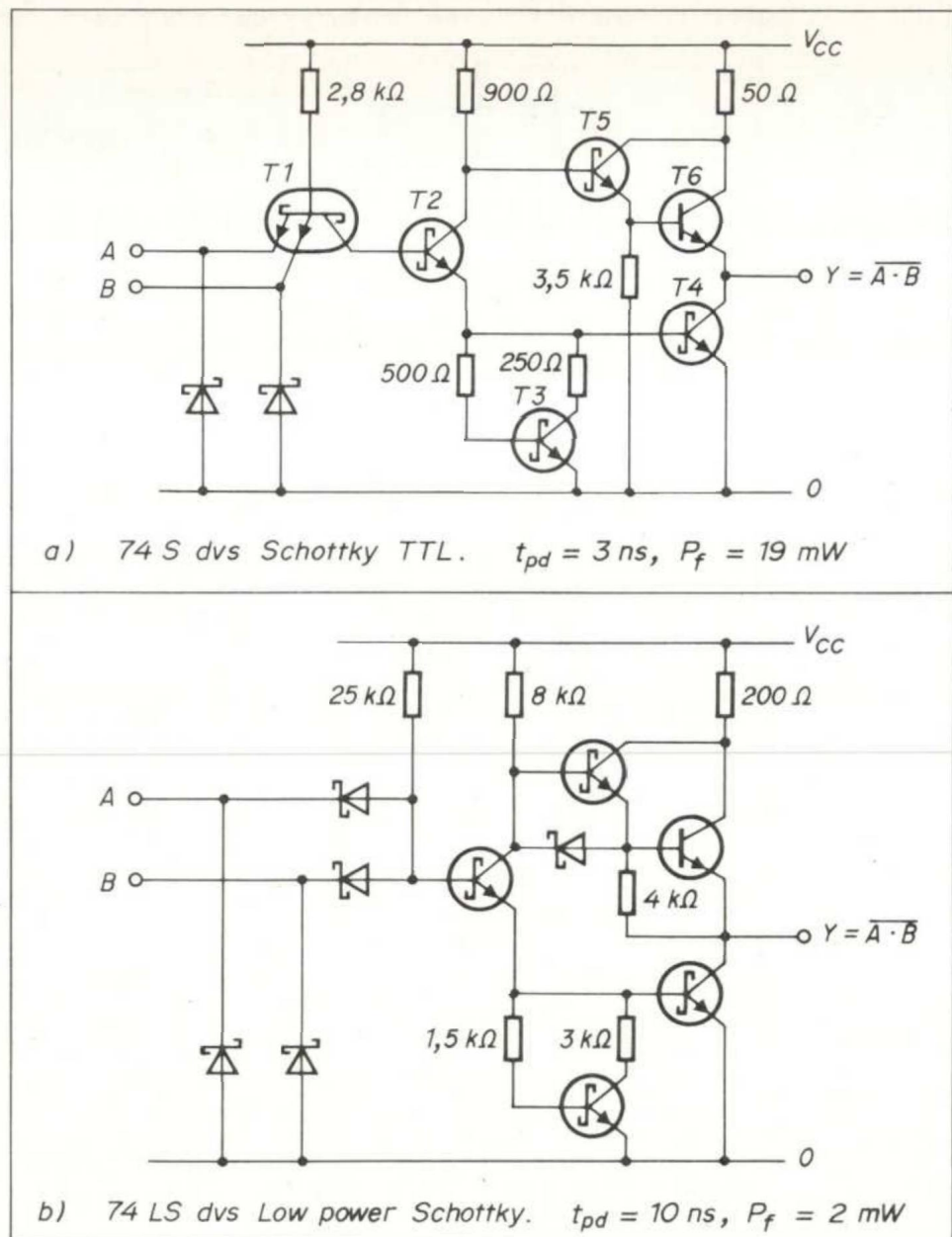
Figur 3.26

Schottky-transistorn är en kombination av en kiseltransistor och en Schottky-diod.

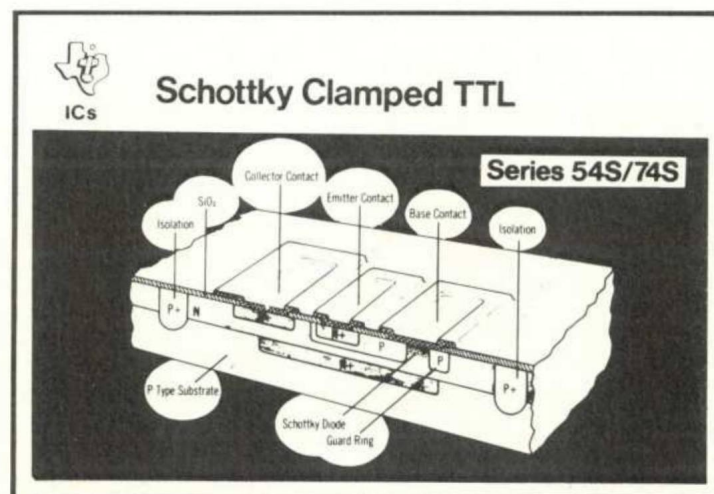
vader en LS-krets är och kan
man gjort.

Med Schottkytransistorn har en ny "teknologi" introducerats och därmed har man uppnått ett lägre värde på "tid-effekt-produkten" vilket framgår av figur 3.24. Man har nu utkommit med två Schottkyfamiljer. De kallas 74S (S = Schottky) och 74LS (LS = low power Schottky). Typgrindarna framgår av figur 3.27.

Att framställa en Schottkytransistor är ganska enkelt (när man väl behärskar processerna). Som framgår av figur 3.28 bildas en Schottkydiod (bas/kollektor) om man breder ut metallkontakteringen till basen över en del av kollektorskiktet.



Figur 3.27
Typgrindarna hos TTL och TTLS



Figur 3.28
Schottky-transistorns kristallstruktur.

4.2.2 Schottky-grindarna

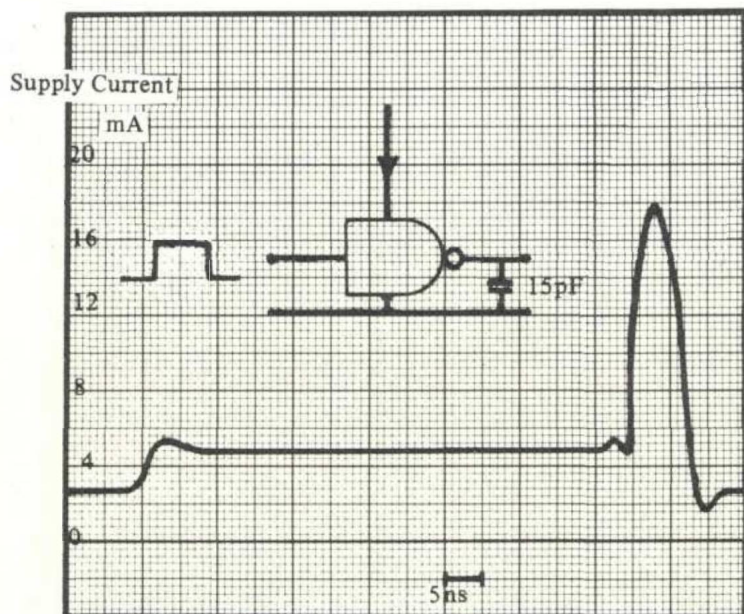
När detta skrivs, 1975, sker en stor del av nyutvecklingen av bipolära digitala system med Schottky-kretsar. Det finns därför anledning att diskutera grindarna i figur 3.27 något utförligare.

I Schottky-grinden i figur 3.27a är alla transistorer utom en av Schottky-typ. Det är bara emitterföljaren T6 ("pull up") på utgången som är av konventionell typ men den bottnas ju aldrig så därför är det inte nödvändigt med Schottkytransistor här. Den aktiva kollektordrivningen i läge hög ("pull up") sker som synes med ett Darlingtonkopplat steg, dvs två kaskadkopplade emitterföljare (T5 och T6).

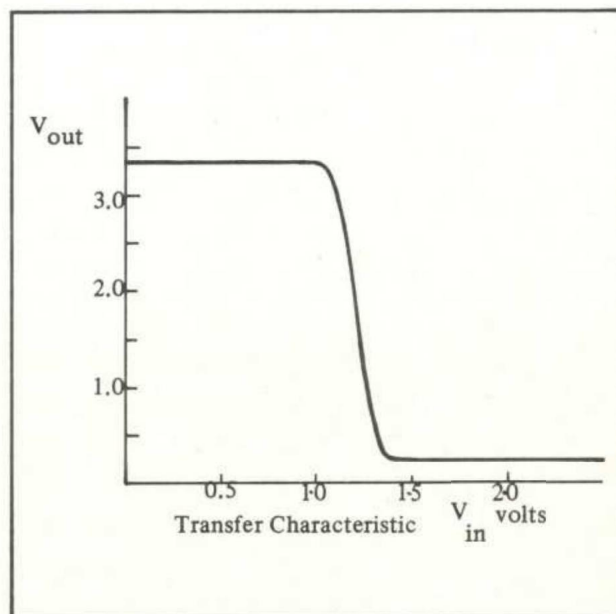
Det sitter en transistor (T3) mellan inverterartransistorns bas och jord och den har två väsentliga uppgifter:

- För det första stryker den effektivt inverterartransistorn T4 vid frånslag ("pull down") och därigenom begränsas storleken av den strömspik som uppträder vid omslag (L till H). Figur 3.29 visar en typisk kurvform.
- För det andra inverkar T3 på överföringskurvens utseende så att denna blir mer "fyrkantig" och grinden får därigenom bättre störmarginaler. Figur 3.30.

På ingången till såväl 74S som 74LS-grinden i figur 3.27 sitter klippdioder (diode clamps). Dessa begränsar eventuella negativa transienter som annars lätt uppträder vid omslagsförlopp i snabba kretsar.



Figur 3.29
Strömspiken hos 74S. (Texas Instruments.)



Figur 3.30
Överföringsdiagrammet hos 74S.
(Texas Instruments.)

74LS-grinden har en vanlig diodgrind på ingången medan 74S-grinden har en multiemittertransistor av Schottky-typ. Det är anmärkningsvärt att man med low power Schottky uppnår samma snabbhet som hos TTL-grinden, dvs $t_{pd} = 10$ ns, trots att effektförlusten hos LS enbart är 2 mW dvs femtedelen av standardgrindens förlusteffekt. Detta är givetvis orsaken till att 74LS-familjen blir allt mer dominerande vid all nyutveckling.

Typiska värden för högsta räknefrekvens är för 74S ca 125 MHz och för 74LS ca 50 MHz. Nedanstående tabell visar en sammanställning av de data vi tidigare diskuterat (Texas Instruments):

	74	74LS	74S	74L	74H	
Matningsspänning	5	5	5	5	5	V
Belastbarhet	10	20	10	10	10	-
Typisk störmarginal	1	1	1	1	1	V
Frekvens	35	50	125	3	50	MHz
Stegfördröjning	10	10	3	33	6	ns
Förlusteffekt (per grind)	10	2	19	1	33	mW
Tid-effekt-produkt	100	20	57	33	138	pJ

4.3 Antalet typer i TTL-familjen

Sommaren 1975 uppgår sortimentet från enbart en försäljare av TTL-kretsar (Texas Instruments) till mer än 550 olika typer. Grovt kan dessa delas upp på följande kategorier:

grindar (gates)
vippor och räknare (flip-flops and counters)
skiftregister (shift registers)
multiplexrar och demultiplexrar (multiplexers and decoders)
aritmetiska kretsar (arithmetic devices)
monostabila kretsar (monostables)
linjekretsar (line drivers and receivers)

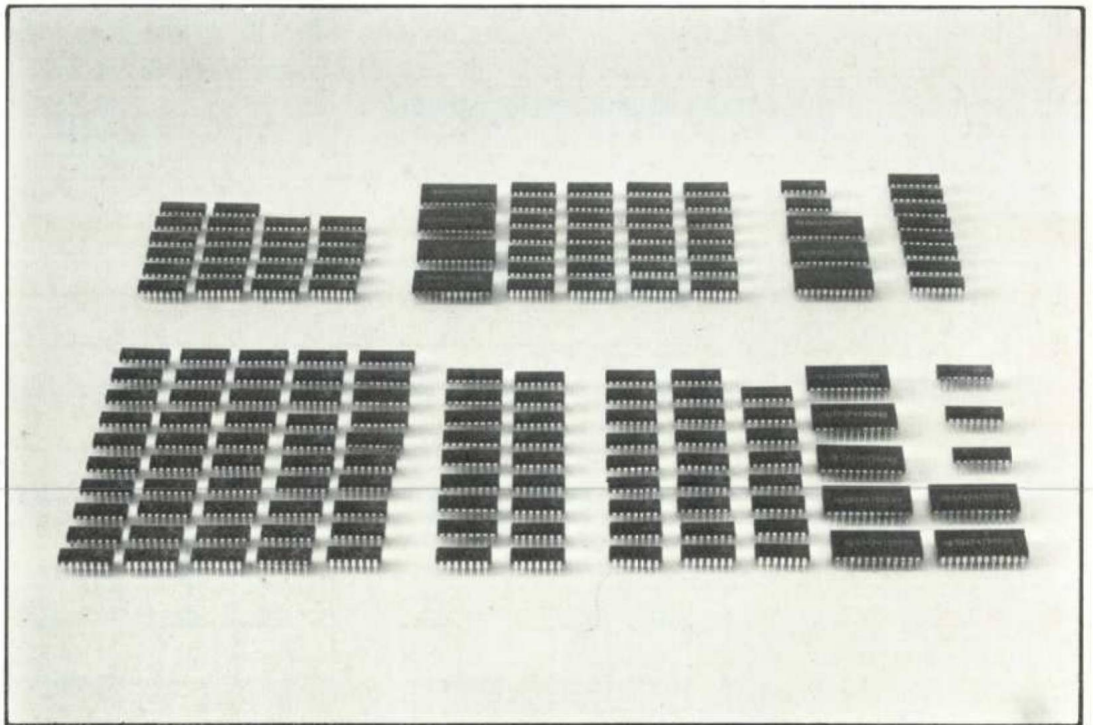
Komplexiteten hos kretsarna ökar och av de totalt 550 olika typerna är ca 200 av MSI-typ dvs innehåller 10-100 grindar per krets.

Den ungefärliga försäljningsvolymen av TTL-kretsar i Sverige år 1975 ca 25 Mkr och 40 % därav avser MSI medan resterande 60 % är SSI. (SSI = small scale integration.) Medelpriset på TTL-kretsar, vid försäljning i stora antal, år 1975 ca 1,50 kr.

Om man tar hänsyn till att TTL-kretsar säljs för olika temperaturområden (54- och 74-serierna) och i olika kapslar (DIL-plast, DIL-keramik och flatkapsel) samt dessutom i olika kvalitetsnivåer (ex HIREL = high reliability och consumer) blir totala utbudet av TTL-kretsar ca 10.000 olika typer.

En så komplett utvecklad familj kommer att leva kvar länge på marknaden även om nya familjer, exempelvis CMOS, konkurrerar starkt på vissa områden.

Figur 3.31 visar exempel på några kretsar ur TTL-familjen. Utbudet av kretsar är som sagt mycket stort och den bästa översikten av tillgängliga TTL-kretsar får man från fabrikanternas databöcker*).



Figur 3.31
Exempel på några kapslar ur TTL-familjen. (Texas Instruments.)

5 Buss- och linjekretsar

Innan vi lämnar TTL-kretsarna skall vi kort beröra två typer av kretsar som används vid datakommunikation via bussar eller via längre ledningar. Vid matning av bussar kan man använda drivkretsar (drivers) av typ "öppen kollektor" eller också "tristate". För dataöverföring på längre ledningar finns speciella linjekretsar, s k "line drivers" och "line receivers". Följande sidor utgör merläsning för speciellt intresserade.

5.1 Krav på grindar för bussmatning

Vi har tidigare diskuterat hur grindar kan användas för att avläsa information från databussar, figur 1.25.

Det är flera saker man måste tänka på när det gäller bussar som skall arbeta med snabba förlopp.

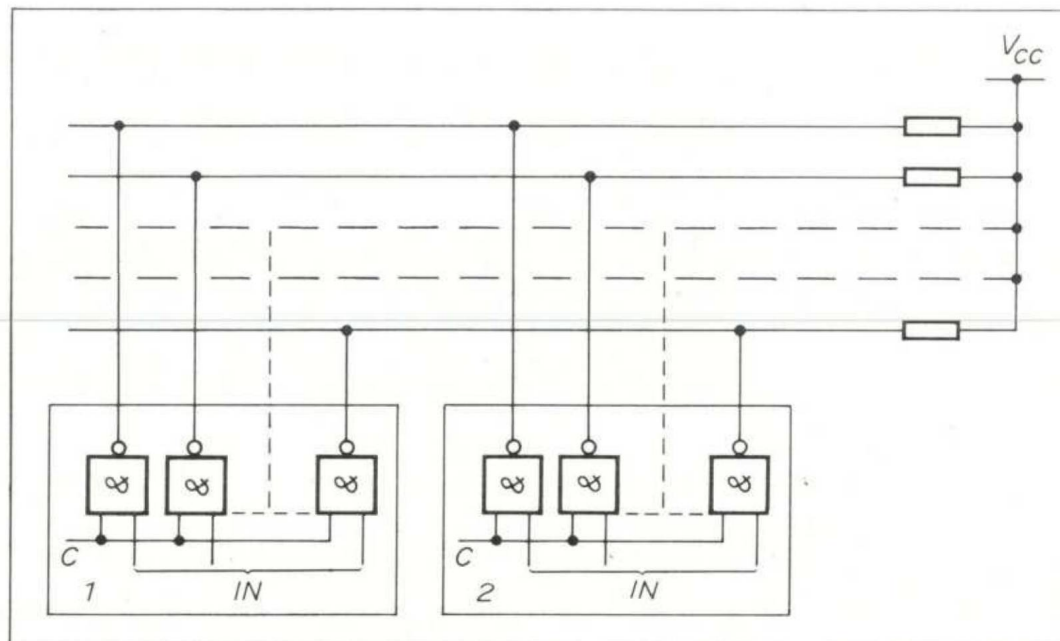
- En bussledning är ju i princip en transmissionsledning och därför uppstår reflexioner i dess ändar om den inte anpassas i ändarna med lämpliga resistorer (med resistansvärden som motsvarar ledningens karakteristiska impedans).

*) Exempelvis "System 74. Designers Manual" från Texas Instruments.

- Om ledningen matas från flera källor måste man se upp så att inte dessa samtidigt styr ledningen, exempelvis så att en signalkälla vill lägga ledningen hög medan en annan vill lägga den låg.

5.2 Öppen kollektor

I figur 3.32 har vi två signalkällor som matar samma bussledning. Signalerna matas ut på bussen via 2-ingångars NAND-grindar varav den ena är styringång (C) medan den andra är dataingång. Bussledningen är anpassad i ena änden (och skulle kunna vara det även i den andra, om krav ställs på små reflexioner).



Figur 3.32

Två signalkällor som matar samma buss.

Om grindarna i figur 3.32 vore vanliga 74-grindar med totempåle skulle en inaktiv signalkälla lägga alla ledningarna höga och därmed blockera bussledningen.

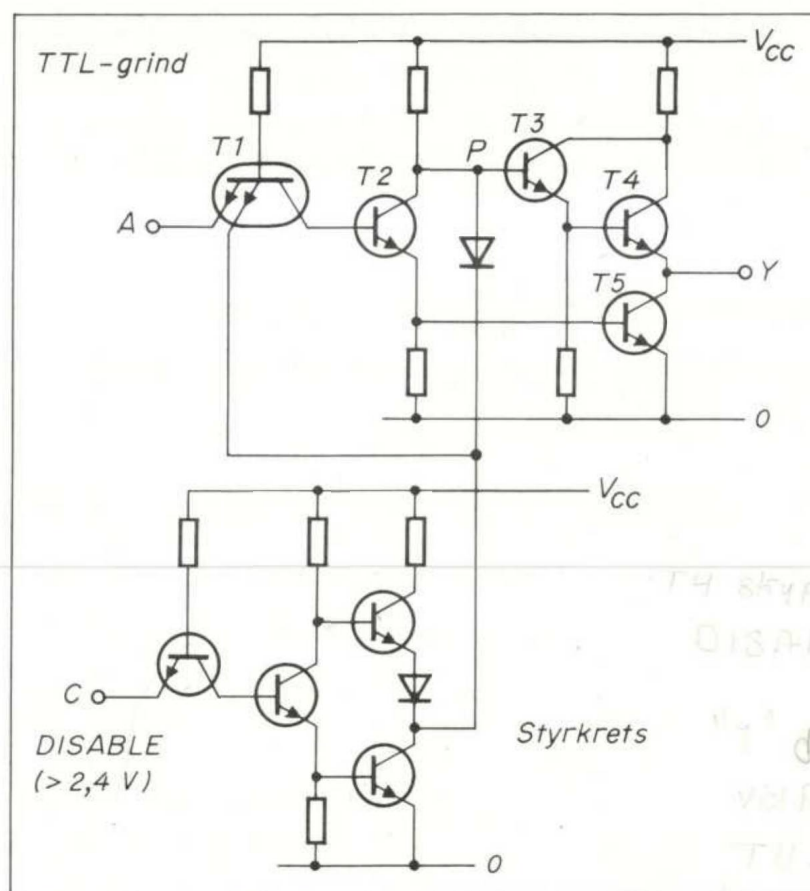
Om grindarna i figur 3.32 däremot är av typ "öppen kollektor" kommer ledningen att vara fri vid inaktiv dvs låg styrspänning. Att ledningen är "fri" betyder i detta fall att anpassningsresistorerna fungerar som yttre kollektorresistorer (wired AND) och håller ledningen hög om ingen transistor ligger i till-läge.

Den signalkälla som inkopplas "sänker" ledningen vid signal. Även om flera signalkällor av misstag skulle inkopplas samtidigt skadas inte kretsarna (av motstridiga signaler).

Öppen kollektor är mycket vanlig vid matning av bussar. Den har emellertid en nackdel, den är långsam vid omslag $L \rightarrow H$ eftersom man inte har en "active pull-up" som i totempålen.

5.3 TRI-STATE

För att få bättre drivning och mindre belastning av bussledningar har NS (National Semiconductor Corp.) utvecklat en "TRI-STATE" grind. Den kallas tristate därför att utgången har tre lägen: låg, hög och frikopplad.



Figur 3.33
TRISTATE-grinden.

Tristate-grinden fungerar vid sändning eller mottagning som en normal TTL-grind med totempåle och den är därför snabb vid omslag (i båda riktningarna). I stället för styringången i öppen kollektor-kretsen har man en frikopplingsingång (disable). Läggs denna hög är både ingång och utgång "frisvävande" och utgör därför ingen belastning på bussen, vare sig tristate-grinden arbetar som sändare eller mottagare. Figur 3.33 visar tristate-grindens schema.

Om disable-ingången på styrkretsen läggs hög ($>2,4$ V) bringas punkten P i den övre TTL-kretsen till spänningen 1 V samtidigt som en av multi-emitteringångarna läggs på 0,2 V. I detta läge är både T4 och T5 på utgången ströpta och ingången A på T1 utgör en backspänd diod (inom arbetsområdet för V_P).

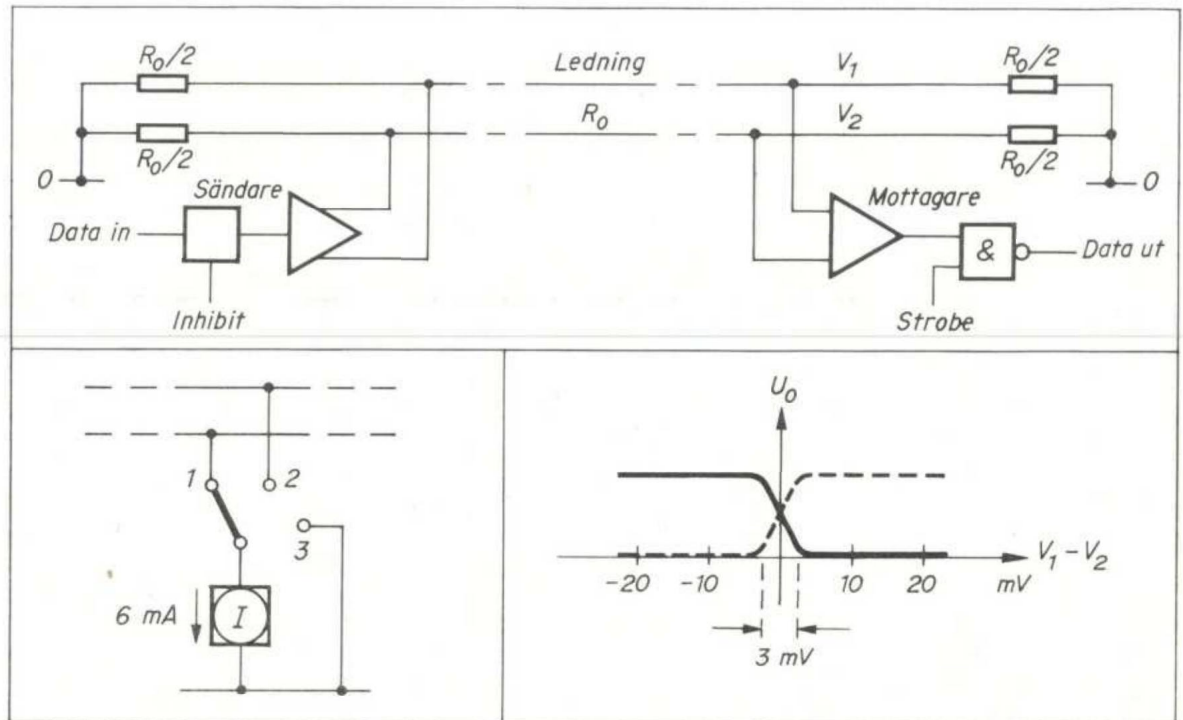
Fördelen med en tristatekrets som mottagare är alltså att den inte belastar bussledningen (med T1:s basresistor) när den är frikopplad. Man kan därför hänga på många tristate-grindar på en buss utan nämnvärd belastning.

Fördelen med en tristatekrets som sändare är att den driver hårt i båda riktningarna (till skillnad från öppen kollektor). Här måste man dock se upp så att inte flera tristatekretsar blir inkopplade samtidigt för då kan de skada varandra.

5.4 Line drivers och line receivers

För dataöverföring på längre ledningar används normalt tvinnade parledningar (twinned pairs) eller koaxialkablar. Störningar som inkommer på tvinnade parledningar kan utbalanseras om man använder differentialförstärkare som mottagare. En differentialförstärkare är ju endast känslig för differensspänning och störningar inkommer ju lika på båda ledningarna och bildar därmed ingen differensspänning.

Figur 3.34 visar principen för en sändare (line driver) och en mottagare (line receiver). Ledningen är anpassad i båda ändar och symmetriskt jordad.



Figur 3.34
Linjekretsar på en parledning.

Sändaren fungerar i princip som en strömgenerator som suger en konstant ström ur ledning 1 eller 2 beroende på insignalen (data in). Vid inhibit-signal kopplas strömgeneratoren bort (läge 3 i principskissen i figur 3.34b). Denna inhibit-signal (eller frisisignal) frikopplar alltså sändaren från ledningen på samma sätt som disable-signalen på tristate-grinden i figur 3.33. (Frisignal, disable och inhibit är synonyma begrepp.)

En vanlig line driver (75107) suger 6 mA från ledningen och normal ledningsimpedans kan röra sig om $R_0 = 150 \Omega$. Man får därvid en differensspänning mellan ledarna på ca

$$\Delta U = I \cdot \frac{R_0}{4} \approx 225 \text{ mV}$$

Mottagaren ger fullt utslag för en differensspänning på några få mV. Överföringskurvan framgår av figur 3.34c och inimpedansen är hög. Ut-signalen "strobas", dvs avkänns, när hög spänning läggs på strob-ingången. Utspänningens nivåer är anpassade till TTL-kretsar och detta är även fallet med sändarens insignalnivåer.

Tydligt är att flera sändare och mottagare kan operera i tidsmultiplex (dvs inkopplas vid olika tidpunkter) på samma ledning precis som vi tidigare beskrivit i samband med tristate och öppen kollektor. Man kallar ibland ledningar som används på detta sätt för "party line".

Repetitionsfrågor

1. Vilka fördelar har TTL jämfört med DTL?
2. Varför är omslagsområdet en bruten linje hos 7400 men relativt rät hos 74S00?
3. Vilka är spänningsnivåerna och hur stor är belastbarhet och strömsänkingsförmåga hos en standard 74-krets?
4. När uppträder strömspikar i TTL, då utgången går låg eller hög? Varför?
5. Varför är typiska och garanterade störmarginaler olika för TTL-grinden, och hur stora är de?
6. Vilket samband råder mellan stegfördröjning och förlusteffekt för en given bipolär teknologi?
7. Hur fungerar en antibottningsdiod och vad är en Schottky-transistor?
8. Beskriv kortfattat de fem grenarna i TTL-familjens stamträd.
9. Varför kan man inte använda 7400 när flera sändare skall mata samma buss?
10. Vad menas med sink och source?

D4 CMOS-kretsar

CMOS-kretsar 90

1 MOS-transistorn, fälttransistorn och bipolar-transistorn 90

2 PMOS och NMOS 92

- 2.1 PMOS-teknologins fördelar 92
- 2.2 Grindar och vippor i PMOS 93
- 2.3 Dynamiska MOS-register 94
- 2.4 NMOS 96

3 Principen för CMOS 97

- 3.1 CMOS-switchen 97
- 3.2 NAND- och NOR-grindar i CMOS 102

4 CMOS-vippor 105

- 4.1 Transmissionsgrinden 105
- 4.2 MS-vippor i CMOS 108

5 CMOS-kretsar och deras specifikation 110

- 5.1 Nivåer och störmarginaler 110
- 5.2 Snabbhet 111
- 5.3 Förlusteffekt 111

Repetitionsfrågor 113

Målruta

När du läst detta kapitel skall du kunna:

- rita överföringsdiagrammet för en CMOS-switch och förklara varför det är symmetriskt
- förklara varför CMOS-grindar är känsliga för statisk elektricitet och hur kretsarna skyddas
- rita och förklara funktionen hos NAND- och NOR-grinden samt transmissionsgrinden
- förklara varför CMOS har så liten förlusteffekt och hur denna varierar med frekvens, spänning och kapacitiv belastning
- rita en principskiss för ett dynamiskt PMOS-register
- rita ett schema för en MS-vippa med transmissionsgrindar och förklara dess funktion
- ange hur störmarginaler kan specificeras vid CMOS

CMOS-kretsar

I detta kapitel skall vi bekanta oss med CMOS-kretsarna, dvs den logikfamilj som nu konkurrerar hårt med de tidigare helt dominerande TTL-kretsarna.

Vi börjar med en kort återblick på PMOS och NMOS. PMOS, eller P-kanal MOS, var den första MOS-teknologin och den utvecklades vid mitten av 1960-talet. NMOS kom först vid slutet av 1960-talet. CMOS är en kombination av PMOS och NMOS och introducerades först i slutet av 1960-talet av RCA.

Vi skall studera CMOS-switchen, NAND-grinden och NOR-grinden. Det är framför allt dessa kretsars låga förlusteffekt och stora störmarginaler som gjort CMOS till ett så populärt byggsystem för digitala kretsar.

Vi har tidigare sett hur TTL-grindar kan användas som signalgrindar - men enbart i en riktning. Transmissionsgrinden är en signalgrind i CMOS som har den unika egenskapen att den kan användas i båda riktningar. Vi skall studera transmissionsgrinden och dess egenskaper samt hur den används bl a för uppbyggnad av vippor.

Som avslutning skall vi ge en liten översikt över CMOS-kretsarnas specifikation.

1 MOS-transistorn, fälttransistorn och bipolartransistorn

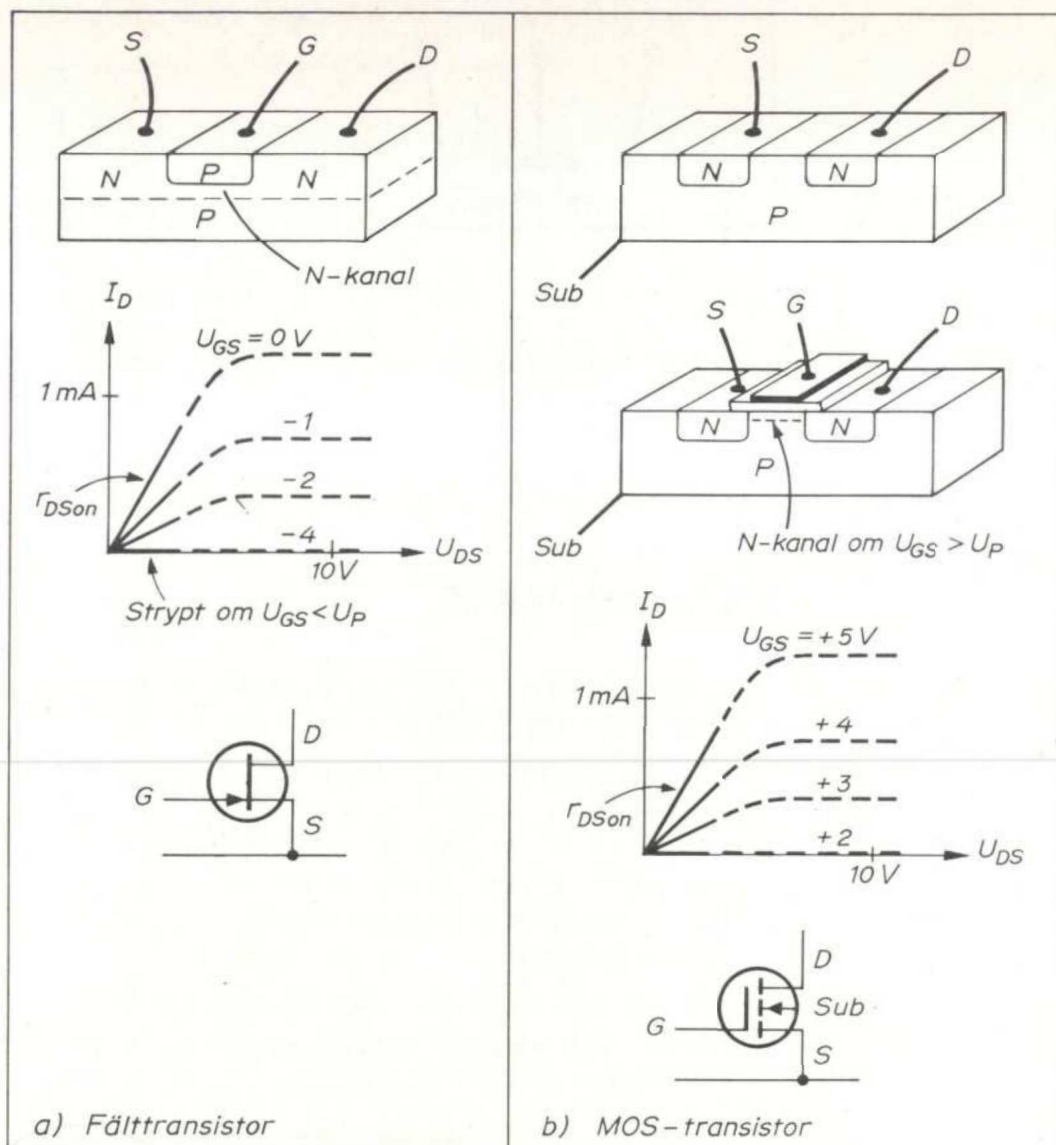
Vi har tidigare (kapitel 1 i Puls-kretsar) visat två typer av fälteffekttransistorer: fälttransistorn och MOS-transistorn. I figur 4.1 återges två kristallstrukturer (i princip), båda är av N-kanal typ. Som tidigare nämnts är MOS en akronym eller förkortning för metal oxide semiconductor, medan FET är en akronym för field effect transistor.

Fälttransistorn i figur 4.1a har vid tillverkningen utformats så att den innehåller en tunn kanal mellan emitter (source) och kollektor (drain). Denna kanal "avsnörs" eller stryps om man lägger backspänning på styret (gate). Strypning sker när styrspänningen uppnår pinch off-spänningen (eller strypspänningen) U_p som i figur 4.1a är $U_p = -4$ V.

MOS-transistorn har två åtskilda N-områden, S och D i figur 4.1b. Genom att lägga en (av ett tunt lager kiseldioxid) isolerad elektrod över P-området mellan de två N-områdena får man en kondensator. Metallelektroden utgör det ena belägget medan den underliggande P-kristallen utgör det andra. Läger vi nu positiv spänning på metallelektroden blir kristallområdet närmast under oxiden anriktat på elektroner. Vi har på kapacitiv väg "inducerat" en kanal mellan emitter och kollektor. För att denna kanal skall uppstå måste styrspänningen överstiga tröskelspänningen U_p som i figur 4.1b är $U_p = +2$ V.

Vi har tidigare sett hur man kan utnyttja fälttransistorn som en analog switch, som vid $U_{GS} = 0$ släpper fram signalen genom den lågohmiga kanalresistansen r_{DSon} .

I MOS-transistorn måste vi lägga spänning på styret för att kanalen skall bildas. Ju större spänning vi lägger på desto lägre blir härvid värdet på kanalresistansen r_{DSon} .

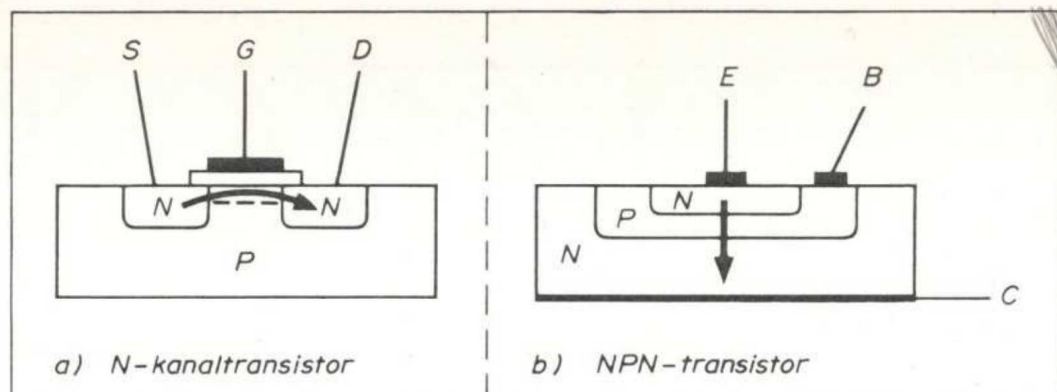


Figur 4.1

Kristallstrukturer (i princip) och diagram för fälttransistor och MOS-transistor.

Finessen med MOS-transistorn ligger bl a i att styrningen sker kapacitivt. Det kostar oss ingen styreffekt för att hålla MOS-transistorn i till- eller frånläge, till skillnad från den vanliga bipolära transistorswitchen. Däremot måste vi ju räkna med att det krävs en viss laddning vid omslag, eftersom styret är en kondensator.

I figur 4.2 jämför vi en MOS-transistor av N-kanaltyp med en bipolär NPN-transistor. Vi ser att elektronströmmen sker i kristallytan i MOS-transistorn men i kristallens inre i en bipolartransistor av planartyp. För att få en MOS-transistor att fungera bra, måste man tydligen kunna tillverka kristaller med mycket goda ytegenskaper. Det är just detta som varit så svårt att åstadkomma och därför har utvecklingen av MOS-transistorer tagit lång tid i anspråk jämfört med de planära bipolärtransistorernas snabba utveckling.



Figur 4.2

Elektronernas rörelse i en MOS-transistor av N-kanaltyp och en bipolär NPN-transistor.

2 PMOS och NMOS

När man i mitten av 1960-talet lyckats framställa stabila P-kanal MOS-transistorer, dvs PMOS-transistorer, låg vägen öppen för LSI-kretsar.

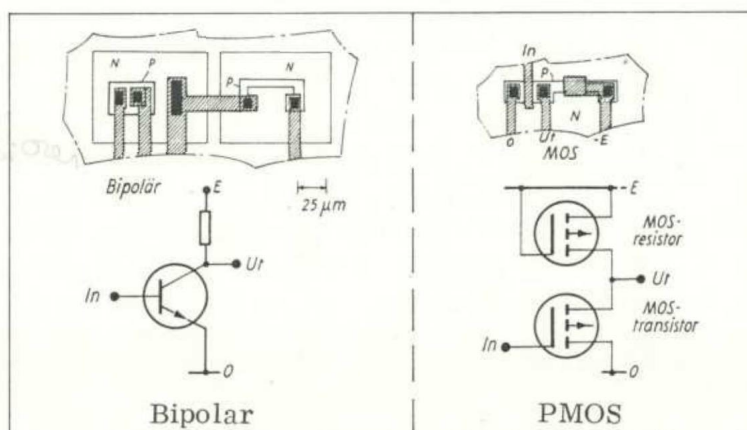
Det är flera faktorer som härvid är väsentliga. Kiselkristaller är alltid behäftade med kristallfel och därför blir "yielden" låg om man försöker göra kretsar som tar stor kristallarea i anspråk. En rimlig kompromiss mellan utbyte och pris var ca 4 mm² år 1970. Vill man på begränsad area få plats med många transistorer måste tydligen varje transistor göras så liten som möjligt.

Därtill kommer problemet med effektförbrukningen. En kiselkristall monterad i DIL-kapsel tål en effekt på ca 200 mW - mer kan man inte kyla bort utan att kristallen överskrider tillåten temperatur, ca 125 °C.

Kristallarea och effektkonsumtion är alltså två fundamentala begränsningar vid konstruktion av högintegrerade kretsar.

2.1 PMOS-teknologins fördelar

Vad har då PMOS-transistorn för fördelar jämfört med den bipolära? Figur 4.3 visar typiska kristallmönster för en bipolär switch och en MOS-switch. Vi ser att MOS-switchen enbart upptar en bråkdel av bipolarswitchens area. Eftersom hela PMOS-strukturen ligger med backspänning mot substratet behöver den inte isoleras mot omgivningen - de backspända PN-övergångarna, från de negativa P-områdena till det jorade N-substratet, är tillräcklig isolering.



Figur 4.3
Bipolar transistor-switch jämfört med MOS-switch.

Bipolartransistorn måste däremot som vi sett tidigare isoleras med P-kanaler (isolationsdiffusion) som tränger igenom det N-ledande epitax-skiktet och ner till P-substratet.

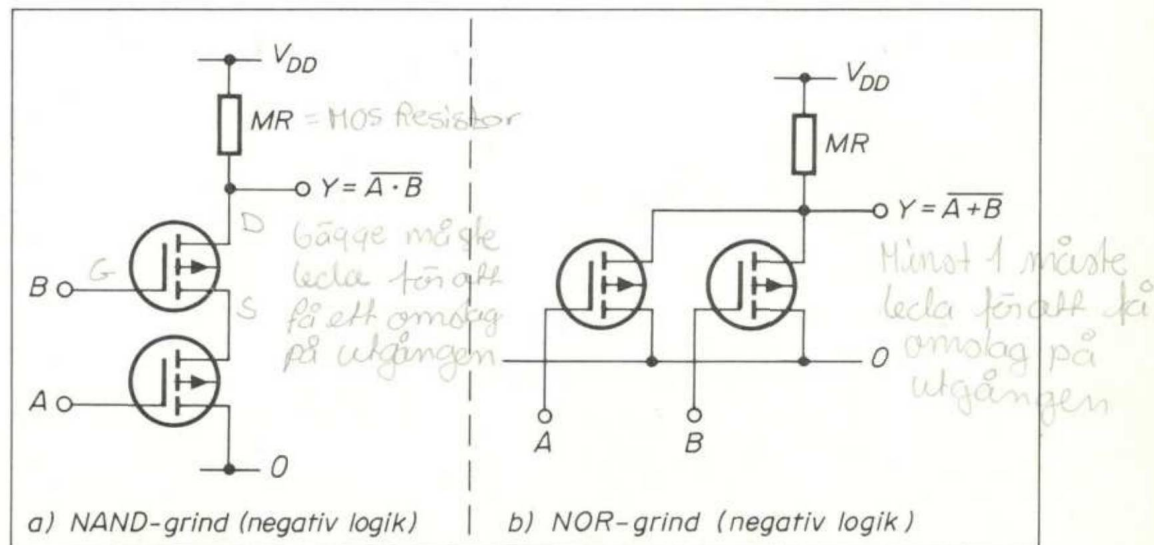
Förutom den avsevärda vinsten i kristallarea (upp till en tiopotens) har man en annan väsentlig fördel vid PMOS-teknologin. Vi har tidigare sett hur man vid bipolar teknologi behöver 6 masker och 4 diffusioner, figur 1.30. Vid PMOS behövs en enda diffusion, P-diffusionen. Det blir givetvis både billigare och ger större yield om man endast har få steg i tillverkningsprocessen.

Vi skall lägga märke till en detalj i figur 4.3. I MOS-switchen använder man en MOS-transistor med fast styrspänning som kollektor-resistor. Den tar mindre plats på kristallytan och önskat resistansvärde får man genom att välja lämplig kanalbredd på "MOS-resistorn".

MOS-transistorn behöver ingen styreffekt (kondensatorstyre !) och därför blir den avsevärt effektsnålare än en bipolar transistor. Däremot blir det en viss effektförlust i MOS-resistorn i kollektorkretsen.

2.2 Grindar och vippor i PMOS

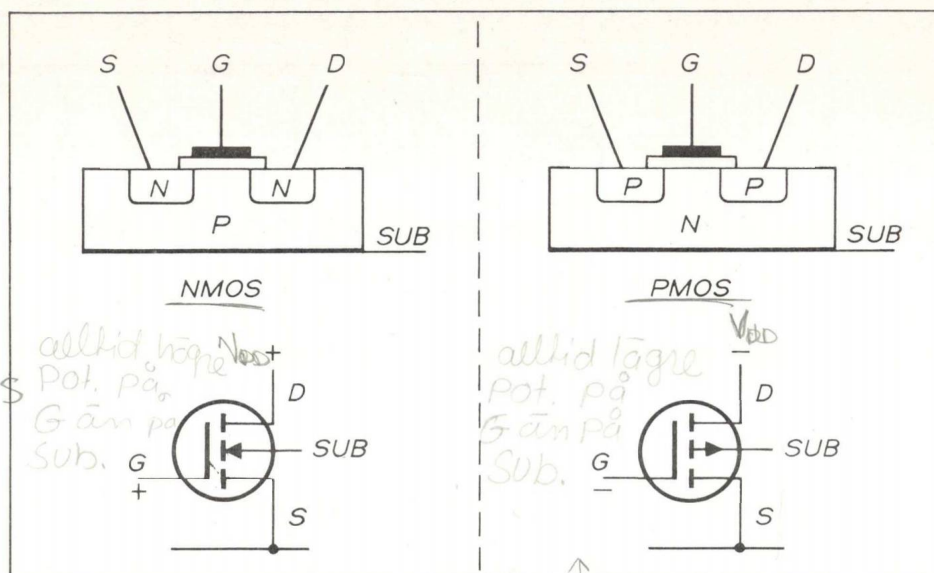
Figur 4.4 visar hur enkelt NAND- och NOR-grindar är uppbyggda i MOS-teknik. MR är liksom tidigare fast förspända MOS-transistorer som utnyttjas som kollektorresistorer. En PMOS-transistor har negativ kollektorspänning, exempelvis $V_{DD} = -15\text{ V}$, och det ligger därför nära till hands att här använda negativ logik.



Figur 4.4
NAND- och NOR-grindar i PMOS. (MR = MOS-resistor.)

En MOS-vippa kan bestå av två korskopplade MOS-switchar som i figur 4.5. Med grindar och vippor som grundkretsar kan man bygga upp alla de typer av kombinations- och sekvenskretsar som vi tidigare sett exempel på.

Normalt SUB till S
G hög transistor
bottnad



Figur 4.9
NMOS- och PMOS-transistorer.
S = emitter (source)
G = styre (gate)
D = kollektor (drain)
SUB = substrat.

Normalt läggs SUB till S
Transistor strypt när G ligger
på noll

I övrigt är NMOS och PMOS helt likartade i uppbyggnad och användning. I figur 4.9 jämförs de två strukturerna. NMOS som motsvarar NPN-transistorer matas med positiv kollektorspänning medan PMOS matas med negativ kollektorspänning. Substratet, dvs pilen, är normalt anslutet till emittern (source).

3 Principen för CMOS

Komplementära kopplingar har många fördelar tack vare den symmetriska uppbyggnaden. De har sedan länge flitigt utnyttjats i bipolartekniken (PNP- och NPN-transistorer i olika typer av komplementära steg).

Det stod redan från början klart att de största fördelarna med MOS-tekniken skulle man uppnå först när man kunde bygga komplementära switchar eller CMOS-switchar. Vi skall nedan först studera CMOS-switchens funktion och sedan visa uppbyggnaden av enkla NOR- och NAND-grindar i CMOS.

3.1 CMOS-switchen

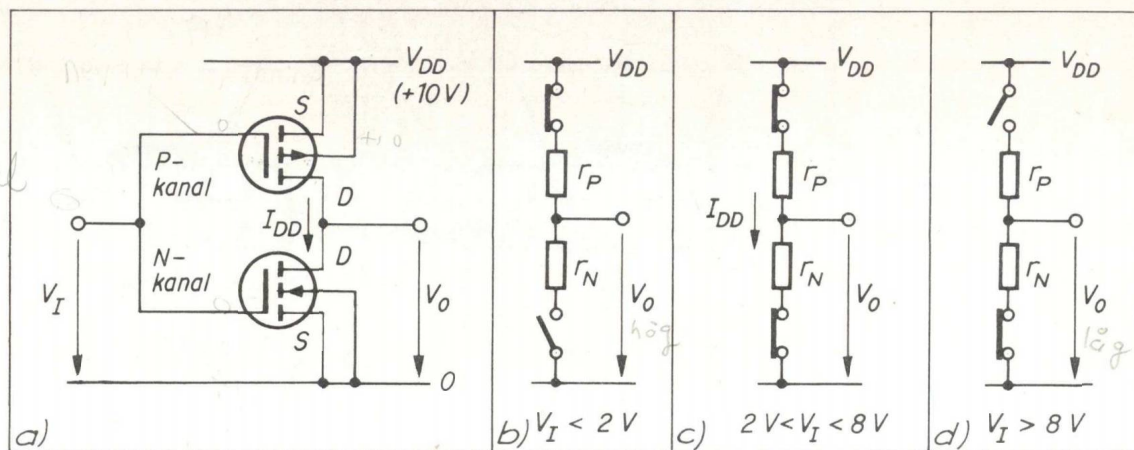
Figur 4.10a visar en CMOS-switch. PMOS-transistorns emitter är ansluten till matningsspänningen V_{DD} (exempelvis $V_{DD} = +10\text{ V}$) och NMOS-transistorns emitter är ansluten till jord. På ingången är styrena parallellkopplade och på utgången är kollektorerna (D) parallellkopplade.

Vi antar nu att transistorerna har lika stora strypspänningar:

$$\text{PMOS-transistor } U_p = -2\text{ V}$$

$$\text{NMOS-transistor } U_p = +2\text{ V}$$

D har alltid
lägre potential
än S.



Figur 4.10

CMOS-switchens funktion i princip ($|U_p| = 2 \text{ V}$).

= Varierbar resistans känslig
för statiska urladdningar

Se datablad?
när P och N,
är hög resp. låg

Är inspänningen låg ($V_I = 0 \text{ V}$) kommer den nedre NMOS-transistorn att vara strypt - medan den övre PMOS-transistorn ligger i tilläge, figur 4.10b. Utspänningen är därmed $V_C = V_{DD} = 10 \text{ V}$.

När inspänningen ökas till $V_I = +2 \text{ V}$ kommer den nedre PMOS-transistorn att börja leda. Som vi sett tidigare har en MOS-transistor en resistans r_{DSon} som varierar med strypspänningen.

Följande värden kan vara typiska för transistorerna i en CMOS-switch:

V_{GS}	5 V	10 V	15 V
r_{DSon}	800 Ω	500 Ω	300 Ω

I figur 4.10 b-d har vi lagt in två resistorer, r_P och r_N , som symboliserar PMOS-transistorn respektive NMOS-transistorns ledresistanser.

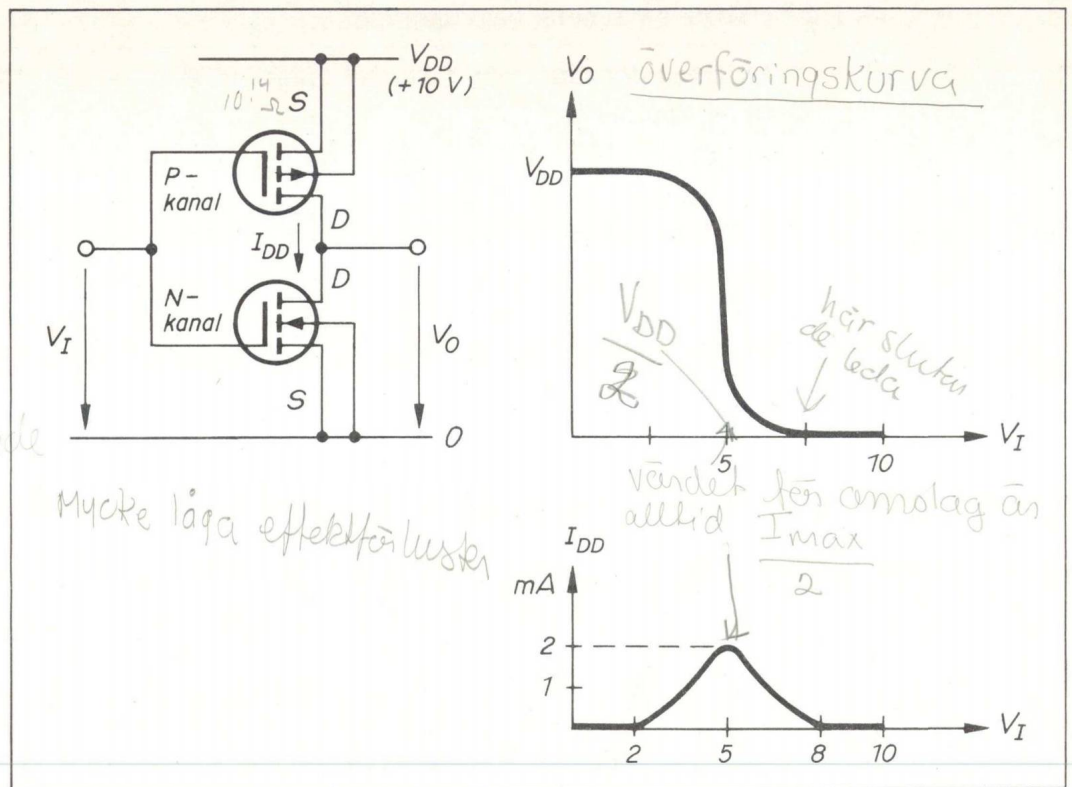
Mellan inspänningarna $V_I = 2 \text{ V}$ och $V_I = 8 \text{ V}$ leder tydligen båda transistorerna och om de är symmetriskt uppbyggda är $r_P = r_N$ vid inspänningen $V_I = 5 \text{ V} = 1/2 V_{DD}$. Därvid blir utspänningen $V_O = 1/2 V_{DD} = 5 \text{ V}$ på grund av spänningsdelningen mellan r_P och r_N . Kretsen arbetar alltså här i sitt aktiva område. Förstärkningen är hög och därför faller V_O -kurvan brant inom detta område, vilket framgår av figur 4.11. I denna figur ser vi även hur matningsströmmen I_{DD} får ett maximum för $V_I = 1/2 V_{DD} (= 5 \text{ V})$.

3.1.1 CMOS-switchens fördelar

Vad är då finessen med CMOS-switchen?

CMOS-switchen är lågimpediv såväl i läge hög som låg på utgången. Den har alltså samma goda egenskaper som TTL-grindens totempåle. Men den drar trots detta ingen förlusteffekt vare sig på ingång eller utgång i dessa lägen. Det är enbart vid omslag som CMOS-switchen drar ström ett kort ögonblick.

OBS!



Figur 4.11
CMOS-switchen och dess överföringsdiagram.

På grund av den symmetriska uppbyggnaden sker omslaget alltid vid $V_I = V_{DD}/2$. Man kan därför använda godtycklig matningsspänning (i praktiken begränsat till 3 - 20 V).

TTL-kretsens omslagsspänning beror som vi tidigare sett på två diodspänningsfall, $V_I \approx 2 \cdot 0,7 \text{ V} = 1,4 \text{ V}$, och för att hålla störmarginalerna under kontroll måste TTL därför matas med en fix spänning, $V_{CC} = 5 \text{ V}$ ($\pm 10 \%$). CMOS-kretsar ställer därför inte alls så stora fordringar på matningsspänningen som TTL-kretsar.

Ingången på en CMOS-switch är kapacitiv och av storleken 5 pF. Man kan därför belasta en CMOS-grind på utgången med ett godtyckligt antal likadana grindar. Belastbarheten är alltså mycket större jämfört med TTL-kretsar. Det enda man behöver tänka på är att stor kapacitiv last gör omslagen långsammare och ger förlusteffekt.

Som sammanfattning kan vi alltså konstatera att en CMOS-switch

- har symmetrisk överföringskurva och därför kan matas med godtycklig spänning V_{DD} (3 - 20 V)
- är lågimpediv både för $V_O = H$ och $V_O = L$
- har stor belastbarhet
- drar stationärt ingen effekt
- har hög inimpedans ($C = 5 \text{ pF}$)
- drar ström enbart vid omslag
- har lägre utimpedans vid högre matningsspänning V_{DD} .

Det är verkligen inte förvånande att CMOS är en stark konkurrent till TTL-kretsarna.

3.1.2 CMOS-switchens begränsning

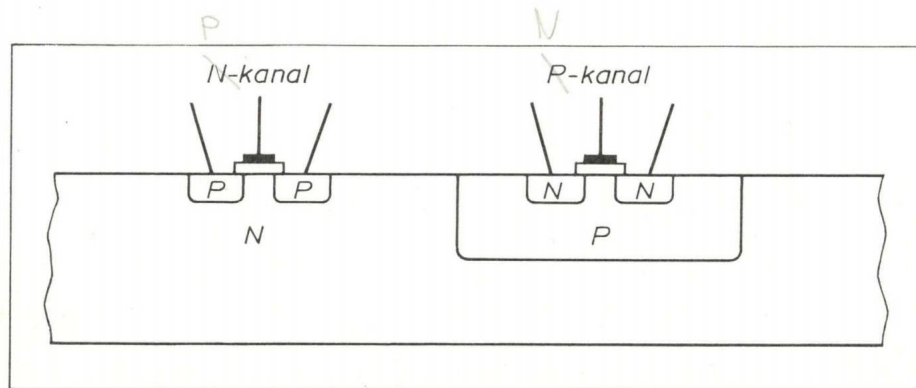
Nu finns det emellertid en rad omständigheter med CMOS som man måste se upp med.

En inimpedans på 5 pF och en inresistans på $10^{14} \Omega$ betyder att obetydliga laddningar kan ge avsevärda spänningar. Och därmed blir det genomslag i den tunna oxiden mellan styre och strömkanal. En CMOS-krets kan alltså förstöras enbart genom att man tar i den. Numera har man visserligen satt in skyddskretsar på alla ingångar, men känsligheten för laddningar vid beröring kommer man aldrig helt ifrån.

Jämfört med PMOS är CMOS en snabb krets. Räknefrekvenser på upp till 10 MHz kan användas. Snabbheten beror på matningsspänningen eftersom utimpedansen, och därmed tidkonstanten för omladdning av den kapacitiva lasten, minskar med ökande V_{DD} .

3.1.3 CMOS-switchens kristallstruktur

Det har varit ett mödosamt arbete att utveckla CMOS-teknologin. Även om man kan göra bra PMOS- och NMOS-transistorer var för sig, är det inte så enkelt att integrera dem i samma kristall. Figur 4.12 visar ett tvärsnitt genom CMOS-kristallen.

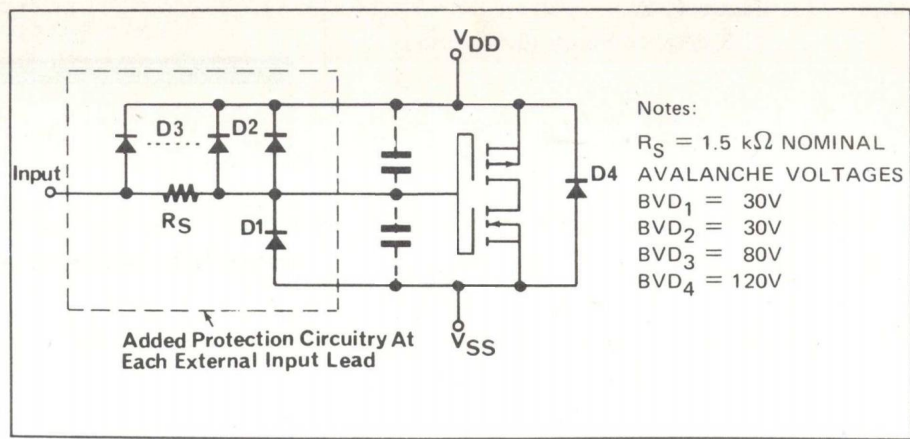


Figur 4.12
CMOS-strukturen.

För att åstadkomma denna struktur åtgår fler masker och fler diffusioner än vid PMOS och processen blir därför både dyrare och svårare att bemästra. För att få P-området för N-kanaltransistorn lagom dopat används ofta s k "joninplantering". Borjoner accelereras därvid och skjuts in i kristallen med stor hastighet. Figur 4.13 visar stegen i CMOS-processen.

En CMOS-switch innehåller, som framgår av figur 4.12, en rad PN-övergångar. Dessa fungerar som dioder vilka normalt ligger backspända. De medför emellertid kapacitiva effekter, parasit-element. Figur 4.14 visar fem av dessa parasitelement som normalt inte utritas men som alltid finns med i en CMOS-switch.

Figur 4.15 visar slutligen hur man försett ingången på en CMOS-switch med en skyddskrets av dioder (Added Protection Circuitry).



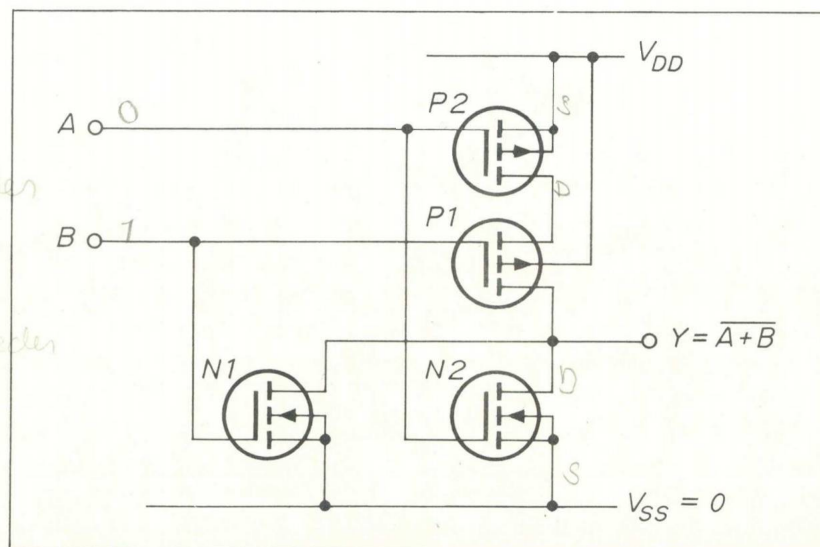
Figur 4.15
Skyddskrets på CMOS-ingång (Motorola).

Fabrikanterna har ytterligare en rad problem att ta hänsyn till. Man måste bli medveten om "stoppkanaler" mellan transistorerna för att förhindra ytläckströmmar, vilket framgår av figur 4.13. CMOS-strukturen blir därmed inte lika utrymmesbesparande som PMOS.

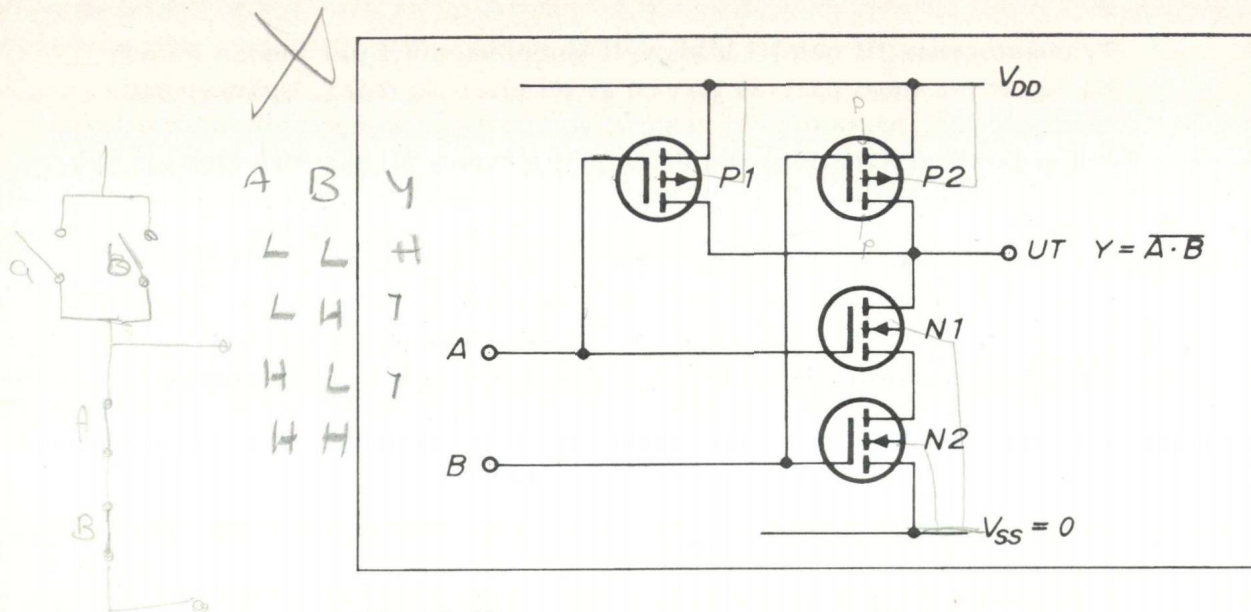
Utvecklingen av CMOS-kretsar befinner sig 1975 i ett expansivt skede. Olika teknologier provas för att öka såväl elementtätheten som snabbheten. Ett sätt är härvid att bygga upp PMOS- och NMOS-transistorer på ett isolerande underlag av safirkristall. Denna teknik kallas SOS (silicon on sapphire).

3.2 NAND- och NOR-grindar i CMOS

CMOS-kretsar kan enkelt kombineras till NAND- och NOR-grindar. Figur 4.16 visar en NOR-grind. Alla NMOS-transistorer har substraten jordade och alla PMOS-transistorer har substraten anslutna till V_{DD} .



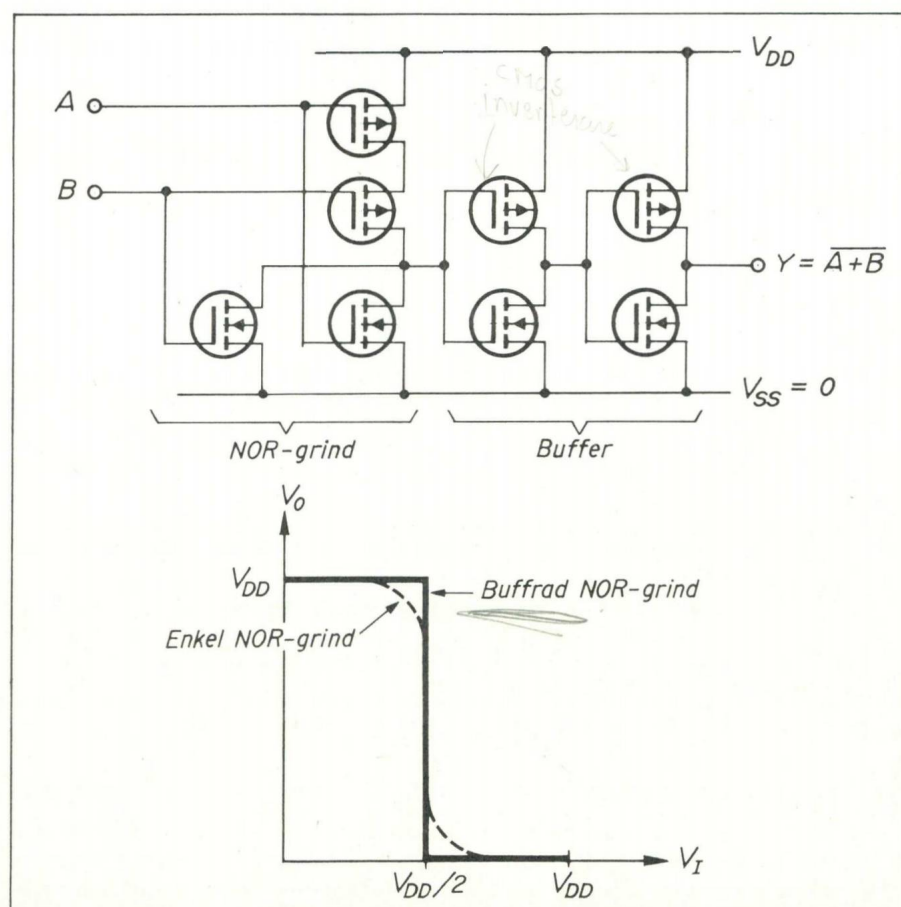
Figur 4.16
NOR-grind med två ingångar.



Figur 4.18
NAND-grind i CMOS.

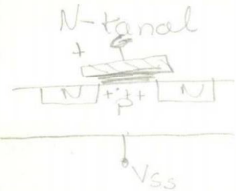
Om man parallellkopplar PMOS-transistorerna och seriekopplar NMOS-transistorerna så som framgår av figur 4.18 får man en NAND-grind. Likheten med den tidigare NOR-grinden i figur 4.16 är påfallande.

För att få bättre överföringsdiagram har man infört "buffrade grindar". Figur 4.19 visar ett exempel. NOR-grinden matar här två kaskadkopplade CMOS-switchar och förstärkningen blir härigenom mycket hög, 10 000 ggr. Överföringskurvan får en nästan ideal form.



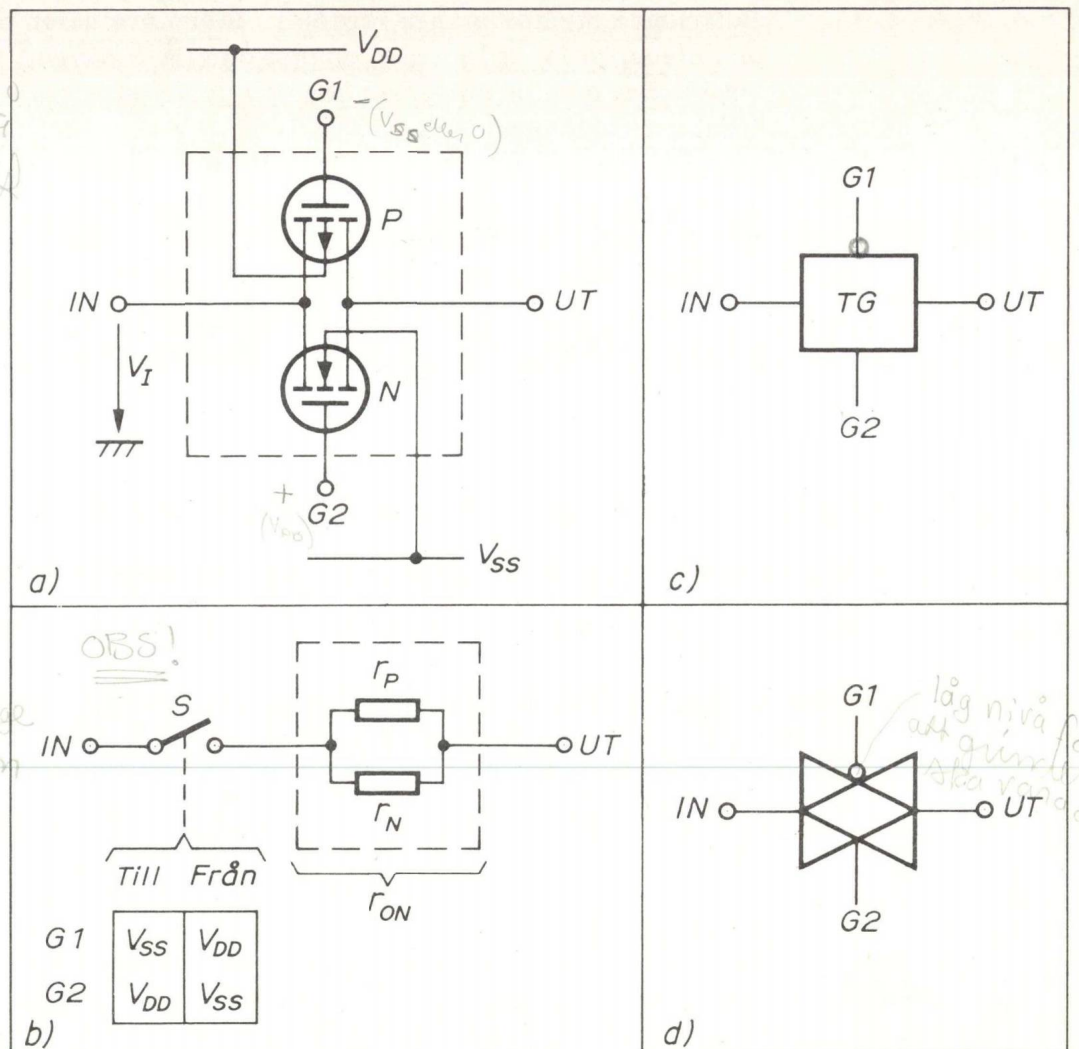
Figur 4.19
Buffrad CMOS-grind (Fairchild).

en spänning på
styret gör att
den leder



Denna leder ström
om G lägg till +

en transistor i tillägg
representeras av en
resistans r_{ON}



Figur 4.21
Transmissionsgrunden TG.

En PMOS + 1 NMOS används för
att få en linjär totalresistans
 r_{ON}

Transmissionsgrunden är ströpt om G1 och G2 ges spänningarna

$$\left. \begin{array}{l} G1: V_{DD} \\ G2: V_{SS} \end{array} \right\} \text{TG ströpt (spännt)}$$

Substratet och styret hos P ligger då på samma spänning och transistorn P är ströpt. Samma är förhållandet med N. I figur 4.21b har vi markerat detta med en switch S som därvid ligger i frånläge. Resistansen i frånläge är av storleksordningen $10^9 \Omega$.

Transmissionsgrunden är ledande om G1 och G2 ges spänningarna

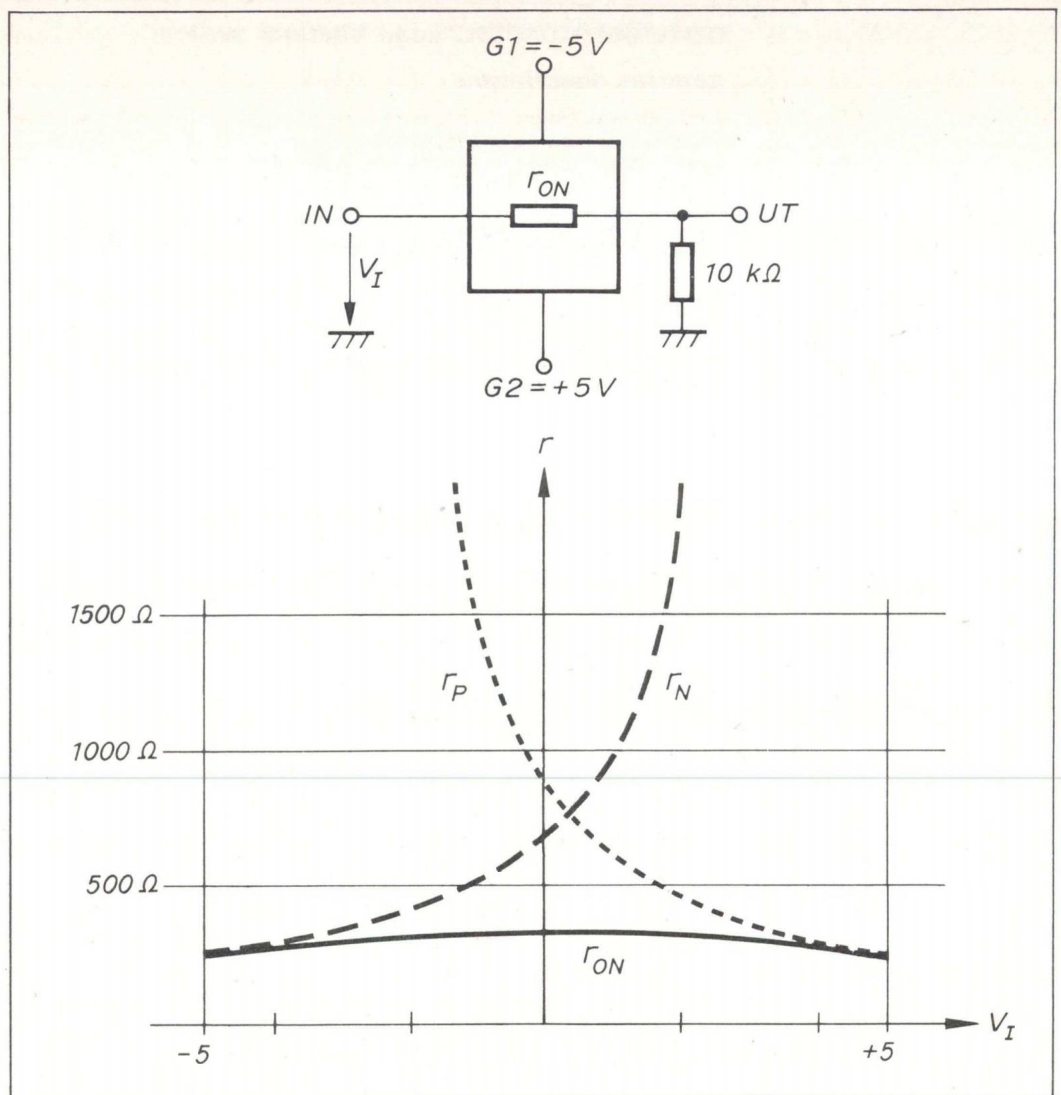
$$\left. \begin{array}{l} G1: V_{SS} \\ G2: V_{DD} \end{array} \right\} \text{TG till (öppent)}$$

Om inspänningen V_I varierar kommer styrspänningarna på P och N att ändras på sådant sätt att P blir mer ledande om N blir mindre ledande och vice versa.

Totala ledresistansen

$$r_{ON} = r_P // r_N$$

kan på detta sätt hållas relativt konstant inom hela inspänningsområdet.



Figur 4.22

Transmissionsgrindens serieresistans r_{ON} .

resistanserna är rel. konst. om man ändrar inspänningen från plus till minus. Samma signal som sänds in kommer ut.

Figur 4.22 visar ett exempel på hur resistanserna r_N och r_P varierar i en transmissionsgrind som matas med spänningarna

$$\left. \begin{array}{l} V_{DD} = +5 \text{ V} \\ V_{SS} = -5 \text{ V} \end{array} \right\}$$

Olika fabrikanter använder olika grafiska symboler för transmissionsgrinden. Motorola använder två motriktade pilar, figur 4.21d, och vill därmed markera att transmissionsgrinden kan arbeta i båda riktningarna (bidirectional switch). När den ringmarkerade ingången är låg och den andra ingången hög leder transmissionsgrinden.

Transmissionsgrindar kan köpas separata, fyra i en kapsel (RCA typ CD 4016) och används ofta som switchar för analoga signaler. I kapseln ingår därvid även inverterare så att man enbart behöver en styringång. Figur 4.23.

även använda grinden till att sampla insignalen

CD4016M/CD4016C quad bilateral switch

general description

The CD4016M/CD4016C is a quad bilateral switch which utilizes P-channel and N-channel complementary MOS (CMOS) circuits to provide an extremely high "OFF" resistance and low "ON" resistance switch. The switch will pass signals in either direction and is extremely useful in digital switching.

features

- Wide supply voltage range 3V to 15V
- High noise immunity 0.45 V_{CC} typ.
- Wide range of digital and analog levels $\pm 7.5 V_{PEAK}$
- Low "ON" resistance 300 Ω typ.
 $V_{DD} - V_{SS} = 15V$
- Matched switch characteristics $\Delta R_{ON} = 40\Omega$ typ.
- High "ON/OFF" output voltage ratio 65 dB typ.
@ $f_{is} = 10$ kHz
 $R_L = 10k$
- High degree of linearity .5% distortion typ.
@ $f_{is} = 1$ kHz

- Extremely low leakage

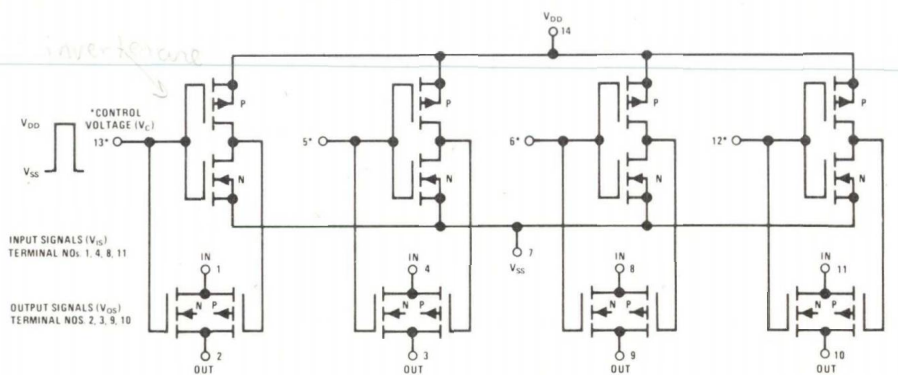
$$\begin{aligned} V_{IS} &= 5 V_{D-P} \\ V_{DD} - V_{SS} &= 10V \\ R_L &= 10 k\Omega \end{aligned}$$

- Transmits frequencies up to 10 MHz

applications

- Analog signal switching/multiplexing
 - Signal gating
 - Squelch control
 - Chopper
 - Modulator
 - Demodulator
 - Commutating switch
- Digital signal switching/multiplexing
- CMOS logic implementation
- Analog to digital/digital to analog conversion
- Digital control of frequency, impedance, phase, and analog-signal gain

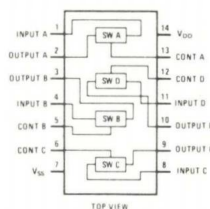
schematic and connection diagrams



Note 1: All switch P-channel substrates are internally connected to terminal No. 14.
Note 2: All switch N-channel substrates are internally connected to terminal No. 7.

Signal-level range: $V_{SS} < V_{in} < V_{DD}$

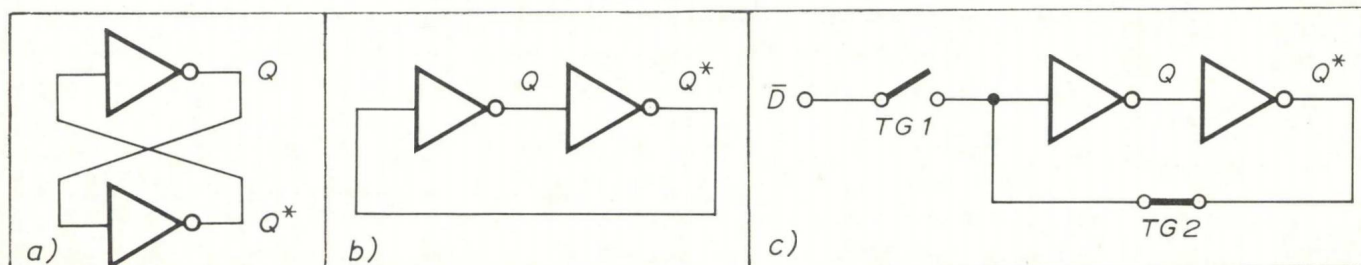
Normal operation: Control-line biasing, switch ON $V_c = V_{DD}$, switch OFF $V_c = V_{SS}$



Figur 4.23
Fyra transmissionsgrindar i samma kapsel.
(National Semiconductor CD 4016.)

4.2 MS-vippor i CMOS

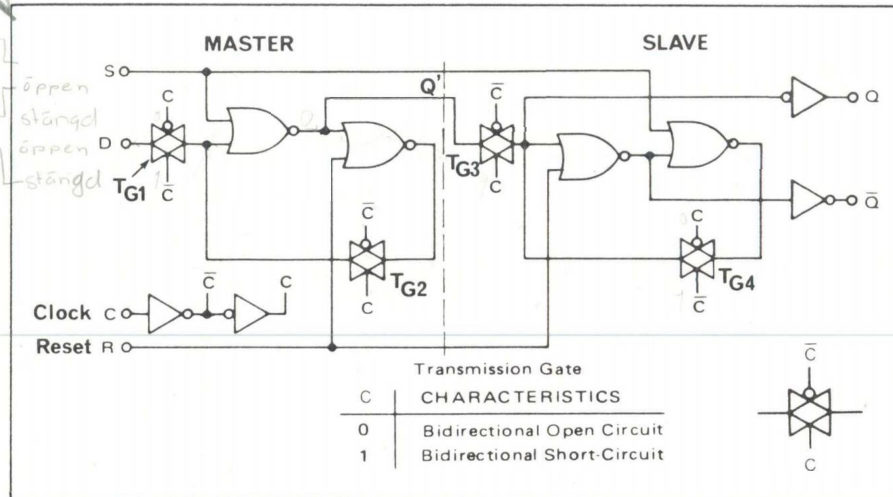
Med transmissionsgrindar (TG) kan man göra enkla CMOS-vippor. Figur 4.24 visar principen.



Figur 4.24
CMOS-vippa av D-typ.

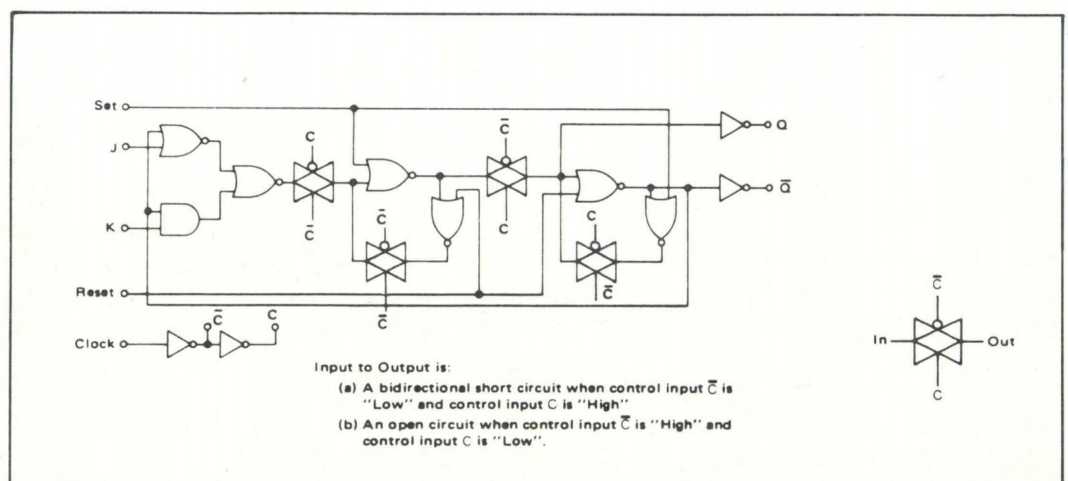
Figur 4.24a visar två inverterare - kan också vara NOR- eller NAND-grindar - som korskopplats och bildar en vippa. Figur 4.24b visar samma vippa men utritad så som man ofta ser CMOS-vipporna i datablad. I figur 4.24c har vi slutligen satt in två transmissionsgrindar, TG1 och TG2. TG2 ligger i normalläget till medan TG1 ligger från.

För att läsa in information från D-ingången till vippan ändrar vi ett ögonblick polariteten på transmissionsgrindarnas styringångar så att TG1 går till och TG2 går från. När transmissionsgrindarna åter ställs i normal-läge intar vippan det läge som under klockpulsen angavs av D-ingången.



Figur 4.25
MS-vippa av D-typ. (Motorola MC 14013)

Figur 4.25 visar en MS-vippa av D-typ uppbyggd med vippor av den ovan beskrivna typen. Här ingår NOR-grindar i stället för inverterare. Härigenom kan man enkelt införa direktverkande set- (S) och reset- (R) ingångar. Två vippor av den typ som visas i figur 4.25 ingår i kretsen CD 4013 från RCA eller MC 14013 från Motorola. Med en enkel modifiering av ingången till mästaren i figur 4.25 får man en JK-MS-vippa, figur 4.26.



Figur 4.26
JK-MS-vippa (Motorola MC 14027).

5 CMOS-kretsar och deras specifikation

Ett byggsystem av digitalkretsar får vidsträckt användning först när flera fabrikanter tillverkar samma kretsar. Användarna har då "second source" och därigenom större trygghet om någon av tillverkarna får leveranssvårigheter.

CMOS-kretsarna kom först från RCA (1969) men det var inte förrän Motorola ett par år senare gav sig in på CMOS-marknaden och man därigenom fick second source som det blev vind i seglen för CMOS. I dag tillverkar ett stort antal fabrikanter CMOS.

RCA:s 4000-serie har blivit normgivande på marknaden. Serien omfattar mer än 100 kretsar. Dessa tillverkas i olika kapslar, plast, keramik och "militärkeramik" och i två olika spänningsnivåer, 3 - 15 V respektive 3 - 20 V.

Det finns förutom 4000-serien en serie 4500 som innehåller ett stort antal MSI-kretsar (MSI = medium scale integration).

Motorola har dessutom en serie MC 14400 som innehåller system eller subsystem, exempelvis hela elektroniken till en digitalvoltmeter.

National Semiconductor (NS) har introducerat 47C-serien. Det är en serie CMOS-kretsar med samma funktioner, och även samma uttagsplacement, som motsvarande TTL-kretsar.

Försäljningen av CMOS-kretsar i Sverige rör sig om ca 10 Mkr (1975) och ökar starkt. Medelpriset på dessa CMOS-kretsar ligger mellan 1,50 och 2,00 kr (i stora antal).

5.1 Nivåer och störmarginaler

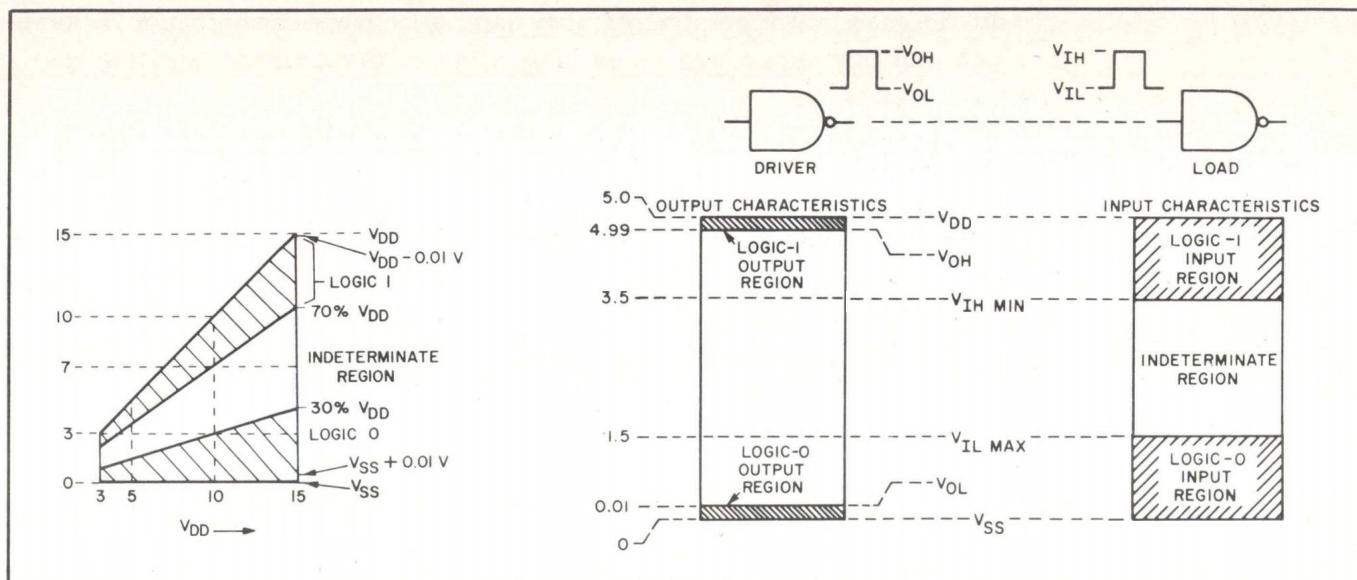
Eftersom CMOS-kretsar kan matas med godtycklig matningsspänning - (inom ett begränsat område, exempelvis 3 - 15 eller 3 - 20 V) kan inte spänningsnivåerna på utgången anges i absoluta tal på det sätt vi tidigare mött hos TTL-kretsarna. Utnivåerna måste därför anges i relation till matningsspänningen V_{DD} .

Belastningen av CMOS-grindar är kapacitiv och därför kommer spänningsnivåerna på utgången att bli

$$\left. \begin{array}{l} V_{OH} = V_{DD} \\ V_{OL} = 0 \end{array} \right\}$$

och detta gäller med en marginal på endast några få mV.

Störmarginalerna hos en buffrad grind, figur 4.19, kommer att ligga mycket nära $0,5 V_{DD}$. De enkla grindarna har en avrundad överföringskurva och därmed blir störmarginalen något lägre. Fabrikanterna brukar garantera störmarginaler på 30 % av matningsspänningen - "for which the output level does not change state" som det heter. Vad som exakt händer på utgången specificeras emellertid olika av olika fabrikanter. Här saknas fortfarande internationell standard (1975). Figur 4.27 visar hur RCA anger störmarginaler för sin 4000-serie.



Figur 4.27
Garanterade störmarginaler för 4000-serien (RCA).

Normalt skall en CMOS-grind endast mata kapacitiv belastning. För det fall man har resistiv belastning, specificeras även den utström kretsen kan ge. Motorola anger exempelvis i familjespecifikationen för 14 000-serien, dvs samma som RCA:s 4000-serie, följande typiska värden:

$V_{DD} = 10 \text{ V}$	$V_{OH} = 9,5 \text{ V}$	$I_{OH} = -0,9 \text{ mA (source)}$
	$V_{OL} = 0,5 \text{ V}$	$I_{OL} = 2,0 \text{ mA (sink)}$

5.2 Snabbhet

Stegfördröjningen hos en CMOS-grind beror dels på matningsspänningen V_{DD} dels på den kapacitiva belastningen. Kretsen fungerar ju som en RC-krets. Resistansen bestäms av kanalresistanserna r_P eller r_N vilka båda avtar med V_{DD} . Kapacitansen bestäms av antalet belastande kretsar.

Motorola anger exempelvis följande familjevärden för MC 14 000-serien vid $V_{DD} = 5 \text{ V}$ och $C_L = 50 \text{ pF}$:

$$\begin{cases} t_r = 180 \text{ ns} & (\text{typiskt värde}) \\ t_f = 130 \text{ ns} & (\text{typiskt värde}) \end{cases}$$

Vid 15 V och $C_L = 15 \text{ pF}$ sker omslagen snabbare:

$$\begin{cases} t_r = 25 \text{ ns} & (\text{typiskt värde}) \\ t_f = 25 \text{ ns} & (\text{typiskt värde}) \end{cases}$$

om belastningskapacitansen minskar ökar snabbheten

För vippor ligger maximala räknefrekvensen normalt vid $5 - 10 \text{ MHz}$.

5.3 Förlusteffekt

Eftersom ingen ström flyter stationärt genom en CMOS-switch (frånsett små läckströmmar) blir den stationära förlusteffekten försumbart liten, 10^{-8} W per grind. Omladdningen av kapacitanser medför emellertid, som

vi sett tidigare, en förlusteffekt som både blir proportionell mot frekvensen och mot den kapacitiva belastningen. Den "dynamiska" förlusteffekten kan tecknas

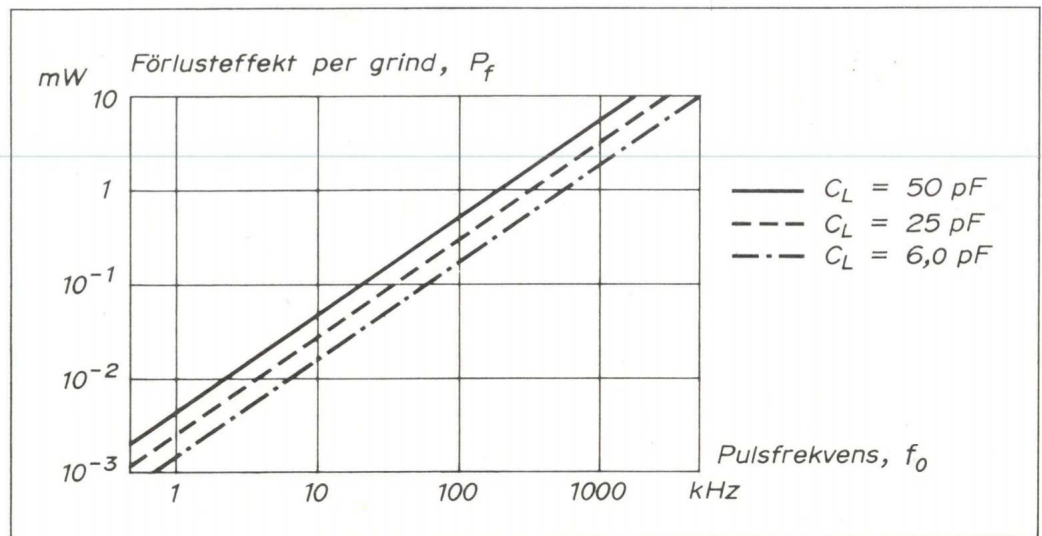
$$P_f = V_{DD}^2 \cdot f \cdot C$$

V_{DD} = matningsspänning

f = frekvens

C = kapacitiv belastning

Figur 4.28 visar kurvor för förlusteffektens variation med frekvensen. Lägg märke till att vid en frekvens något över 1 MHz och vid 50 pF belastning är förlusteffekten för en CMOS-grind 10 mW, dvs lika stor som hos en standard TTL-grind.



Figur 4.28

Förlusteffektens beroende av frekvensen i CMOS-grind. (Motorola.)

De flesta kretsar i ett digitalsystem arbetar emellertid med mycket lägre frekvenser. Effektbesparingen, och därmed minskade problem med kylningen, är därför avsevärd om man jämför CMOS med TTL.

Utförligare behandling av CMOS-kretsar återfinns i fabrikanternas manualer, exempelvis:

RCA COS/MOS Integrated Circuits Manual
RCA Solid State Division Sommerville NJ 08876

McMOS Handbook, Products - Characteristics - Applications, Motorola Semiconductors.

Repetitionsfrågor

1. Hur skiljer sig en MOS-transistor, en fälttransistor och en bipolar-transistor med avseende på inimpedans?
2. Vilka fördelar har NMOS framför PMOS?
3. Jämför ett statiskt skiftregister med MS-vippor och ett dynamiskt PMOS-register. Likheter och skillnader?
4. Varför leder CMOS-switchen ström vid $V_I = 1/2 V_{DD}$ (om V_{DD} är tillräckligt stor)?
5. Ange tre egenskaper för vilka CMOS-switchen är överlägsen bipolar-switchen.
6. Varför kan överföringskurvans form variera med antalet höga ingångar i en NOR-grind?
7. Hur har man lyckats hålla impedansen konstant vid varierande V_I i en transmissionsgrind?
8. Rita en MS-vippa med NOR-grindar och transmissionsgrindar.
9. Hur stora är vanligen de garanterade störmarginalerna för CMOS?
10. Varför ökar förlusteffekten i CMOS med frekvensen?

D5 Halvledarminnen

Halvledarminnen 116

1 Några viktiga minnestermer 116

- 1.1 Serieminnen och random access-minnen 117
- 1.2 ROM och RAM 118

2 Serieminnen 120

- 2.1 Skiftregister och fördröjningsledningar 120
- 2.2 Bucket brigade 121
- 2.3 Laddningskopplare 122
- 2.4 LIFO och FIFO 123

3 Läsminnen, ROM 124

- 3.1 Principen för ett läsminne 124
- 3.2 Programmering av ROM 129

4 Skriv-läsminnen, RAM 132

- 4.1 Statiska RAM 133
- 4.2 Dynamiska RAM 140

5 Mikrodatorsystem 145

Repetitionsfrågor 147

Målruta

När du läst detta kapitel skall du kunna:

- beskriva ett minnes organisation med avseende på adressering, ordlängd och kapacitet
- ange för- och nackdelar med statiska kontra dynamiska minnen
- beskriva funktionen hos ett ROM, fPROM och rader-PROM
- rita scheman för statiska minnesceller till RAM-minnen av TTL- och MOS-typ
- ange för- och nackdelar med PMOS, NMOS, CMOS och bipolär teknologi när det gäller uppbyggnad av halvledarminnen

Halvledarminnen

Minnet utgör ett väsentligt byggblock i en dator. Sedan 1950-talet har kärnminnen dominerat som primärminnen (main frame memories) men i början av 1970-talet hade integrerade halvledarminnen utvecklats så långt att de kunde ta upp konkurrensen med kärnminnena. År 1975 har halvledarminnena passerat kärnminnena både i snabbhet och pris. Halvledarminnen är i dag den mest expansiva sektorn inom halvledartekniken.

Vi skall börja detta kapitel med att klargöra några viktiga begrepp och särskilja några viktiga grundtyper av minnen, exempelvis serieminnen och random access-minnen.

Serieminnen har vi tidigare mött i form av skiftregister. Det finns flera nya typer av serieminnen, exempelvis CCD (charge coupled device) eller laddningskopplare.

Läsminnen (ROM = read only memory) har fått stor användning både för programlagring och som kombinationskretsar. Vi skall se hur data kan lagras permanent i olika typer av ROM.

Skriv-läs-minnen (RAM = random access memory) förekommer både som statiska minnen (med vippor) och som dynamiska (med kondensator som minneselement). Vi skall se hur minnescellerna är uppbyggda i några vanliga RAM.

Som avslutning skall vi bekanta oss med byggblocken i ett mikrodatorsystem. Det är i sådana system de flesta halvledarminnena kommer att sitta i framtiden.

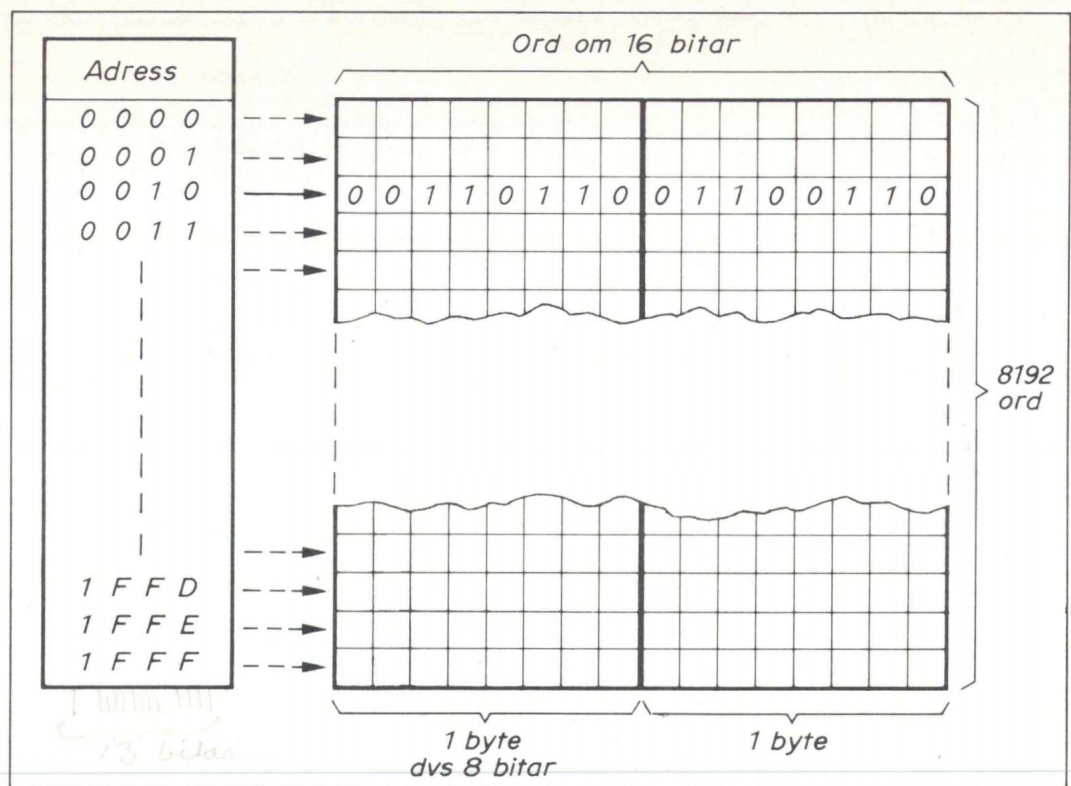
1 Några viktiga minnestermer

Ett minne är lagringsplats för kodad information - minne kallas på engelska store eller memory. Informationen lagras i form av ord (word), men ord har här en annan betydelse än vi är vana vid. Ett "ord" i ett minne utgörs av ett antal bitar. Minidatorer arbetar vanligen med en ordlängd av 16 bitar medan de flesta mikrodatorer arbetar med ord om 8 bitar. Åtta bitar kallas ofta 1 byte (uttalas bajt). Varje ord i ett minne har en given adress. Adressen anger alltså var i minnet ordet är lagrat.

Med kapaciteten hos ett minne avser man antalet ord som kan lagras. En minidators minne kan exempelvis vara av storleken 8192 ord om 16 bitar och man säger då att det har kapaciteten "8 k". Kilo, dvs k, har alltså betydelsen $k = 2^{10} = 1024$ när man talar om minneskapacitet.

Figur 5.1 visar exempel på ett minne för en minidator. Det består av ett antal "fack", vardera innehållande ett "ord". Varje fack har en adress och till vänster i figuren är några adresser markerade i hexadecimal kod. Minnet i figur 5.1 har kapaciteten 8 k (dvs högsta adressen $1FFF_{16}$).

Man använder termen "skriva" när man lagrar ett ord i minnet och "läsa" när man vid ett senare tillfälle tittar efter vad en minnescell innehåller. Med "cell" menar man ofta ett helt fack, 16 bitar i figur 5.1, när det gäller programmering av datorer. Med "cell" i en halvledarkristall avser man normalt en vipa eller minnescell om en bit på kristallen. Cell är alltså ett mångtydigt begrepp.



Figur 5.1

Exempel på ett minne med kapaciteten 8 k och ordlängden 16 bitar.

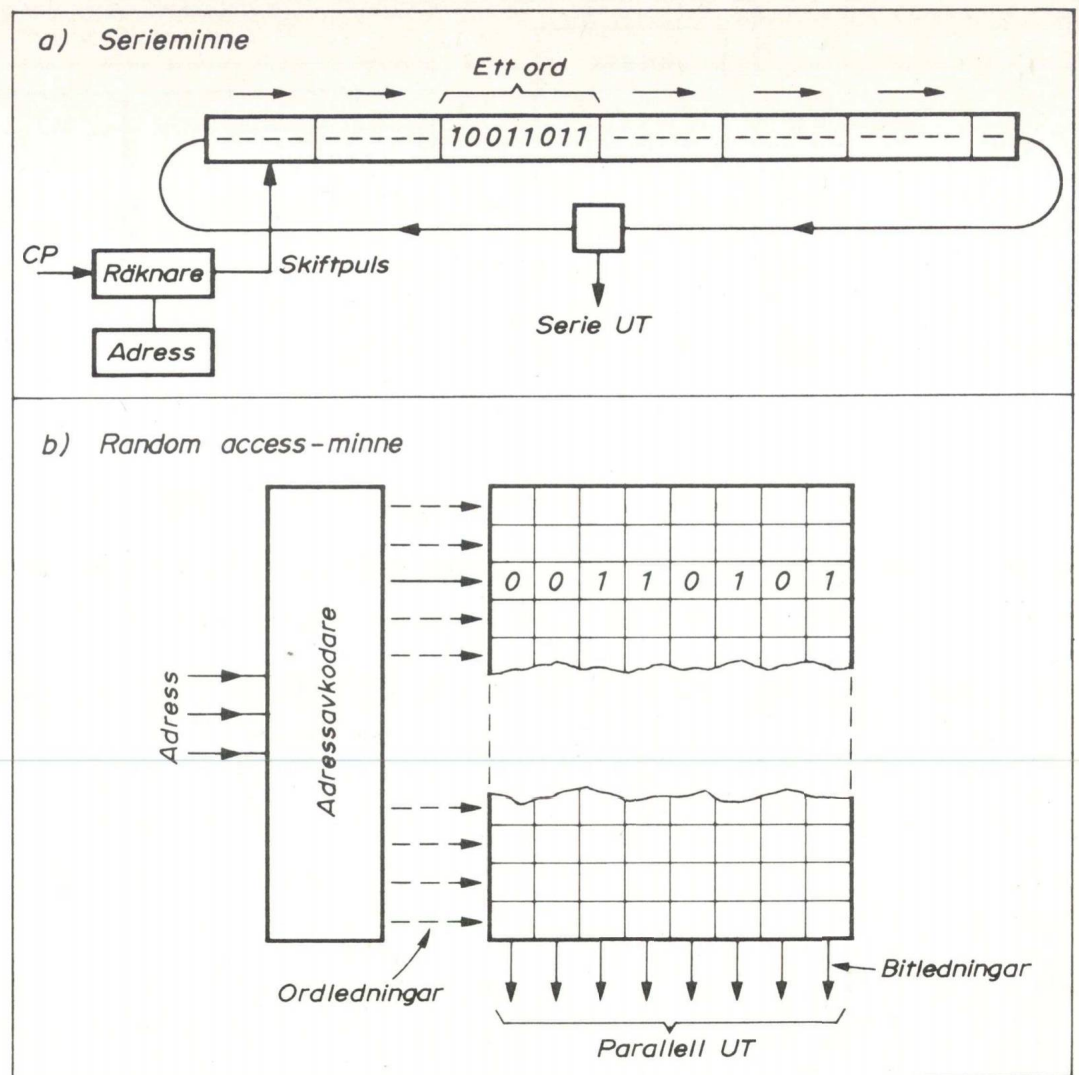
ord finns på en adress
13 bitar på adressen ger $2^{13} = 8192$ Adresser $\Rightarrow$ 8k ord
 $= 8 \cdot 1024$
 $1024 = 2^{10}$

1.1 Serieminnen och random access-minnen

En väsentlig fråga är hur man "plockar fram" ett ord ur ett minne. Man kan ha orden placerade som på ett "radband" som man läser av, kula för kula. Man kan också ha orden placerade som på ett schackbräde och direkt plocka upp den pjäs man vill använda. När det gäller minnen talar man om serieminnen eller sekventiella minnen och random access minnen (RAM). I ett serieminne måste man skifta och rotera ett antal steg för att komma åt önskat innehåll. I ett random access minne kan man direkt plocka fram innehållet från en önskad cell ur minnet, oberoende av var cellen är placerad.

Figur 5.2 visar principen för ett serieminne och ett random access minne.

Ett serieminne kan, som i figur 5.2a, bestå av ett långt skiftregister där orden ligger lagrade efter varandra i en följd. För att få adressen måste man räkna skiftpulserna och räknarens ställning kan därför direkt användas som adress. Åtkomsttiden (eller accesstiden) till ett ord i ett serieminne beror tydligen på var ordet just uppehåller sig i minnet när vi börjar söka det. Vid utläsning får vi ordet i serieform på utgången i takt med skiftpulserna.



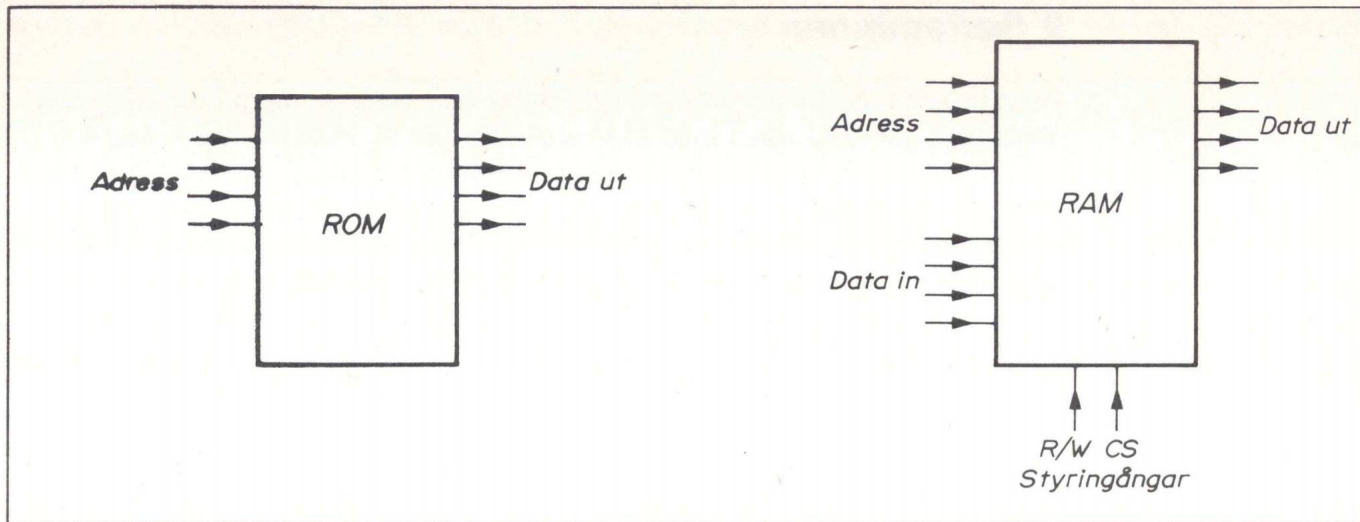
Figur 5.2
Serie- och random accessminnen (princip).

I random access minnet, figur 5.2b, lägger vi på adressen till det sökta ordet på adressgångarna. Adressavkodaren väljer då ut den aktuella "ordledningen" som kopplar in det utvalda ordet till de gemensamma bitledningarna. Vi får ordet i parallell form på utgången. Accesstiden är tydligen oberoende av var i minnet ordet är placerat.

1.2 ROM och RAM

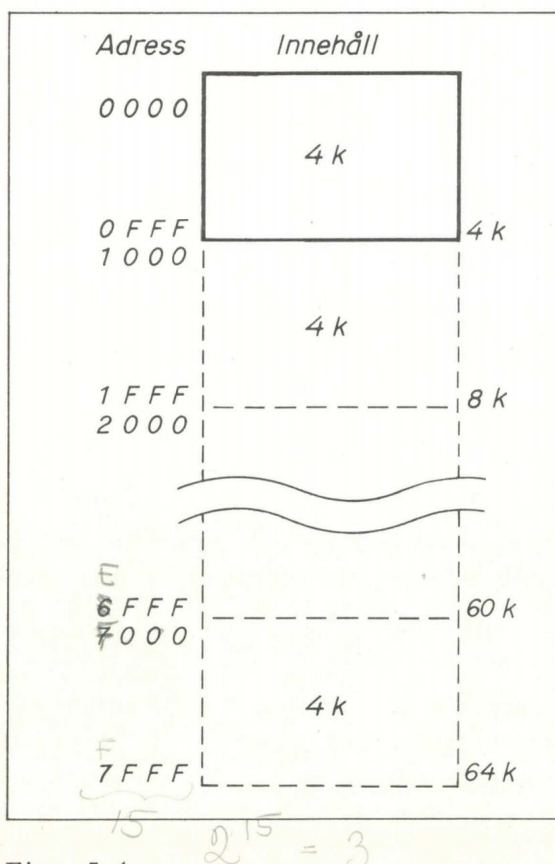
Man använder ofta beteckningen ROM (read only memory) eller läsminne för ett minne där informationen är permanent lagrad. Principen för ett läsminne framgår av figur 5.3a. Ett ROM är egentligen ett random access minne som enbart kan läsas. Man lägger på en adress på ingången och ut kommer det adresserade ordet (data ut).

Ett skriv-läsminne eller RAM (random access memory) har liksom ROM:et en adressgång och en datautgång men därutöver har ett RAM både en dataingång och en kontrollgång. Med den senare ger man minnet order om att det skall läsa den adresserade cellen (läs = read = R) eller skriva in (skriv = write = W) dataingångens inställning i den adresserade cellen. Beteckningen RAM avser vanligen skriv-läsminnen trots att denna beteckning är lika adekvat för ett läsminne (ROM).



Figur 5.3
ROM och RAM.

Vill man utöka ett minnes kapacitet kan man antingen öka antalet ord, och därmed antalet adresser, eller också öka ordlängden. Vanligen arbetar digitala system med given ordlängd och därför utökas vanligen kapaciteten genom att fler ord med samma ordlängd kopplas in. En minidator med 16 bitars ordlängd kan med ett 16-bitars ord adressera $2^{16} = 65536$ celler, dvs 64 k ord. Ofta är ett sådant minne uppbyggt i moduler om 4 k eller 8 k ord. Figur 5.4.



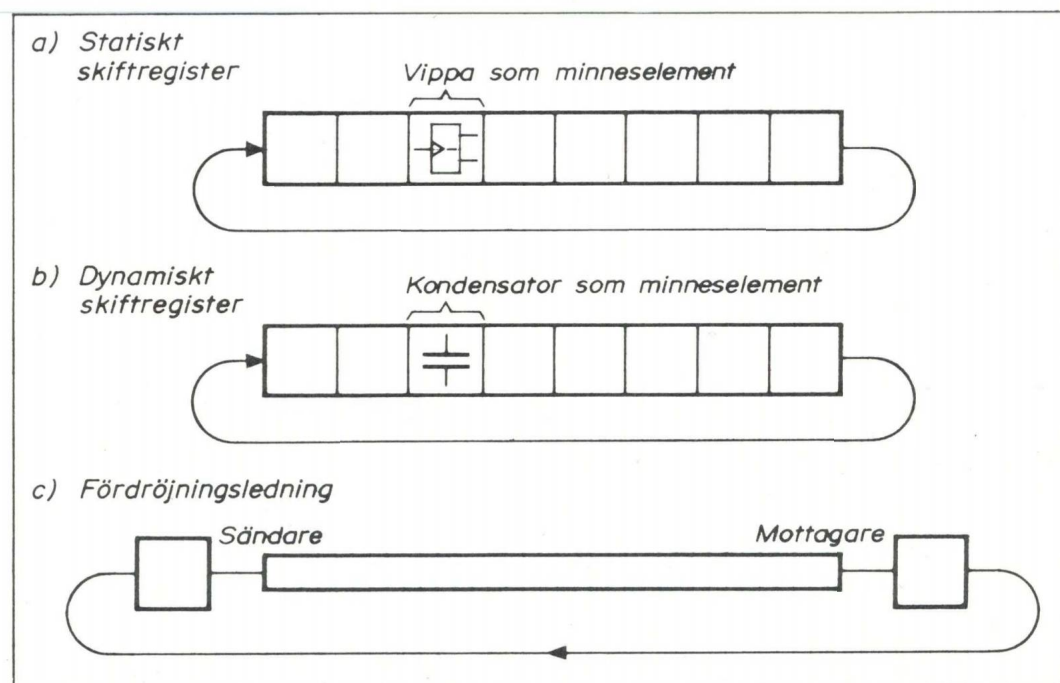
Figur 5.4
Halvledarminne till minidator med moduler om 4 k ord.

2 Serieminnen

Serieminnen (eller sekventiella minnen) kan byggas upp på en mångfald olika sätt. Serieminnen med skiftregister har vi redan tidigare varit i kontakt med och därför behandlas de endast översiktligt nedan. Två nya integrerade serieminnen med stora utvecklingsmöjligheter kallas BBD (bucket brigade device) och CCD (charge coupled device). Vi skall kort orientera oss om deras uppbyggnad. Avsnitten 2.2 - 2.4 är merläsning för speciellt intresserade.

2.1 Skiftregister och fördröjningsledningar

Figur 5.5 visar tre olika typer av serieminnen. Statiska och dynamiska skiftregister har vi redan tidigare stiftat bekantskap med, figur 2.7 och figur 4.7. Skillnaden mellan dessa är att vi fritt kan välja klockfrekvensen i ett statiskt register (det kan stå stilla obegränsad tid utan att tappa innehållet så länge matningsspänningen finns kvar) medan vi måste hålla en viss minimifrekvens hos det dynamiska registret. Det gäller ju att fylla på förlorad laddning i kondensatorerna (refresh).



Figur 5.5
Tre olika serieminnen.

Fördröjningsledningar är vanliga som serieminnen. I praktiskt taget alla färg TV-mottagare sitter ett sådant serieminne och det har till uppgift att komma ihåg en linje på bildskärmen (i PAL-system).

Fördröjningsledningen är vanligen akustisk och matas i ena änden av en liten sändare. I andra änden sitter en mottagare som förstärker signalen och återmatar den till sändarsidan. I ett fördröjningsledningsminne kan man tydligen inte variera "klockfrekvensen" utan här blir det fördröjningen som bestämmer omloppstiden.

Såväl dynamiska skiftregister som fördröjningsledningar kan i princip lagra såväl digitala som analoga värden och man talar i det senare fallet om analoga serieminnen.

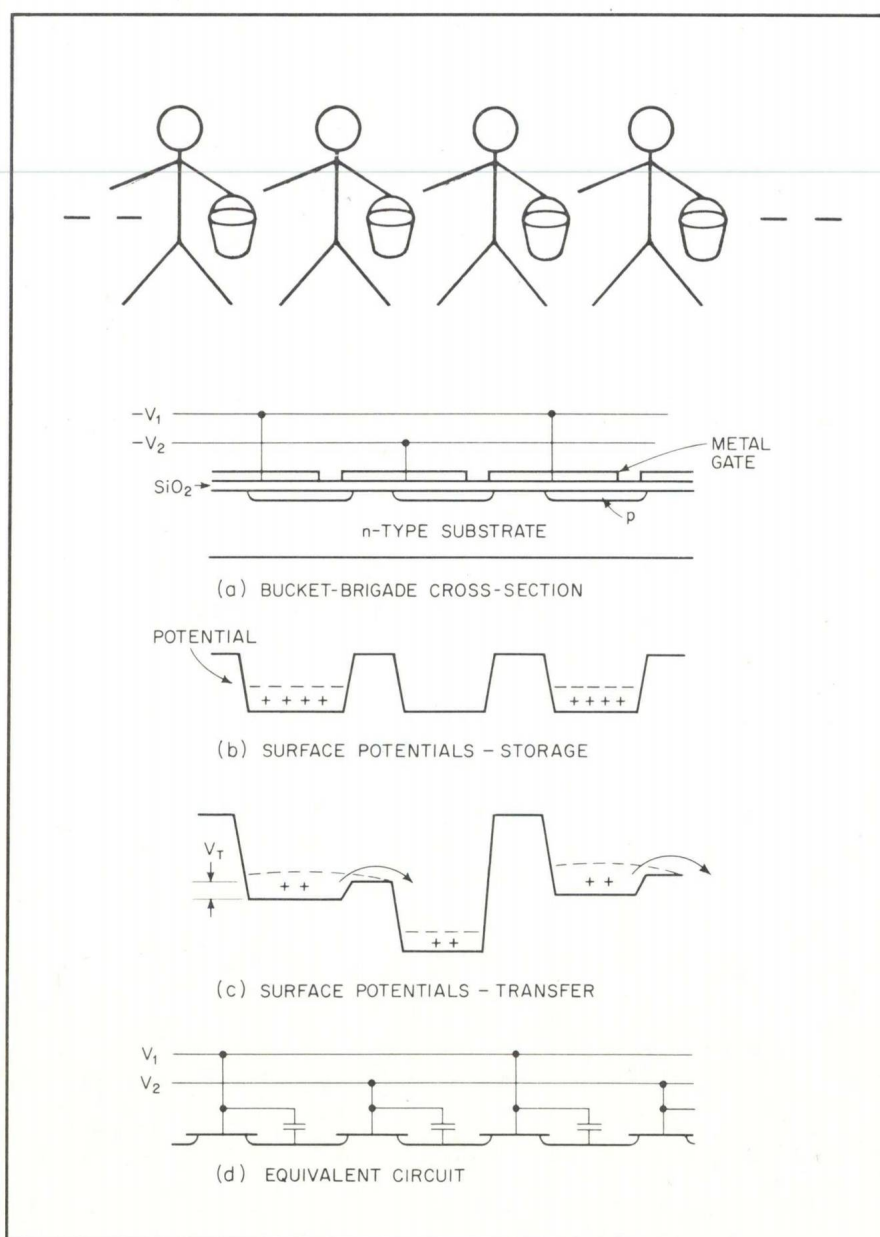
2.2 Bucket brigade

Bara veta vad
det är

En ny typ av analoga serieminnen har fått namnet "bucket brigade" dvs "hink-brigaden". Namnet härrör från den kedja som brandsoldater bildar när de langar hinkar med vatten, figur 5.6.

Vattenhinkarna motsvaras i en "bucket brigade"-kristall av små P-områden och kedjan av brandsoldater motsvaras av metallstyren (metal gate) ovanpå en mycket tunn oxid. Genom att omväxlande lägga spänning på varannan styrelektrod trycker man ner potentialbarriären mellan P-skikten, dvs mellan hinkarna, så att laddningen kan rinna över till angränsande hink.

Bucket brigade device uppfanns på Philips laboratorier år 1970. Strax därefter annonserades "laddningskopplarna" (CCD) från Bell laboratoriet i USA.

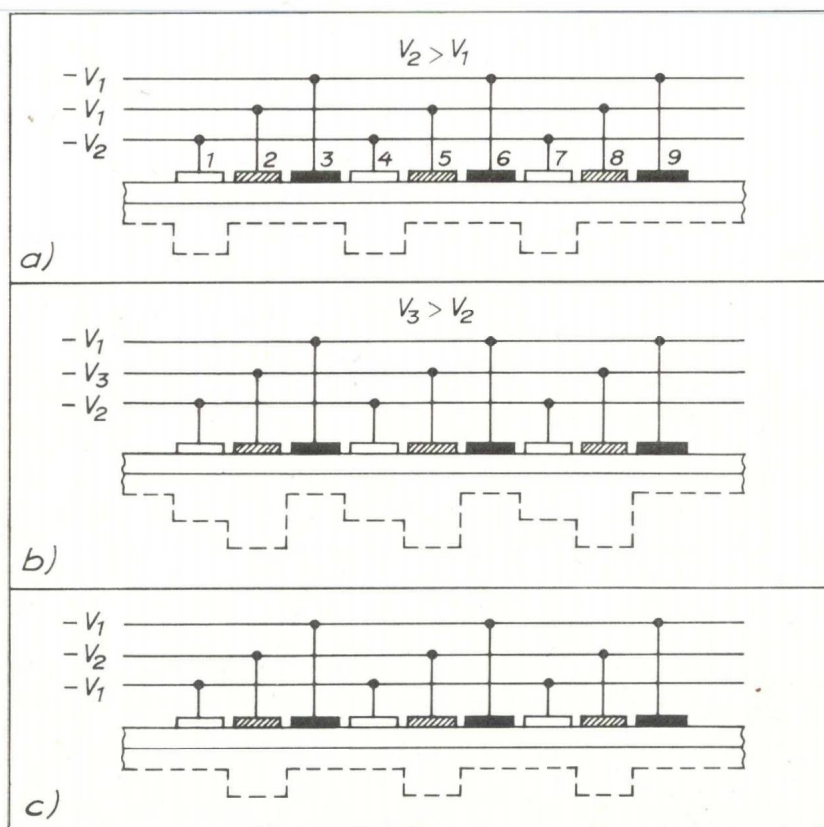


Figur 5.6
Bucket brigade device. (Electronics.)

2.3 Laddningskopplare CCD-minne

Vippor och MOS-switchar upptar båda stor plats på kristallen. De kräver dessutom många process-steg vid tillverkningen. I sökandet efter enkla re principer för laddningsförflyttning i en halvledarkristall gav BBD en elegant lösning. Man har dock kunnat gå ytterligare ett steg i förenkling av strukturen och det gjordes med CCD-elementen eller "laddningskopplarna".

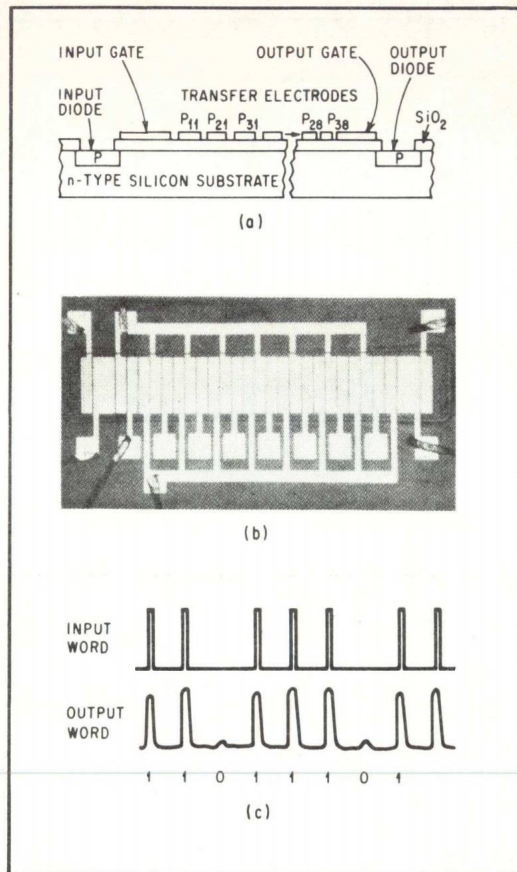
En CCD-struktur innehåller inga P-områden (hinkar) som BBD-strukturen. CCD-strukturen innehåller enbart styrelektroder som är isolerade från N-kristallen av en tunn oxid. Dessa elektroder är ordnade i ett "3-fas system" som framgår av figur 5.7. I viloläget, figur 5.7a, har elektrod ① mer negativ spänning än ② och ③ och attraherar därmed en tidigare injicerad laddning. Om nu elektrod ② ges mer negativ spänning än elektrod ① kommer ② att överta laddningen från ①, figur 5.7b. Därefter kan spänningarna på både ① och ② förskjutas i positiv riktning och då kommer ② att hålla laddningen på samma sätt som ① gjorde i utgångsläget, figur 5.7c.



Figur 5.7
Principen för laddningsförflyttning i ett CCD-element. (Electronics.)

På detta sätt kan laddningar förflyttas längs raden av elektroder och tack vare "3-fas systemet" kommer de alltid att vara åtskilda från varandra.

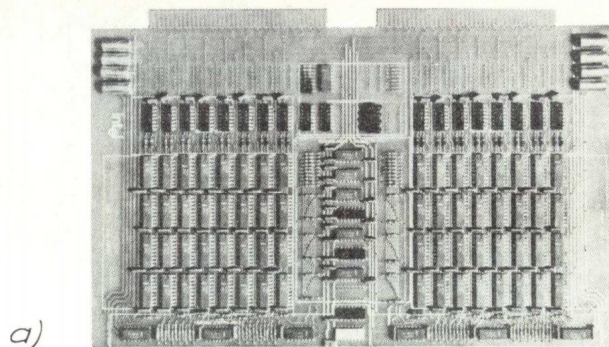
Laddningarna till ett CCD-minne kan injiceras i ena änden med hjälp av en emitterdiod och avkännas i den andra änden med hjälp av en detektor-diod. Figur 5.8 visar principen.



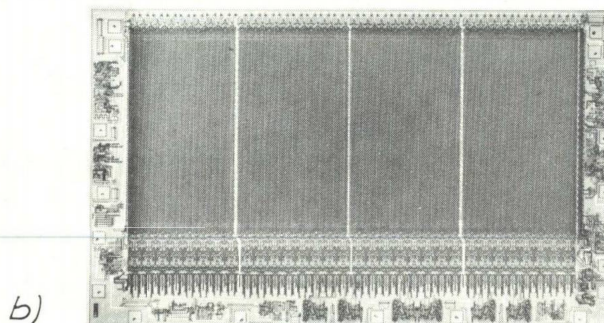
Figur 5.8
Det första CCD-elementet från Bell.

CHARGE COUPLED DEVICE 2416

Pictured below is a 1 million bit CCD Storage Card built with Intel's 2416 CCD Register. The card has an average latency time (access time to any bit) of less than 100 μ s and a maximum data transfer rate of 16 million bits per second which may be increased to 64 million bits per second by using interleaved accesses.



The photomicrograph below is of the 2416 16,384 bit-CCD Register Chip.



Figur 5.9
Intels CCD-minne typ 2416.
a) Kretskort med 64 kretsar ($2^{20} = 1048576$ bitar).
b) Kristall.

Laddningar kan även induceras på fotoelektrisk väg och stora CCD-matriser kan på detta sätt användas i kamerarör för TV. Först belyses matrisen med en bild och sedan skiftas matrisens linjer (i form av CCD-register) ut som en videosignal.

Den enkla uppbyggnaden gör att CCD-minnen får ett mycket lågt pris per bit. De kommer förmodligen att i framtiden kunna ersätta, eller komplettera, skivminnen. Figur 5.9 visar ett exempel på ett CCD-minne från 1975.

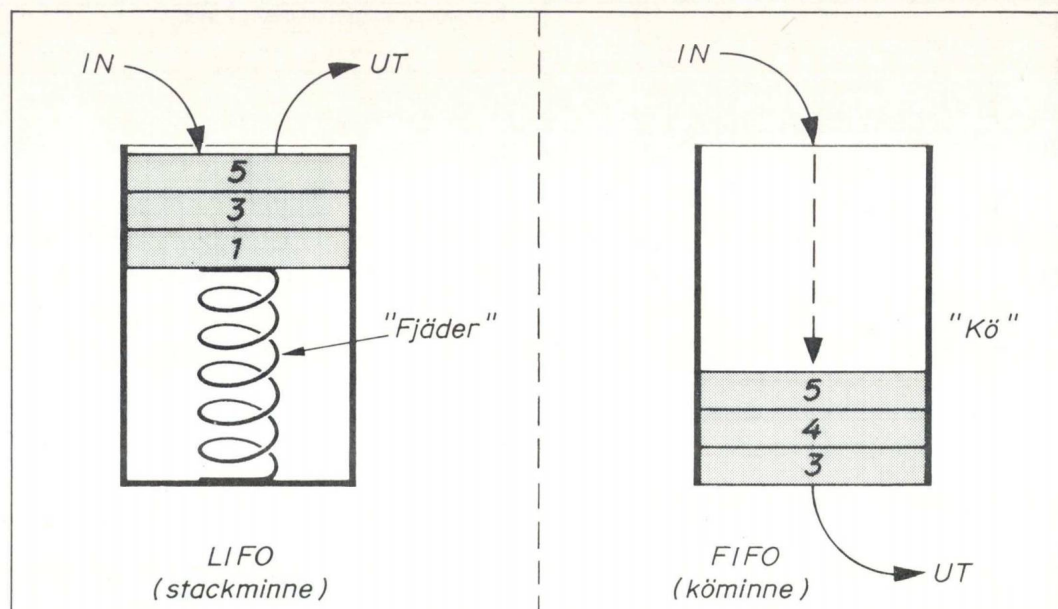
2.4 LIFO och FIFO

LIFO är en akronym eller kortform för "last in first out" och FIFO för "first in first out". En vanlig benämning för LIFO är "stackminne" eller "stack".

Stackminnen förekommer i många datorer för lagring av adresser (pekare) vid hopp eller för lagring av register vid avbrott. I många räknedosor används stackminnen, exempelvis HP-35, HP-45 etc.

FIFO-minnen används när inkommande och utgående data ej är synkroniserade och man därmed har behov av en buffert.

Figur 5.10 visar principen för ett LIFO- och ett FIFO-minne. Båda har ingångarna upptill och staplar inkommande data "på hög". Skillnaden mellan FIFO och LIFO ligger i att man i stackminnet plockar ut data upptriffrån - sist in först ut - medan man i FIFO-minnet plockar ut data nerifrån - först in först ut. FIFO-minnet fungerar alltså som en kö.



Figur 5.10
Principen för minnen av typ LIFO och FIFO.

Om ingångarna i figur 5.10 matas med orden 1 2 3 4 5 och "uttag" görs efter andra och fjärde "insättningen", blir innehållet i minnena så som framgår av figuren.

LIFO- och FIFO-minnen är elektroniska motsvarigheter till många vardagliga lagringsformer. När man tar sin bricka på en självservering, tar man den ur en stack (LIFO). När man mått och belåten sätter brickan på disktransportören, placerar man den i en kö (FIFO) i väntan på diskplockaren.

3 Läsminnen, ROM

Läsminnen (ROM = read only memory eller ROS = read only storage) har fått stor användning både för programlagring i dator- och mikrodator-system och som ersättare för kombinationskretsar uppbyggda av grindar, dvs "wired logic", "random logic" eller fast logik.

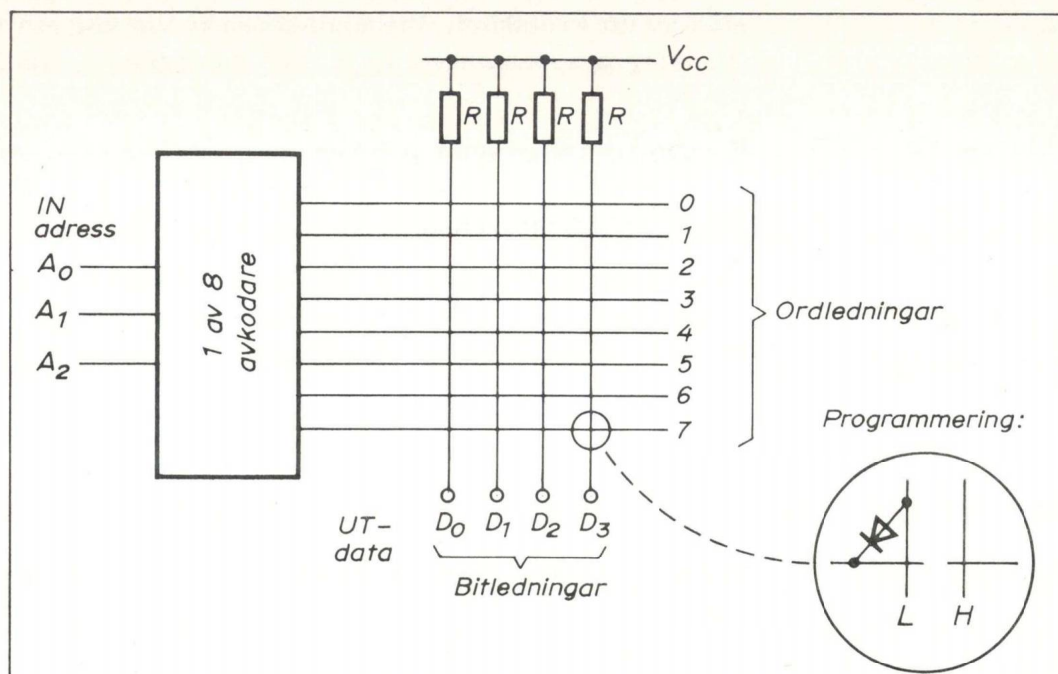
Vi skall nedan först bekanta oss med principen för ett ROM och sedan beskriva några olika typer och deras egenskaper.

3.1 Principen för ett läsminne

Vi skall i detta avsnitt beskriva uppbyggnad och funktion hos ett enkelt läsminne. Ett läsminne kan användas antingen som en kombinationskrets - (variabler in och funktioner ut) eller som ett minne för permanent lagring av data (adress in och data ut). Läsmindet kan därför beskrivas med olika typer av termer.

3.1.1 Uppbyggnad och funktion

Figur 5.11 visar ett läsminne uppbyggt av dioder. Vi har i detta exempel tre adressingångar vilket ger $2^3 = 8$ olika adresser. Till vänster sitter en adressavkodare (1 av 8) som sänker spänningen på en av de åtta utgå-



Figur 5.11

Principen för ett ROM. (Dioder i krysspunkterna ej utritade.)

ende ordledningarna. Varje ordledning har på detta sätt fått en adress, 0 - 7. Endast när denna adress är pålagd adressgångarna, $A_0 - A_2$, ligger ordledningen låg.

Ordledningarna överkorsas av ett antal bitledningar - i vårt exempel är det fyra bitledningar, $D_0 - D_3$. Ord- och bitledningar är isolerade från varandra genom att de ligger i två skilda plan, exempelvis på var sin

summasiffran / carrysiffran

a) Adress				b) Önskat innehåll				c) Diodmatris
Decimal	A_0	A_1	A_2	D_0	D_1	D_2	D_3	
0	L	L	L	L	L	H	H	
1	L	L	H	H	L	L	L	
2	L	H	L	H	L	H	H	
3	L	H	H	L	H	H	H	
4	H	L	L	H	L	L	L	
5	H	L	H	L	H	H	L	
6	H	H	L	L	H	L	L	
7	H	H	H	H	H	H	L	

Figur 5.12

Programmering av ROM:et i figur 5.11 (enligt önskelistan i mitten).

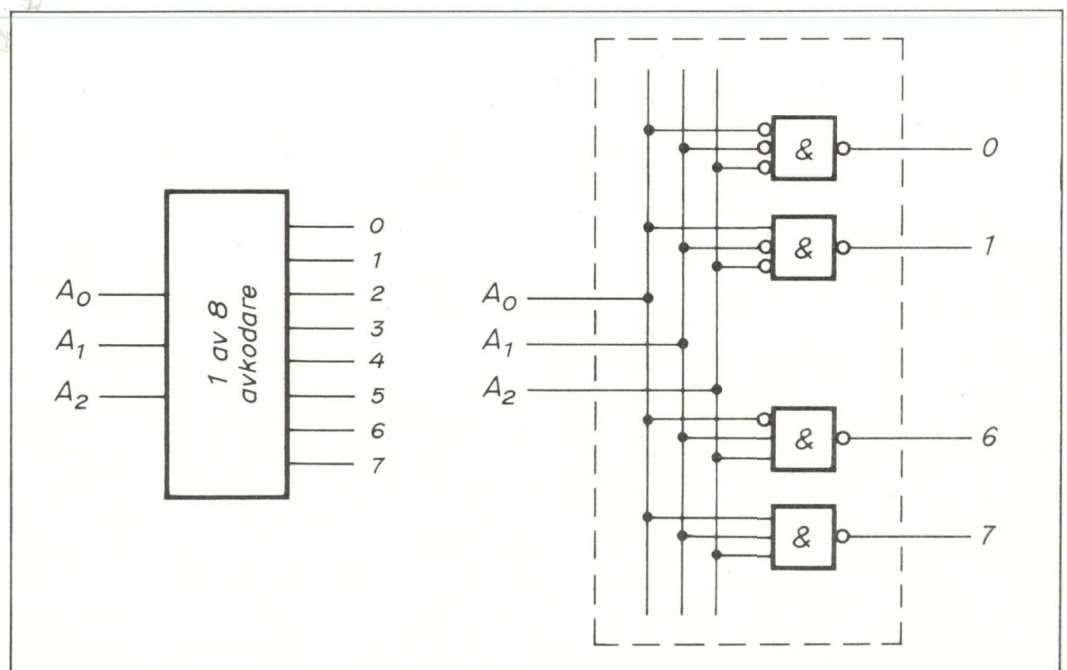
sida av ett kretskort. Bitledningarna är via var sin resistor R anslutna till matningsspänningen V_{CC} . Utspänningen på samtliga utgångar är därför hög.

Nu kommer finessen med läsminnet. Vi kan "programmera" det, dvs ge varje adress det innehåll vi själva önskar. I figur 5.12 ser vi ett exempel på programmering.

Det önskade innehållet i de åtta minnespositionerna anges i figur 5.12b. I figur 5.12c har dioder inkopplats i krysspunkterna så att man får det önskade innehållet på varje adress.

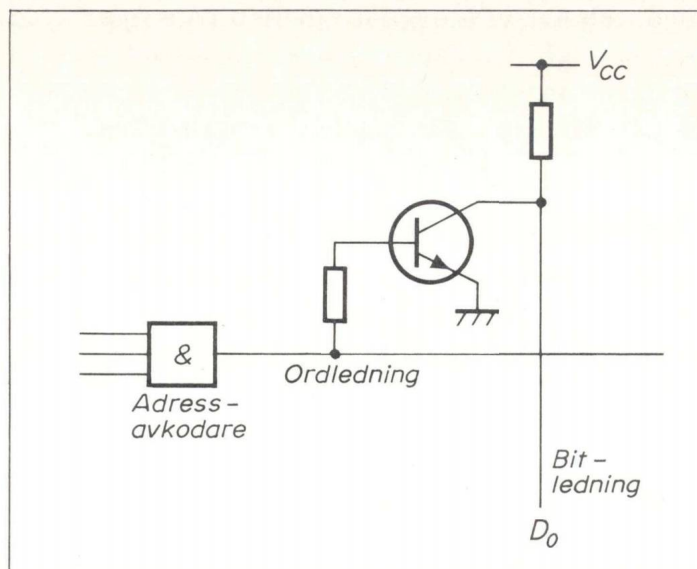
Vi har programmerat vårt ROM i figur 5.11 genom att löda in dioder i en matris. Numera används integrerade ROM där programmeringen tillgår på betydligt enklare sätt vilket vi strax skall se.

Adressavkodaren till ROM:et i figur 5.11 kan bestå av ett antal NAND-grindar. Figur 5.13 visar principen och funktionen har vi beskrivit tidigare i kapitel D 1.



Figur 5.13
Adressavkodaren i figur 5.11

I större ROM kommer utgångarna från adressavkodarna, figur 5.13, att bli belastade med många bitledningar. För att minska belastningen använder man transistorer i stället för dioder i krysspunkterna. Figur 5.14 visar ett exempel. En adresserad ordledning läggs i detta fall hög (AND-grind i stället för NAND-grind) och därvid sänker transistorer spänningen på vissa av utgångarna.



Figur 5.14
ROM med transistorer i krysspunkterna (princip).

3.1.2 Minne eller kombinationskrets

Som vi tidigare nämnt är, enligt IEC, en kombinationskrets "en digital krets för vilken det existerar en och endast en kombination av utsignaler för varje möjlig kombination av insignaler".

Kombinationskretsen matas på ingången med ett antal binära variabler och varje utgång representerar en logisk (eller Boolesk) funktion av dessa variabler. NAND-funktionen är ett exempel på en sådan funktion.

I figur 5.15a har vi återgivit en tabell för fyra olika funktioner (S, C₀, Z och Y) av de tre ingångsvariablerna (C_i, A och B). Två av funktionerna känner vi igen från figur 1.19, det är heladderarens summa (S) och carry ut (C₀). Varje funktion motsvarar en kolumn i sanningstabellen.

heladderare

Detta minne fungerar som en kombinationskrets.

a)

Variabler in			Funktioner ut			
C_i	A	B	S	C_0	Z	Y
0	0	0	0	0	1	1
0	0	1	1	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	1	1
1	0	0	1	0	0	0
1	0	1	0	1	1	0
1	1	0	0	1	0	0
1	1	1	1	1	1	0

b)

Adress in			Data ut			
A_0	A_1	A_2	D_0	D_1	D_2	D_4
0	0	0	0	0	1	1
0	0	1	1	0	0	0
0	1	0	1	0	1	1
0	1	1	0	1	1	1
1	0	0	1	0	0	0
1	0	1	0	1	1	0
1	1	0	0	1	0	0
1	1	1	1	1	1	0

Figur 5.15

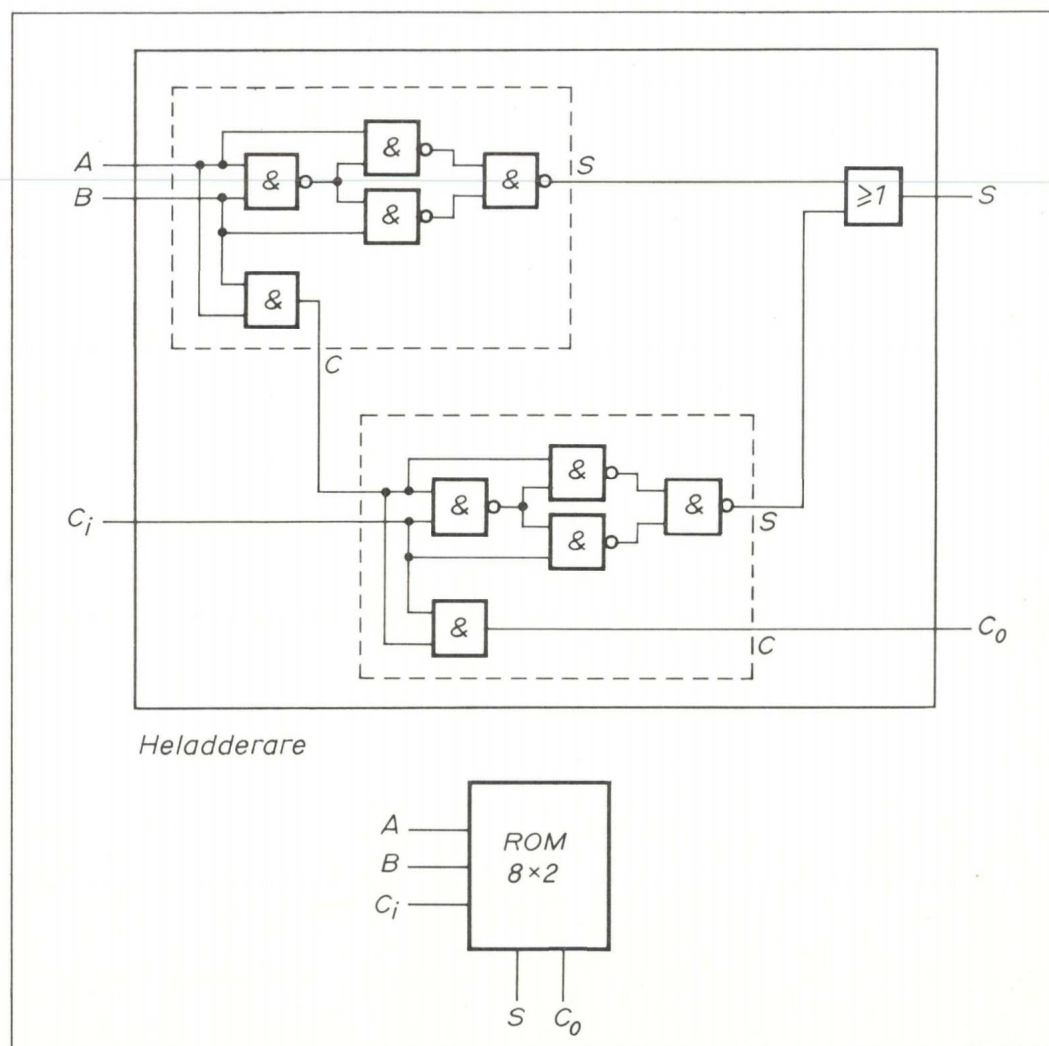
Kombinationskrets eller läsminne?

a) Variabler in ger funktioner ut. b) Adresser in ger data ut.

I figur 5.15b har vi återgivit tabellen från figur 5.12. Varje adress (A_0 , A_1 , A_2) på ingången öppnar motsvarande datafack i minnet så att vi kan avläsa dess innehåll (D_0 , D_1 , D_2 , D_3) på utgången. Vi talar alltså här om adress och data, dvs innehåll i minnesfack.

Som vi ser är tabellerna i figur 5.15 a och b identiskt lika så när som på beteckningarna i tabellhuvudena.

Vi har tidigare sett hur man kan realisera en heladderare med två halvadderare och en OR-grind, figur 1.19. Varje halvadderare kan i sin tur realiseras av en EXOR-krets och en AND-grind, figur 1.18. EXOR-kretsen kan realiseras med fyra NAND-grindar, figur 1.14. En heladderare kan tydligen byggas upp med enkla grindar på det sätt som framgår av figur 5.16a. Det åtgår totalt elva grindar och detta är ett exempel på "wired logic" eller fast logik.



Figur 5.16
En heladderare uppbyggd med grindar ("wired logic") och med ROM ("programmed logic").

Heladderarens funktion kan även programmeras in i ett ROM. Det är just vad vi gjorde i figur 5.12, som även innehåller ytterligare två funktioner. Det åtgår för detta ändamål $8 \times 2 = 16$ bitar av ROM-minnet. Vi har på detta sätt programmerat ett ROM att fullgöra en önskad funktion.

Man jämför ofta "wired logic" och "programmed logic" och i större system motsvaras ofta en grind i "wired logic" av 10 - 20 bitar av ROM. Då läsminnen nu blivit både billiga och lättprogrammerade spar man ofta både tid, pengar och arbete på att programmera läsminnen i stället för att med stor möda minimera och löda ihop stora grindnät. Man slipper därvid även avsevärda problem med testning av kretsar och "debugging" av felsignaler på grund av olika fördröjningstider etc.

Ett läsminne eller ROM är alltså en kombinationskrets som kan innehålla både speciella och komplexa funktioner men som trots detta är uppbyggd och massproducerad som en standardkrets. Vi skall nu se hur användaren programmerar sina ROM. Man gör det numera på betydligt enklare sätt än att löda in dioder i en diodmatris.

3.2 Programmering av ROM

Det finns två huvudtyper av läsminnen:

- maskprogrammerade (mask programmed) ROM
- PROM (programmable ROM), dvs programmerbara ROM

Den första typen programmeras av fabrikanten som sista steg i tillverkningsprocessen. PROM:et kan programmeras av apparatkonstruktören själv.

Det förekommer två typer av PROM:

- fPROM (field programmable ROM) med elektrisk programmering (fusible links)
- EPROM (erasable field programmable ROM) eller på svenska: rader-PROM.

Ett ROM är alltid statiskt eftersom det har permanent inprogrammerade data, som alltså inte laddas ur med tiden.

3.2.1 Maskprogrammerade ROM

Det sista steget både vid MOS- och bipolarprocessen är som vi sett tidigare, figur 1.30, påläggning av ledarmönstret ovanpå kiseloxiden. Vid tillverkningen av ett ROM kan man härvid använda en mask, dvs ett mönster, som enbart ansluter vissa transistorer och därmed ger det önskade innehållet i varje ord.

Apparattillverkaren sänder då in önskat program på hålkort eller hålremsa till ROM-fabrikanten. Med ledning av det insända programmet gör ROM-fabrikanten en specialmask för ledningsmönstret. Programmering enligt denna metod tar lång tid och blir ekonomisk först vid stora serier med lika program.

Maskprogrammerade ROM tillverkas både i MOS- och bipolar teknik. I MOS kan ROM:en göras större (16 kbit) och billigare men blir å andra sidan långsammare än bipolära ROM.

3.2.2 PROM

För att minska "turn-around-tiden", dvs tiden från beställning till leverans, vid programmeringen kan man använda ett PROM, dvs ett ROM som kan programmeras av användaren själv.

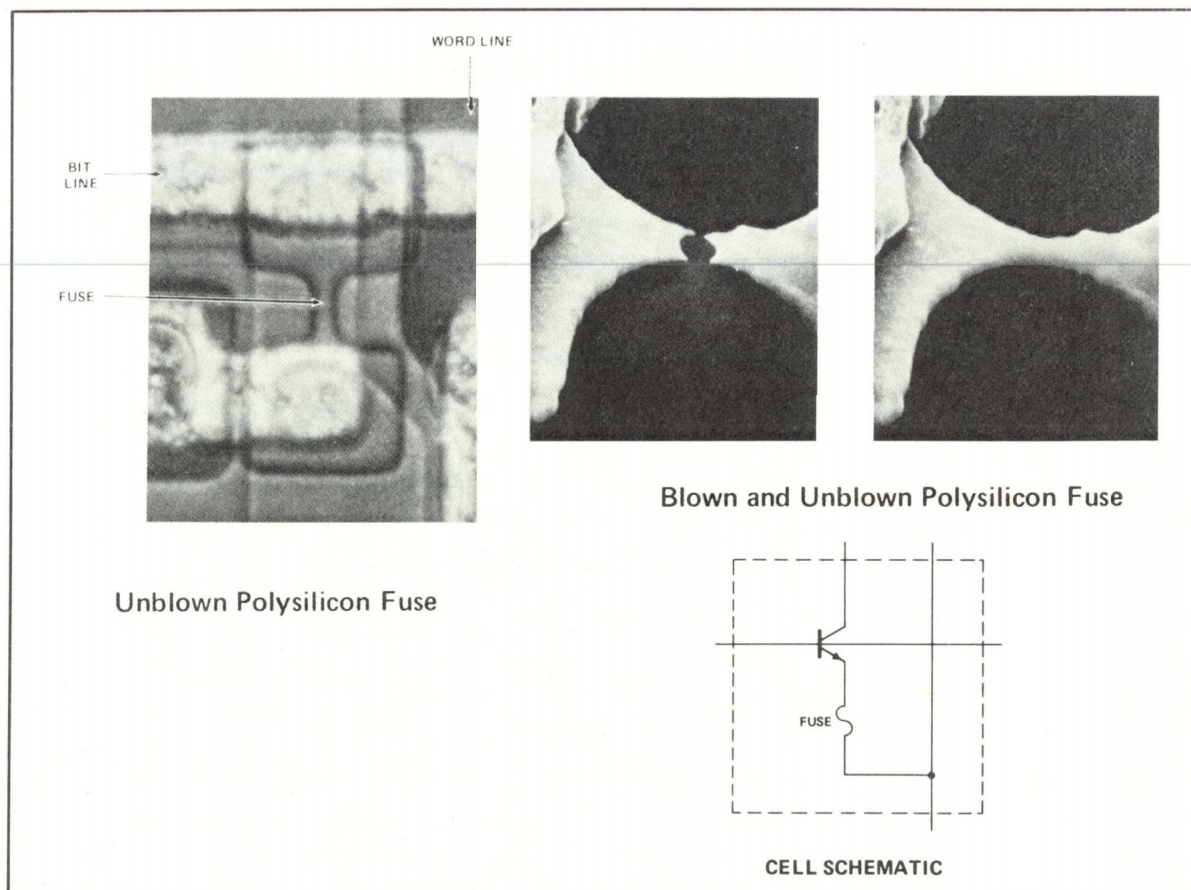
3.2.2.1 fPROM

Man har lyckats tillverka transistorer med en smal till-ledning i form av en "säkring" (fusible link).

Om man kör en stark strömspik genom transistorn smälter säkringen så att transistorn blir bortkopplad. På detta sätt kan användaren själv programmera sitt ROM (eller rättare PROM). Det är vanligen bipolära PROM som programmeras på detta sätt.

En nackdel med PROM är givetvis det förhållandet att om man programmerar in fel eller vill ändra innehållet, får man kassera det tidigare PROM:et och programmera på nytt.

Figur 5.17 visar ett exempel på fPROM, dvs field programmable ROM.



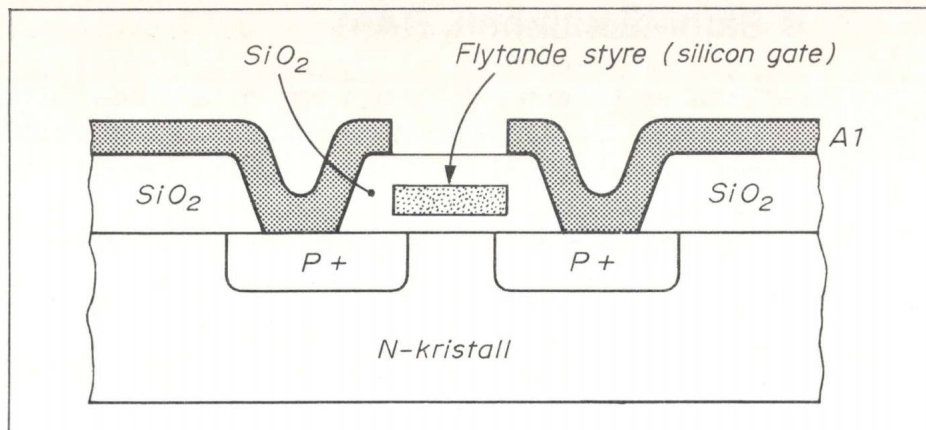
Figur 5.17

"Säkring" (fuse) av kisel i bipolärt fPROM (Intel). = FPLA *array*

3.2.2.2 Rader-PROM

I MOS-tekniken används ofta styren av polykristallint ledande kisel (silicon gate). Fördelen med dessa är bl a att de kan läggas med stor precision över strömkanalerna (self alignment) vilket leder till snabba transistorer med låg kapacitans.

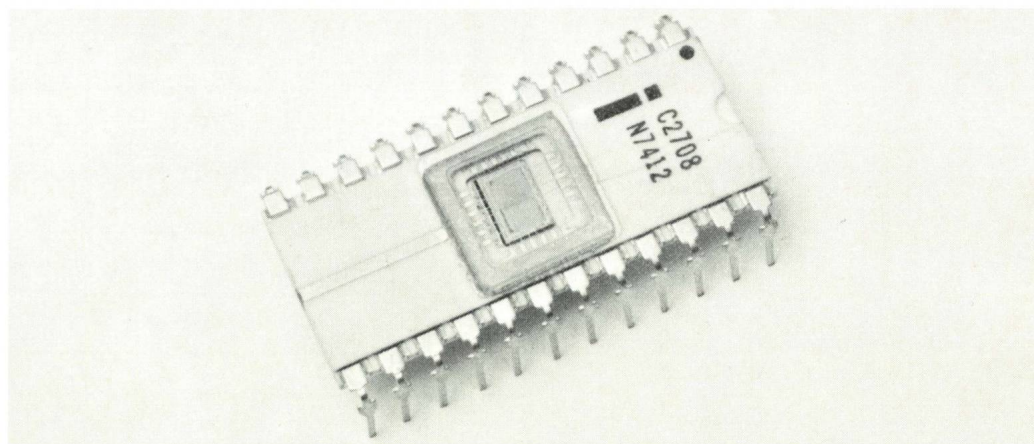
Intel har utvecklat en typ av läsminnen i MOS-teknik som har frisvävande silicon gate-styren. Om man ger dessa transistorer en spänningsspik, storlek 50 V, kommer det flytande styret att laddas genom lavineffekt. Eftersom styret är helt omgivet av välisolerande kiseldioxid kvarstår denna laddning i styret (mer än 10 år vid 125 °C). Minnet kan alltså programmeras permanent med spänningsspikar som, tack vare spännings-tåliga avkodare, kan adresseras till avsedd plats i minnet. Figur 5.18.



Figur 5.18
Rader-PROM med flytande styre.

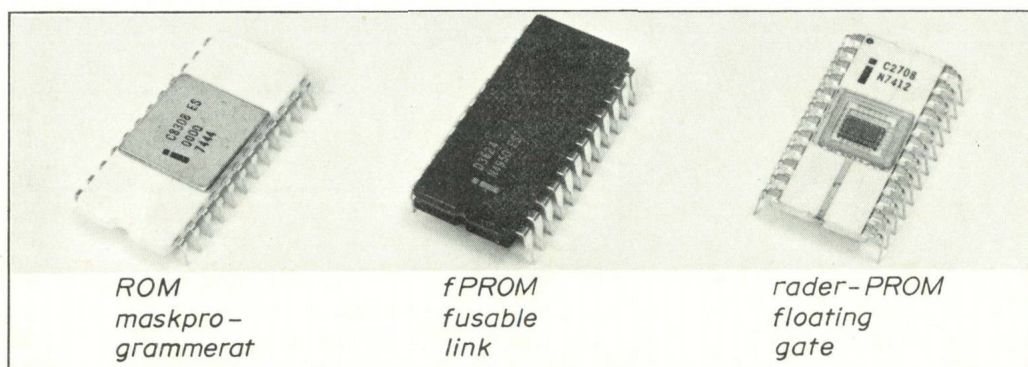
Rader-PROM är försedda med glasfönster av kvartsglas. Om kristallen belyses med röntgen eller starkt ultraviolett ljus, uppstår en fotoström som urladdar styrena. Rader-PROM kan därigenom raderas.

Rader-PROM är dyrare än "fusable-link" PROM, men kan å andra sidan omprogrammeras vid behov. Figur 5.19 visar ett exempel.



Figur 5.19
Rader-PROM (Intel).

I figur 5.20 sammanfattar vi de viktigaste typerna av läsminnen.

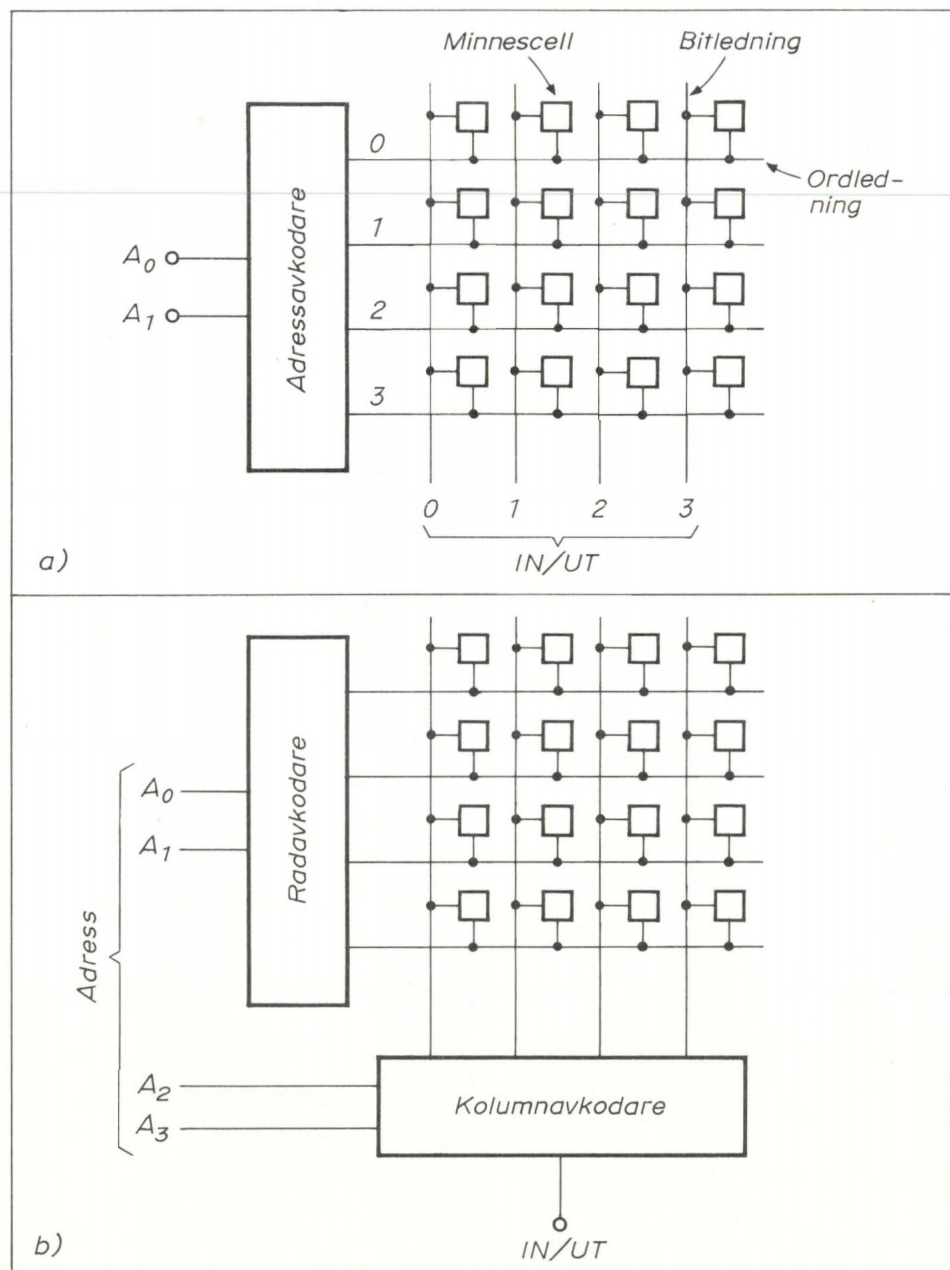


Figur 5.20
Exempel på ROM, fPROM och raderPROM (Intel).

4 Skriv-läsminnen, RAM

Ett skriv-läsminne av halvledartyp kallas vanligen RAM (random access memory). Det är uppbyggt på samma sätt som ett ROM, figur 5.14, men med en väsentlig skillnad. I krysspunkterna på ett RAM sitter minneselement som kan kontrolleras utifrån. Dessa minneselement kan vara statiska (vippor) eller dynamiska (kondensatorer).

Figur 5.21a visar principen för ett RAM. De fyrkantiga blocken är minneselement (eller minnesceller). Läggs en adress på adressingången kommer en av ordledningarna att ändra spänning. Därmed kan det adresserade ordets bitar antingen avläsas via bitledningarna, eller också ställas in i önskade lägen via dessa bitledningar.



Figur 5.21

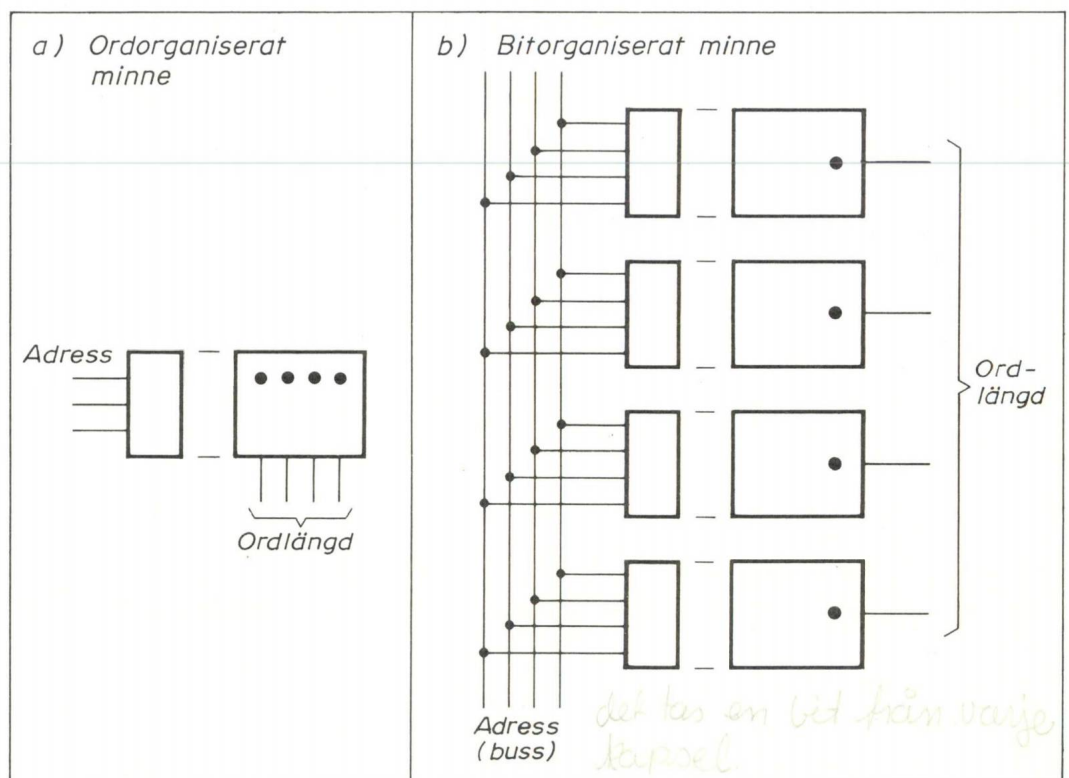
Skriv/läsminne i princip. a) Ordorganiserat. b) Bitorganiserat.

en enda bit fås ut
dessa är det vanligaste
minnet

Ett minne där man som i figur 5.21a adresserar ett ord (innehållande flera bitar) brukar kallas "ordorganiserat".

Om minnesmatrisen innehåller många celler är en kvadratisk uppbyggnad mest ekonomisk ur ledningssynpunkt. För att ändå begränsa antalet bitar som samtidigt avläses brukar man använda en kolumnavkodare (multiplexer). Figur 5.21b ger ett exempel där man använder ytterligare två adressledningar och därigenom kan avkoda varje kolumn för sig. Vi får data (in/ut) i ord om en bit. Ett sådant minne kallas ofta "bitorganiserat".

I figur 5.22 har vi ritat ett ord- och ett bitorganiserat minnessystem. I det ordorganiserade tar vi hela ordet från samma krets men i det bitorganiserade hämtar vi en bit från varje minneskrets. Terminologin "ordorganiserad" och "bitorganiserad" härstammar från kärnminnestekniken där bitorganiserade minnen utformades som ett antal ovanpå varandra liggande minnesmatriser.



Figur 5.22
Ord- och bitorganiserad minnesuppbyggnad.

Vad som ovan sagts om minnens organisation gäller såväl för ROM som för RAM. Vi skall nu bekanta oss med några olika typer av minnesceller i halvledarminnen av RAM-typ.

4.1 Statiska RAM

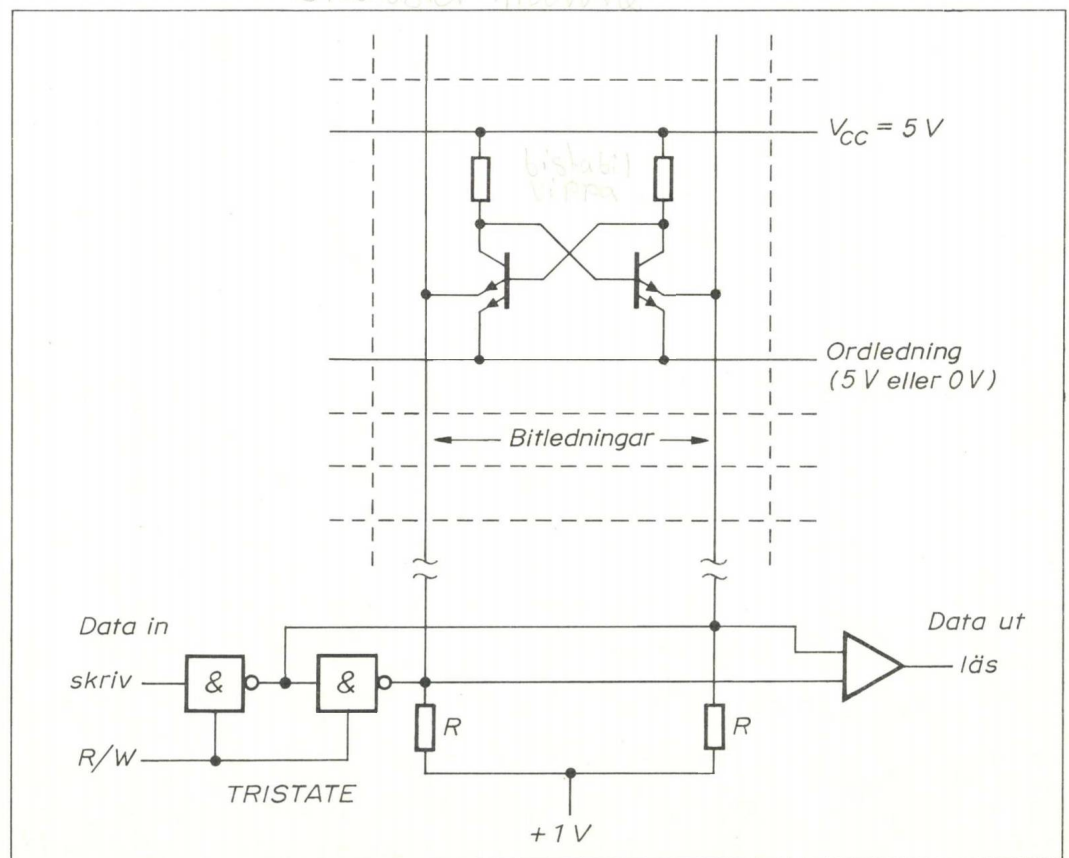
Statiska minnen har minnesceller i form av bistabila vippor. Önskar man ett snabbt minne bör man välja ett bipolärt RAM där celler utgörs av TTL- eller ECL-vippor. Vill man ha ett billigare men långsammare RAM kan man välja ett statiskt MOS-minne, numera vanligen i NMOS. Har man krav på extremt låg effektförbrukning bör man välja ett CMOS-RAM.

4.1.1 Bipolärt RAM

En minnescell i ett bipolärt minne utgörs av en enkel vippa. Bipolära minnen används när man har behov av kort accesstid och minnescellerna byggs därför upp av snabba logikfamiljer, exempelvis ECL eller TTLS. Man uppnår därvid accesstider av storleken 30–60 ns. Figur 5.23 visar en typisk minnescell i TTL.

Inkopplingen av cellen sker med hjälp av de två emitterarna på varje transistor. Bitledningarna hålls därvid på en potential av ca 1 V. Vid läsning läggs ordledningen hög (5 V) och då kommer de båda emitterarna till bitledningarna att leda. Strömmen från den transistor som leder kommer då att ge ett litet spenningsfall över ett av motstånden R i bitledningarna. En differentialförstärkare avkänner spenningskillnaden och ger utsignal (data ut).

Vill man skriva in, dvs tvinga vippan att inta en viss ställning, kan man koppla in en krets som ett kort ögonblick höjer spennningen på den bitledning vars transistor man önskar strypa. I figur 5.23 är detta utfört med två grindar som endast är inkopplade vid inskrivning (tristate).

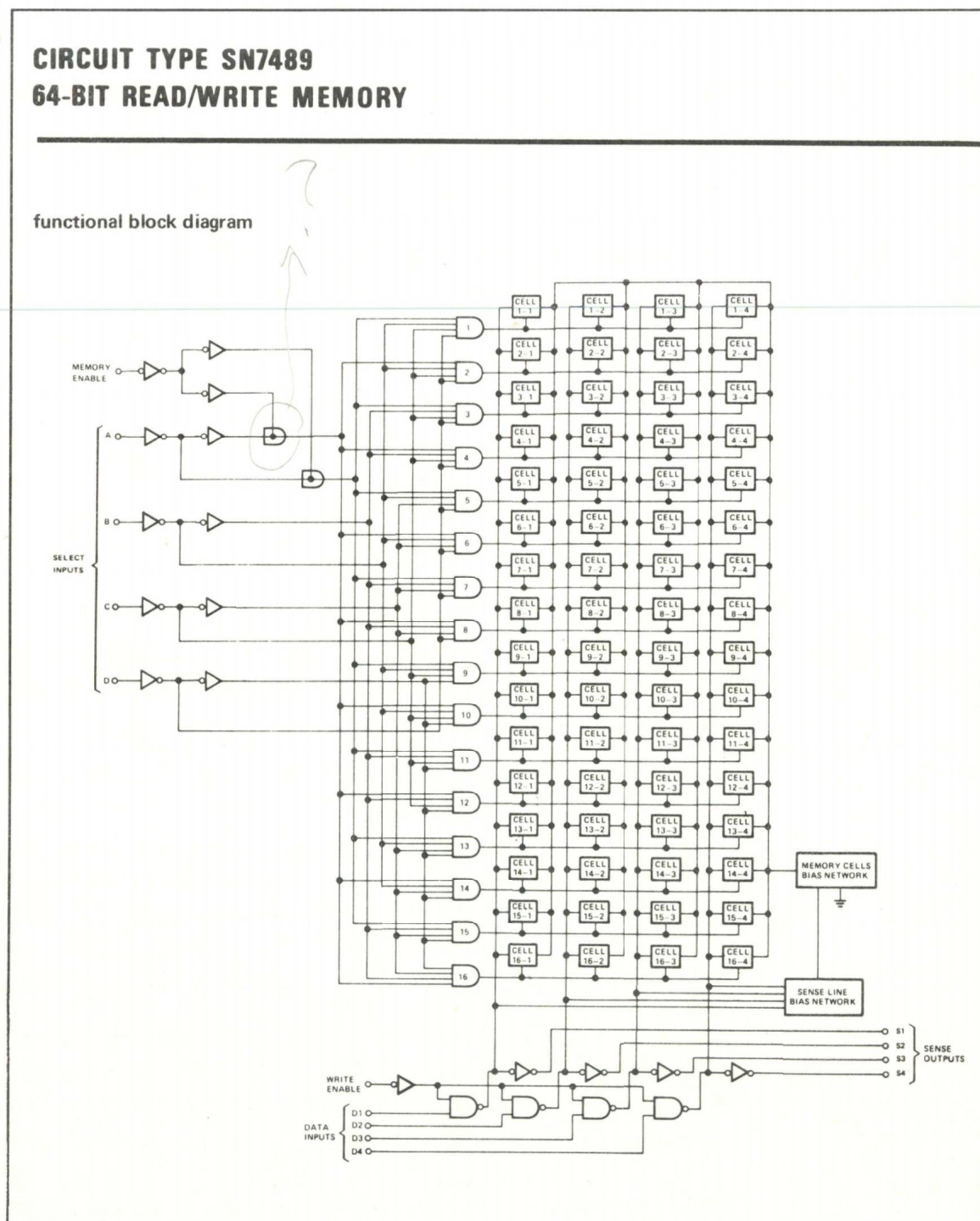


Figur 5.23
Bipolär minnescell (TTL).

Vi har ovan sett hur cellen fungerar när den är inkopplad till bitledningarna, dvs när den är adresserad. Om ordledningen hålls låg (0 V) kommer emitterarna till bitledningarna att strypas och cellen ligger isolerad från omgivningen.

Adressering av cellen i figur 5.23 kan alltså enkelt göras genom att man varierar spänningen på ordledningen. Det finns en mängd varianter av bipolära minnesceller, men principen för deras funktion är i stort sett densamma som i figur 5.23.

Figur 5.24 visar uppbyggnaden av minnet 7484 (Texas Instruments) så som det presenteras i databladet. Minnescellens uppbyggnad specificeras ej men schemat ger en tillräcklig "funktionsbeskrivning" för den som skall använda kretsen.

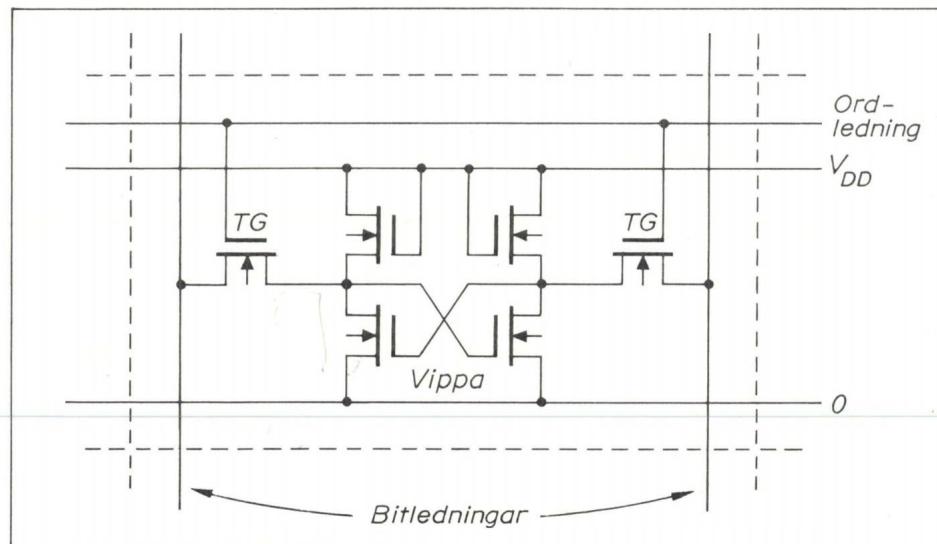


Figur 5.24
Uppbyggnad av ett 16 x 4 bitars bipolärt RAM (7484, Texas Instruments).

4.1.2 Statiskt MOS-RAM

Med MOS-teknik kan man som vi sett få plats med betydligt fler minnesceller på en kristall av given storlek än vad som är möjligt i bipolär teknik. Man kan göra större minnen (räknat i bit) med lägre effektkonsumtion i MOS-teknik.

Figur 5.25 visar en statisk minnescell i MOS-teknik.



Figur 5.25
Statisk minnescell i NMOS.

Bland de mest använda minnena i svensk industri 1975 är det statiska NMOS-minnet 2102. Figur 5.26a visar dess datablad. Det är organiserat i 1024 ord om 1 bit (jfr figur 5.21b) och innehåller en minnesmatris med 32x32 celler.

För radavkodningen krävs 5 adressledningar $A_0 - A_4$ eftersom $2^5 = 32$. För val av kolumn krävs ytterligare fem adressledningar.

Minnet har tristate-utgång (tristate är inregistrerat av National Semiconductor och Intel kallar det därför three-state) och kan "frikopplas" via ingången "chip select" eller CS som den vanligen kallas. Som framgår av figur 5.25 får minnet plats i en 16-bens DIL-kapsel och kan därigenom produceras till lågt pris. Priset rör sig om ca 20 kr (1975) och kan så småningom förväntas sjunka till hälften, varför bit-priset närmar sig 1 öre/bit. Figur 5.26 b och c visar kapseln och cellens kristallstruktur. In- och utgångar är TTL-kompatibla vilket avsevärt förenklar uppbyggnaden av "interface" till omgivande elektronik.

När ett minnessystem byggs upp används CS-ingången (chip select) för val av minneskrets. Figur 5.27 visar principen för uppbyggnaden av ett minne med 4 k ord om vardera 8 bitar. Samtliga kretsar är anslutna till en gemensam adressbuss om 10 bitar och med denna väljer man alltså ut en cell i varje krets.

1024 BIT FULLY DECODED STATIC MOS RANDOM ACCESS MEMORY

- Single +5 Volts Supply Voltage
- Directly TTL Compatible — All Inputs and Output
- Static MOS — No Clocks or Refreshing Required
- Low Power — Typically 150 mW
- Access Time — Typically 500 nsec
- Three-State Output — OR-Tie Capability
- Simple Memory Expansion — Chip Enable Input
- Fully Decoded — On Chip Address Decode
- Inputs Protected — All Inputs Have Protection Against Static Charge
- Low Cost Packaging — 16 Pin Plastic Dual-In-Line Configuration

The Intel[®] 2102 is a 1024 word by one bit static random access memory element using normally off N-channel MOS devices integrated on a monolithic array. It uses fully DC stable (static) circuitry and therefore requires no clocks or refreshing to operate. The data is read out nondestructively and has the same polarity as the input data.

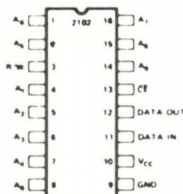
The 2102 is designed for memory applications where high performance, low cost, large bit storage, and simple interfacing are important design objectives.

It is directly TTL compatible in all respects: inputs, output, and a single +5 volt supply. A separate chip enable (CE) lead allows easy selection of an individual package when outputs are OR-tied.

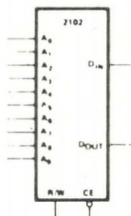
The Intel 2102 is fabricated with N-channel silicon gate technology. This technology allows the design and production of high performance easy to use MOS circuits and provides a higher functional density on a monolithic chip than either conventional MOS technology or P-channel silicon gate technology.

Intel's silicon gate technology also provides excellent protection against contamination. This permits the use of low cost silicone packaging.

PIN CONFIGURATION



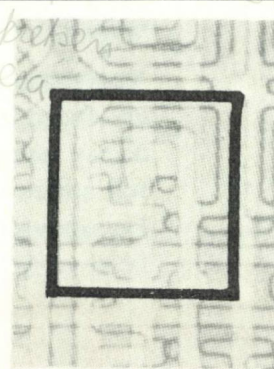
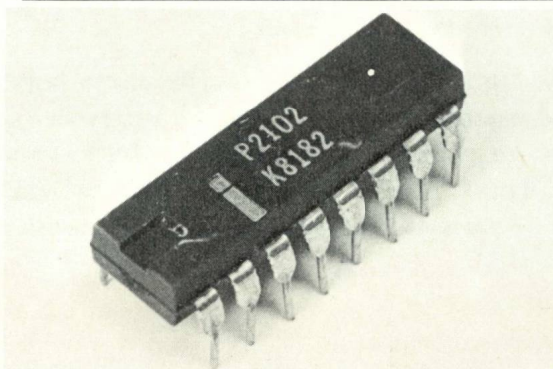
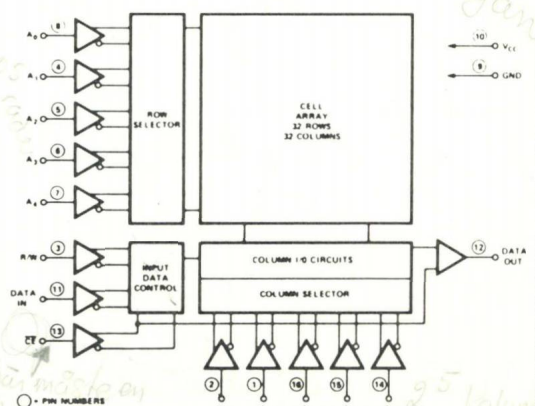
LOGIC SYMBOL



PIN NAMES

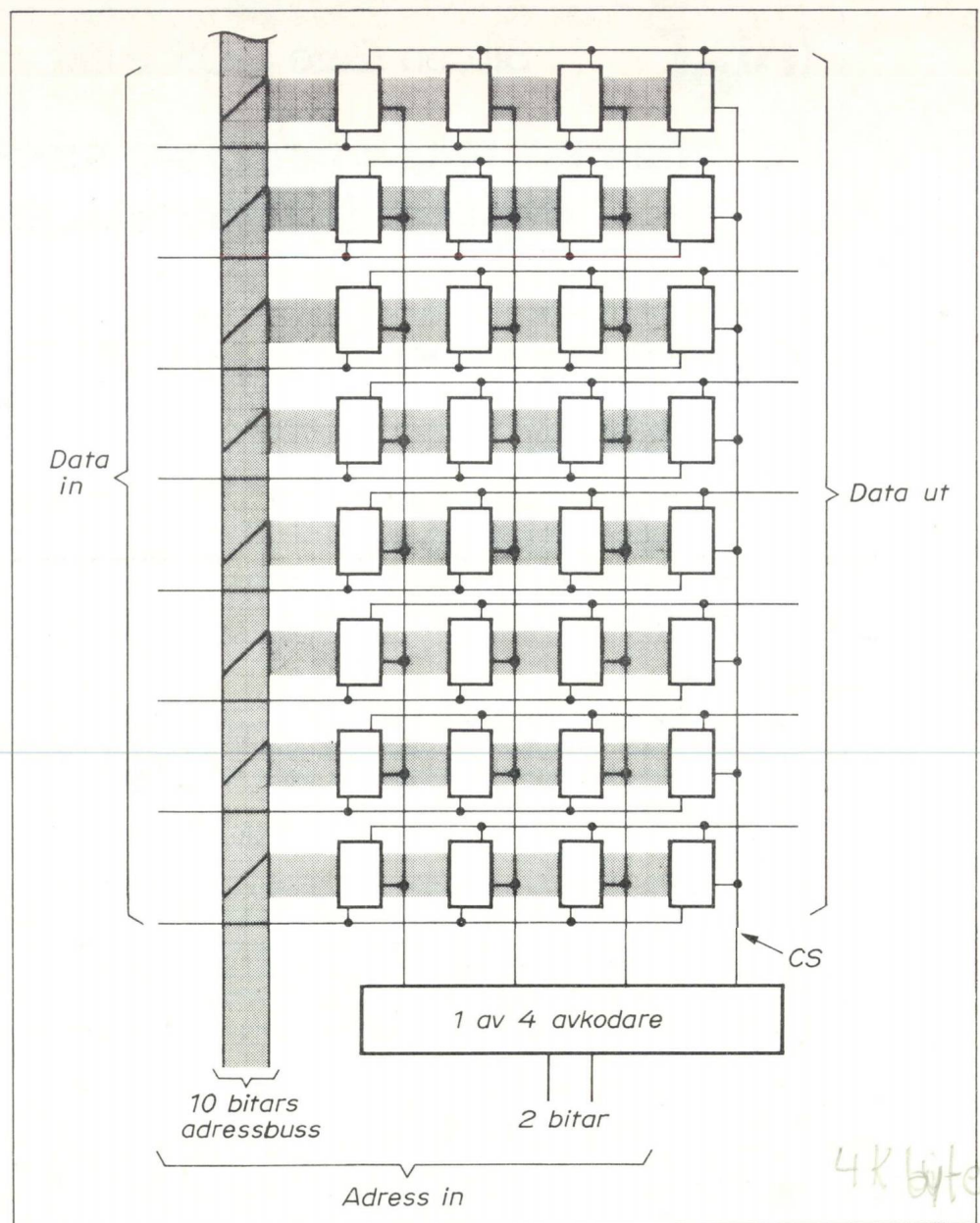
D _{IN}	DATA INPUT	CE	CHIP ENABLE
A ₀ - A ₉	ADDRESS INPUTS	D _{OUT}	DATA OUTPUT
R/W	READ/WRITE INPUT	V _{CC}	POWER (+5V)

BLOCK DIAGRAM



Figur 5.26
2102 - arbets-
håsten bland
minnen.

- a) Datablad.
- b) Kapsel.
- c) Minnescell
på kristallen.



Figur 5.27
Organisation av minnessystem.

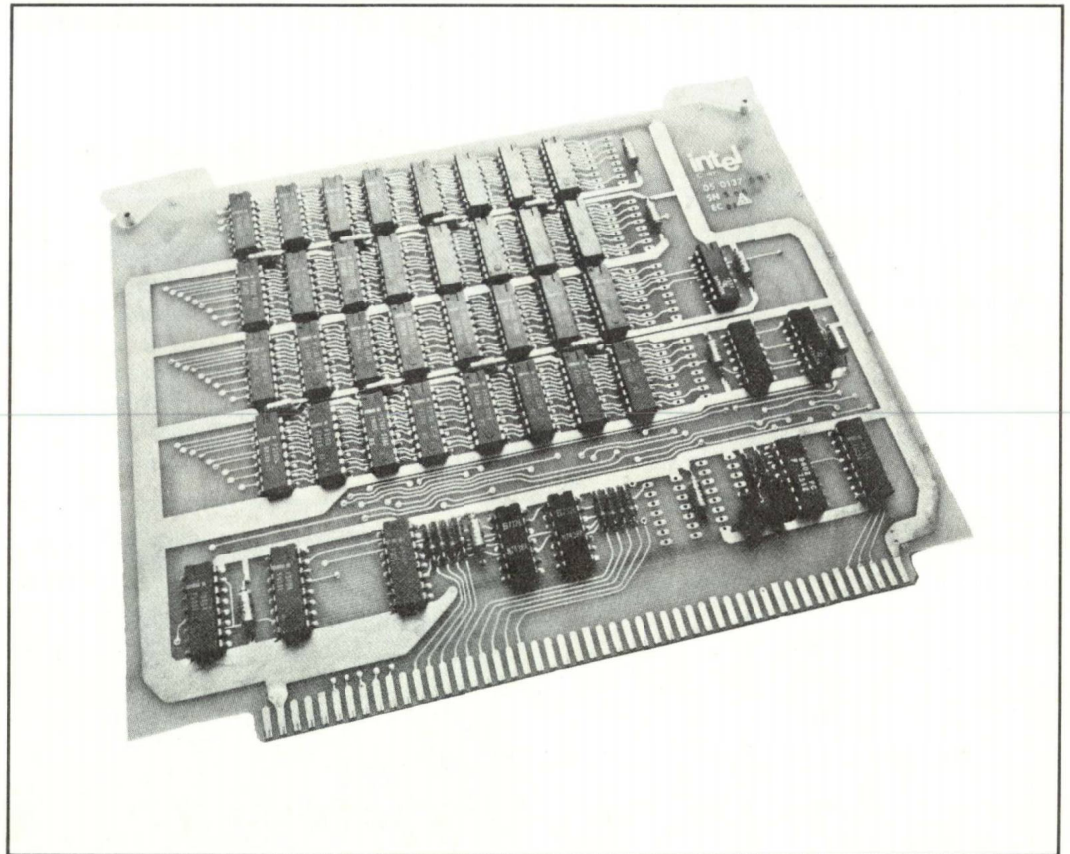
Systemet är organiserat med ett antal rader som motsvarar den önskade ordlängden, 8-bitars ordlängd i figur 5.27.

Kapaciteten bestäms sedan av antalet kolumner. I vårt exempel (4 k) krävs tydligen fyra kretsar - dvs fyra kolumner.

Kretsarna i varje kolumn är anslutna till en för varje kolumn gemensam CS-ledning. En avkodare (en av fyra) väljer ut en av dessa ledningar, därav namnet chip select, och för detta åtgår i vårt exempel ytterligare två adressledningar. Totalt behövs tydligen 12 adressledningar ($2^{12} = 4\text{ k}$) för ordvalet. Internt är minnet "bitorganiserat" eftersom man tar en bit från varje kapsel.

Uttagens placering är så anordnade att det skall vara enkelt att göra "lay out" för kretskortet. Vid uppbyggnaden av minnessystem kommer många adressgångar att ligga parallellt och belasta adressbussen. Man får då se upp så att drivkretsarna till adressbussen tål belastningen och eventuellt använda speciella drivkretsar (bus drivers).

Även minnessystem säljs i dag som standardkomponenter. Figur 5.28 visar ett exempel.



Figur 5.28
Minneskort med 2102. (Intel.)

4.1.3 CMOS-minnen

En CMOS-vippa drar ingen effekt i viloläget. Med CMOS kan man tydligt bygga upp statiska RAM-minnen med mycket liten effektförbrukning. CMOS är därför en teknologi som kan väntas bli av stor betydelse för uppbyggnad av minnen.

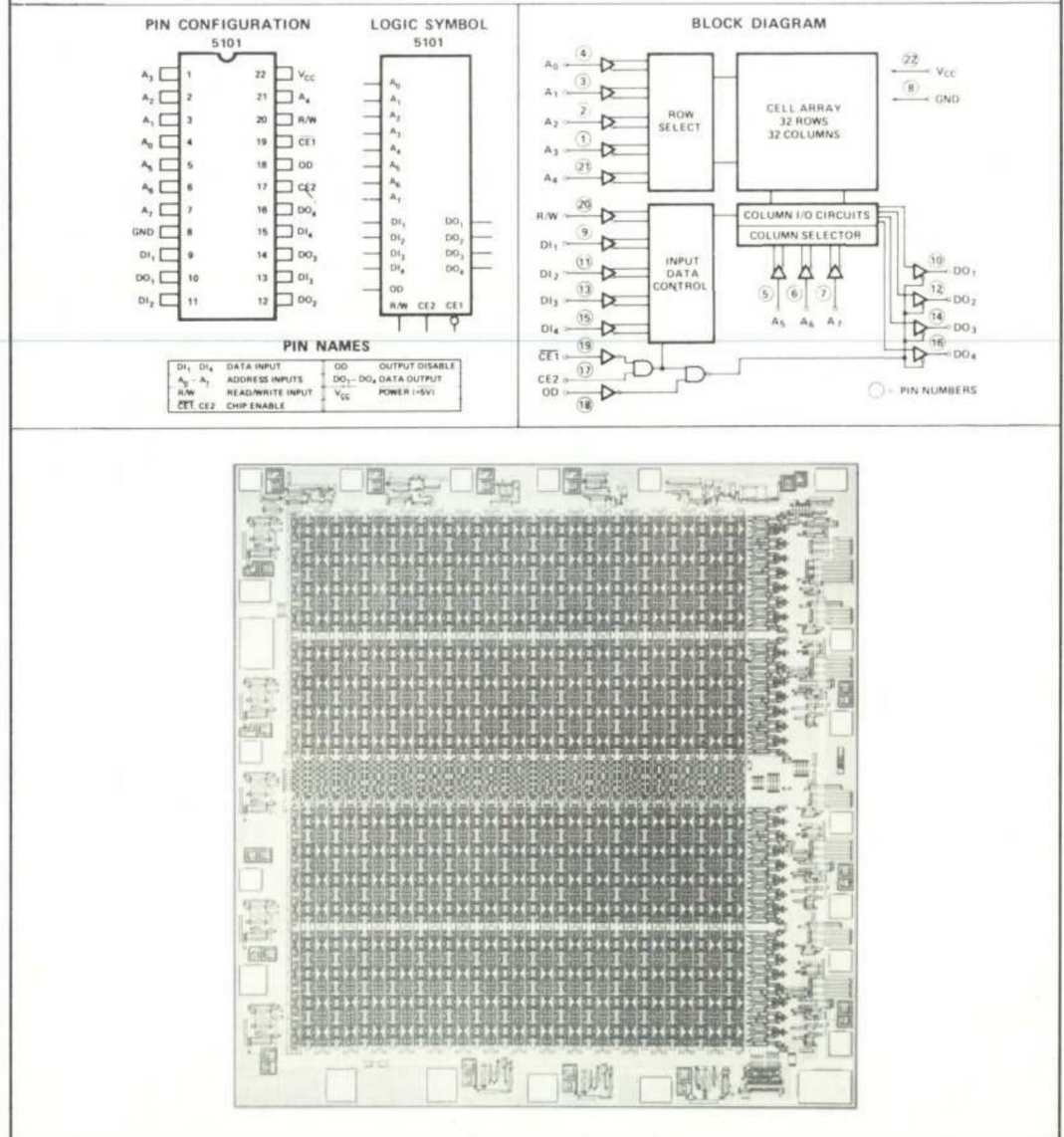
CMOS kräver fortfarande (1975) relativt stor kristallarea och många processteg vid tillverkningen jämfört med NMOS. Minnena blir därför dyrare.

Figur 5.29 visar ett CMOS-minne om 1024 bitar (256×4). Det har TTL-kompatibla in- och utgångar. I vila (stand by) drar minnet totalt strömmen $0,2 \mu\text{A}$ (typiskt värde vid $V_{CC} = 5 \text{ V}$) dvs förlusteffekten $1 \mu\text{W}$. NMOS-minnet 2102 drar som jämförelse 150 mW .

1024 BIT (256 x 4) STATIC CMOS RAM

*Ultra Low Standby Current: 15 nA/Bit for the 5101

- Fast Access Time—650 ns
- Single +5 V Power Supply
- Directly TTL Compatible—All Inputs and Outputs
- Three-State Output



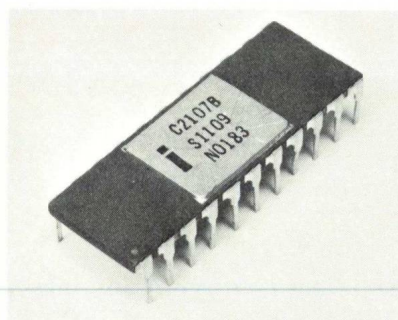
Figur 5.29
CMOS-minnet 5101. (Intel.)

4.2 Dynamiska RAM

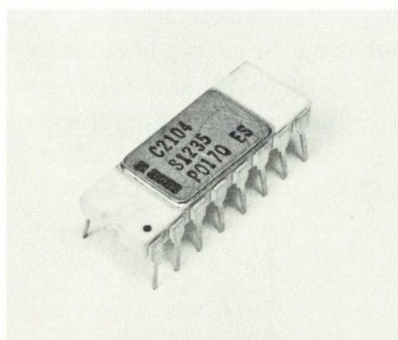
Behovet av minnen med stor kapacitet och lågt pris ökar ständigt. För att nedbringa kostnaden gäller det att få in så många bitar som möjligt i en och samma kristall och i en och samma kapsel. Men med många bitar ökar effektkonsumtionen och därvid kylproblemen. Låg effekt och litet utrymme på kristallen är därför de primära kraven när man vill göra minnen med stor kapacitet.

halva vippan från figur 5.21 och lagt den som en för hela kolumnen gemensam refresh-förstärkare. Eftersom en kolumn innehåller många celler sparar man härigenom plats på kristallen.

I figur 5.33 ser vi uppbyggnaden av 1103 (i princip). Längst ner i varje kolumn sitter refresh-förstärkarna som samtidigt tjänar som utgångar vid avläsningen. Vid inläsning påtvingas "write"-ledningen en yttre spänning av en skrivkrets som "överröstar" läsförstärkarens utsignal. Kolumnerna avkodas på vanligt sätt och 1103 är utifrån sett organiserat i 1024 ord om 1 bit.



22 ben



16 ben



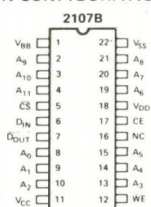
Silicon Gate MOS 2107B

FULLY DECODED RANDOM ACCESS 4096 BIT DYNAMIC MEMORY

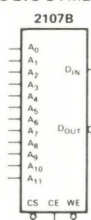
- * Access Time -- 200 ns max.
- * Read, Write Cycle Times--400 ns max.
- * Refresh Period -- 2 ms

- Low Cost Per Bit
- Low Standby Power
- Easy System Interface
- Only One High Voltage Input Signal -- Chip Enable
- TTL Compatible -- All Address, Data, Write Enable, Chip Select Inputs
- Read-Modify-Write Cycle Time--520 ns
- Address Registers Incorporated on the Chip
- Simple Memory Expansion -- Chip Select Input Lead
- Fully Decoded -- On Chip Address Decode
- Output is Three State and TTL Compatible
- Industry Standard 22-Pin Configuration

PIN CONFIGURATION



LOGIC SYMBOL

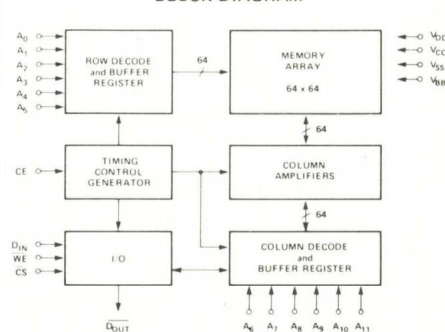


PIN NAMES

A ₀ -A ₁₁	ADDRESS INPUTS*	V _{BB}	POWER (-5V)
CE	CHIP ENABLE	V _{CC}	POWER (+5V)
CS	CHIP SELECT	V _{DD}	POWER (+12V)
D _{IN}	DATA INPUT	V _{SS}	GROUND
D _{OUT}	DATA OUTPUT	WE	WRITE ENABLE
NC	NOT CONNECTED		

*Refresh Address A₀-A₅.

BLOCK DIAGRAM



Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied.
© Intel Corporation 1975

JULY 1975

Figur 5.34

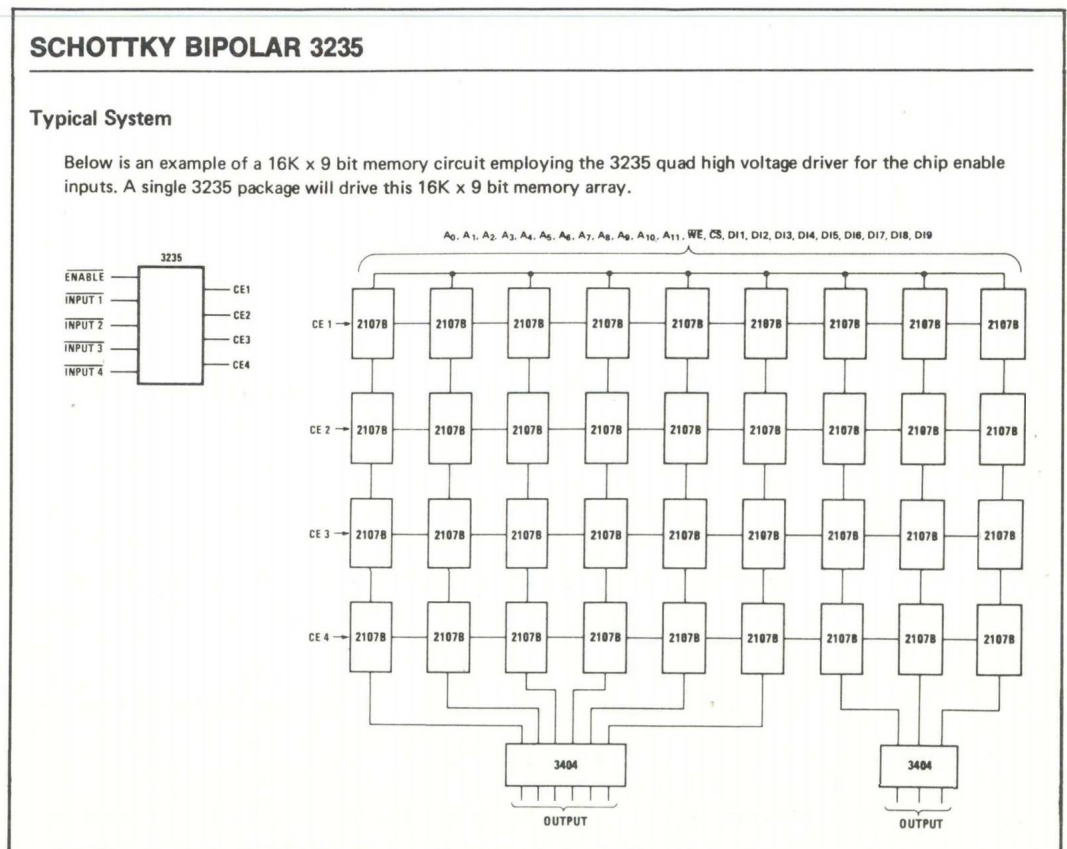
Dynamiska 4 kbit-minnen, kapslar och del ur databladet för Intel's 2107.

4.2.3 Minnessystem

Dynamiska minnen kräver refresh-kretsar och är därmed mer komplicerade att bygga upp än statiska minnessystem. Eftersom dynamiska minnen blir mer ekonomiska för system med stor kapacitet kommer det att dröja innan de ersätts av statiska minnen.

Figur 5.34a visar de två dynamiska 4k-minnen som under senare hälften av 1970-talet kommer att ersätta "arbetshästen" 1103. Ett 4k-minne kräver 12 adressgångar och vill man få in ett sådant minne i en 16-bens kapsel måste adressgångarna multiplexas. Minnet 2104 är ett exempel på en sådan krets. En billigare krets köps alltså här till priset av mer komplexa adresseringskretsar. Med en dyrare kapsel, med 22 ben i stället för 16, får man en enklare adressering. Minnet 2107 är exempel på ett minne med samtliga adressgångar direkt tillgängliga. Figur 5.34b visar en del av databladet för 2107.

Vid uppbyggnad av minnessystem kan man använda speciella driv- och läskretsar som interface till omgivande TTL-logik. Figur 5.35 visar exempel på ett minnesystem uppbyggt av 4k-minnen (2107), drivkretsen 3235 (TTL till MOS) och läskretsen 3404, som innehåller sex latchar och därmed lagrar data ut.



Figur 5.35

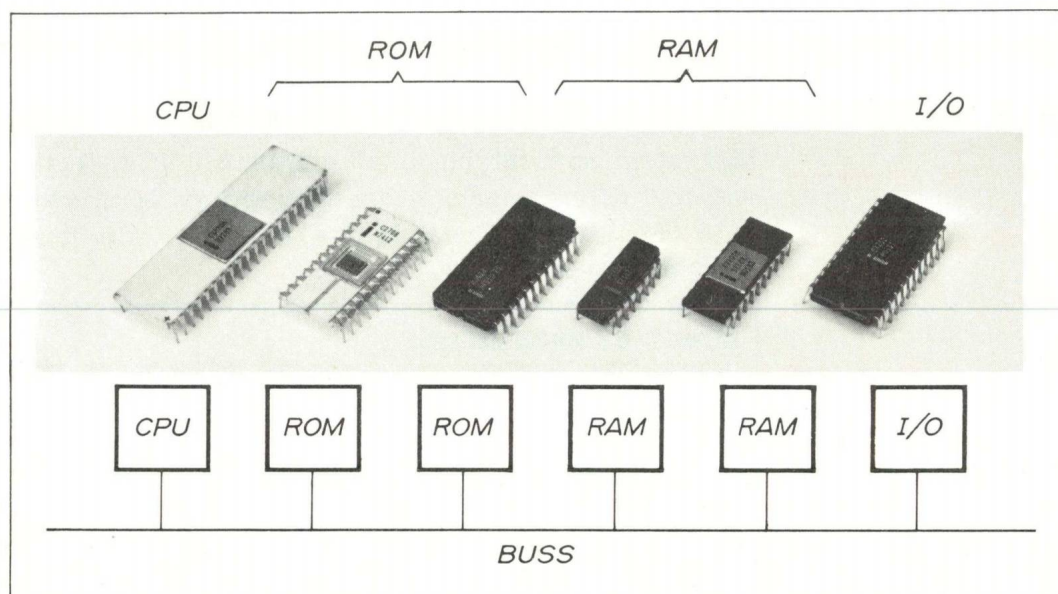
Minnessystem, 16 kord om 9 bitar, uppbyggt med 4k-dynamiska NMOS-kretsar (2107). (Intel.)

5 Mikrodatorsystem

Som avslutning skall vi kort omnämna vart en stor del av alla halvledarminnen tar vägen i dagens teknik. Många halvledarfabrikanter tillverkar mikroprocessorer i form av LSI-kretsar. De fungerar som CPU:n (CPU = central processing unit) i minidatorer men är begränsade till kortare ordlängd och enklare uppbyggnad.

Tack vare mikroprocessorernas låga pris kan man nu datorstyra en rad olika apparater som tidigare av ekonomiska skäl ej kunnat utföras med så "s sofistikerad" styrning. Exempel är trafikljus, kassaterminaler, övervakningssystem, robotar, datainsamlingssystem, mätinstrument, leksaker etc. Det är i dessa apparater halvledarminnena nu kommer in i allt större antal. (Det säljs ca 10 gånger så många minnen som mikroprocessorer till sådana system och det är minnena som utgör den största försäljningen för halvledarfabrikanterna.)

Mikrodatorsystemen byggs ofta upp kring en central buss enligt figur 5.36. Bussen innehåller adressledningar, dataledningar och kontrollerledningar.



Figur 5.36

Exempel på kretsar till ett mikrodatorsystem (Intel).

Mikroprocessorn hämtar order (program) lagrade i ROM, utför beräkningar och lagrar mellan- och slutresultat (eller mätvärden) i RAM. In-data och utdata hämtar mikroprocessorn från speciella IO-kretsar (in/ut-enheter) som är anslutna till de styrda apparaterna (exempelvis en robot).

I framtiden kommer alltså elektronikingenjörerna att få ägna en stor del av sin tid med att programmera mikrodatorer, organisera halvledarminnen och konstruera "interface" till anslutna apparater. "Wired logic", dvs att bygga elektroniksystem med hjälp av lödkolv och enkla grindkretsar, kommer att ersättas av "programmed logic", där man programmerar mikroprocessorer att utföra önskade rutiner.

En stor fördel med "programmed logic"-systemen är flexibiliteten. Vill man ändra funktionen hos apparaten sätter man in ett nytt program dvs ett nytt ROM eller ett omprogrammerat PROM. I "wired logic" måste hela systemet byggas om.

För närvarande pågår en lavinartad utveckling på mikrodatorområdet. Övergången från arbete med "hardware" till "software" kräver nytänkande och utbildning. För att underlätta konstruktionsarbetet med "programmed logic"-system har man utvecklat en mängd hjälpmedel (software support) där datorer används vid programmering av mikrodatorsystem.

Repetitionsfrågor

1. Hur sker adresseringen av orden i ett serieminne?
2. Hur många binära siffror rymmer ett minne organiserat i 4 k ord med 16 bitars ordlängd?
3. Vad ligger bakom begreppen "wired logic" och "programmed logic"?
4. Vad har dioderna för uppgift i ett ROM uppbyggt med diodmatris?
5. Vad är skillnaden mellan ett fPROM och ett rader-PROM?
6. "Ranka" följande minnen med hänsyn till effektförlust: bipolärt, CMOS, statiskt MOS, dynamiskt MOS.
7. "Ranka" följande minnen med hänsyn till snabbhet: PMOS, NMOS, CMOS, bipolärt.
8. Vad betyder CS i samband med halvledarminnen och vad används den till?
9. Vad menas med refresh när det gäller halvledarminnen?
10. Vilka kategorier av ledningar innehåller en minnesbuss?

Svenskt sakregister

- A 71
absorptionssatsen 4
accesstid 117
AC-kopplad 39
active pull-up 84
adress 116
adressavkodare 18, 118, 126
adressbuss 16
adressregister 16, 18
aktiv hög 6
analoga serieminnen 120
and 3
antibottningsdiod 78
AOI-krets 12
associativa lagen 4
asynkron 36, 41
avkoda 15
avkodare 15, 31
avkopplingskondensator 66
- B 71
basdiffusion 23
BBD 120
BCD-kod 32
belastbarhet 59, 72
ben 9
binär addition 13
binärkod 31
binärräknare 39
bipolarprocess 22
bipolärt minne 134
bistabil vippa 30
bitledning 118
bitnummer 31
bitorganiserat 133
bitvikt 31
Boolesk algebra 3
bricka 21
buffrade grindar 104
buss 16, 146
byte 116
- C 14
carry 14
CCD 120
cell 116
chip select 136
CK 37
clear 42
CMOS 97
CMOS-minne 139
CMOS-switch 97
CMOS-vippor 105
CPU 18, 145
CS 136
datafördelare 17
dataingång 37, 38
dataregister 18
dataväljare 17
DC-triggade vippor 39
de Morgans teorem 4, 5
demultiplexer 17
differentialsteg 26
diffusion 21
digital krets 2
DIL-kapslar 9
direktverkande 36, 41
distributiva lagen 4
D-latch 37
drain 90
drivkrets 83, 139
DTL 7, 59
DTL-familjen 7
DTLZ 25
D-vippa 38
dynamisk 94, 132
dynamisk förlusteffekt 112
dynamiskt MOS-minne 94
- ECL 26
effektkonsumtion 92
eller 3
emitter 90
emitterdiffusion 23
enable 52
epitaxi 20
EPROM 129
exclusive OR 12
EXOR-kretsen 12
"ett"
ettställa 41
- FA 14
"falsk" 2
familj 7
fast logik 128
FET 90
FIFO 123
flanktriggad vippa 39
flanktrigging 41
fotolitografi 21
fPROM 129
frekvensdelare 38
frekvensräknare 46
frikopplingsingång 85
frisignal 52, 86
funktionstabell 6, 7
fördröjning 69
fördröjningsledning 120
förlusteffekt 68
- garanterade störmarginaler 75
gate 90
generellt register 45
genomslag 100
GND 9
grind 15
- H 6
HA 14
halvadderare 14
halvledarminne 116
heladderare 14
hexadecimal kod 31
hexadecimala talsystemet 32
high speed 77
high speed TTL 75
HiNIL 25
HLL 25
HNL 25
håll-tid 43
hög 6
högdopad 20
högerskift 45
högnivålogik 25
- icke 3
inclusive OR 12
inhibit 52
inhibit-signal 86
in/ut-enheter 146
inverterare 9
invertering 7
I/O-buss 18
IO-kretsar 146
isolation 20
isolationsdiffusion 23

- JK-vippa 38
- JKMS-vippa 41, 109
- Johnson-räknare 56
- joninplantering 100
- jämförare 12, 19

- kanal 90
- kapacitet 116
- kapacitiv belastning 68
- klippdiod 82
- klockad vippa 36
- klockan 37
- kodare 31
- kollektor 90
- kollektorbotten 23
- kombinationskrets 2
- kombinatorisk krets 2
- kombinatoriska funktioner 3
- kommutativa lagen 4
- komparator 12
- kondensatorstyre 93
- kristallarea 92
- kristallskiva 21
- köminne 124

- L 6
- laddningskopplare 122
- ledarmönster 23
- LIFO 123
- line driver 83, 86
- line receiver 83, 86
- linjekretsar 83
- logikfamilj 8
- logisk etta 6
- logisk nolla 6
- logiska grundfunktioner 3
- low power 77
- low power Schottky 76
- low power TTL 76
- LSB 16
- LSI 70
- LSL 25
- låg 6
- läsa 116
- läsminne 118, 124

- mask 21, 23
- maskprogrammerad 129
- mest signifikanta biten 16
- M_H 25
- mikrodator 144
- mikroprocessor 145
- minimera 5
- minne 30, 116
- minnescell 116

- minnesmatris 141
- M_L 25
- Modulo 2-räknare 39
- Morgans teorem 4, 5
- MOS 90
- MOS-resistor 93
- MOS-transistor 90
- MOS-vippa 93
- mottagare 86
- MSB 16
- MSI 70
- MS-vippa 35, 36, 39
- multiemittertransistor 60
- multiplexer 17
- MUX 17
- mästarna 35

- NAND 7
- NAND-grind 9, 104
- NBCD 32
- negativ logik 7
- nivåer 70
- nivåtriggning 41
- NMOS 92, 96
- "noll" 3
- nollställa 41
- NOR 8
- NOR-grind 102
- not 3

- och 3
- OH 70
- oktal kod 31
- OL 70
- omslagsförlopp 70
- omslagsområdet 64, 66
- or 3
- ord 116
- ordledningen 118
- ordlängd 116
- ordorganiserat 133
- oxid 20
- oxidhål 23

- parallellmatning 33
- parallell/serie-omvand-
ling 33
- parasitelement 100
- party line 87
- periferienheten 18
- pinch off-spänning 90
- pinnar 9
- PIPO 33
- PISO 33
- P-kanal 92

- planarprocessen 19
- planarteknik 19
- PMOS 92
- positionsvekt 31
- positiv flank 41
- positiv logik 6
- preparatoriska ingångar 38
- preset 42
- primärminne 116
- processorn 18
- programmed logic 129
- programmera 126
- programmering 129
- PROM 129
- pseudostabilt 36
- pull down 81
- pull up 80

- rader-PROM 129
- RAM 117, 118
- random access minnen 117
- refresh 94
- register 30
- reset 42
- ringräknare 53
- rippelräknare 48
- ROM 118, 124
- ROS 124
- RS-vippa 36
- RTL 8
- RTL-krets 21
- räknare 30, 38, 46

- S 14
- "sann" 2
- sanningstabell 3, 7
- SBD 78
- Schottky-diod 78
- Schottky-transistor 78
- Schottky TTL 76
- second source 110
- sekvenskrets 30
- sekventiella minnen 117
- serieingång 45
- seriematning 33
- serieminne 117, 120
- serie/parallell-omvand-
ling 33
- serieutgång 45
- signal 52
- signalgrind 15
- silicon gate 130
- sink 58
- SIPO 33
- SISO 33

skiftregister 44, 53
skiva 21
skriva 116
skriv-läsminne 118, 132
skyddskrets 102
slag 35
SOS 102
source 58, 90
specifikation 70
speed-power product 76
spänningsnivåer 70
spärrsignal 52
stack 123
stackminne 123
statisk 94, 132
statiska RAM 133
stegfördröjning 69
stoppkanal 102
strobas 86
strob-ingång 86
strypspänning 90
strömmatande logik 58
strömospik 66
strömsänkande logik 58
strömsänkningsförmåga 59
styre 90
styrledning 19
störmarginall 24, 25, 73,
74, 110
störspänning 73
substrat 97
summa 14
sving 24
synkron 36
synkronräknare 50
säkring 130
sändare 86
sätt-ingång 38
sätt-tid 43

TG 105
tidmätare 46
tidsmultiplex 17
tillstånd 54
totempåle 64
trade offs 23
transientminne 35, 39
transmissionsgrind 105
tristate 83, 85
tråda 11
trådas 64
TTL-kretsar 58
turn-around-tid 129

tvinnade parledningar 86
T-vippa 38
tvåfasklocka 36
typgrind 7
typiska störmarginaler 75

UL 72
UPP/NER-räknare 49
uttag 9

VCC 9
vänsterskift 45
värsta fallet 71

wired-AND 11
wired logic 129

XOR 12

yield 21, 63

åtkomsttid 117

öppen kollektor 11, 59,
64, 83, 84
överföringsdiagram 64

Engelskt sakregister

above 71
active pull-up 84
AND 3
AND OR INVERT 12
AOI 12
arithmetic device 82
asynchronous counter 48

below 71
bidirectional switch 107
binary coded decimal 32
bucket brigade device 120
bus 16
bus drivers 139
byte 116

carry 14
central clock 37
central processing unit 18, 145
charge coupled device 120
chip select 136
clear 41
clock pulse 37
comparator 12
control lines 19
count 52
counter 48, 82

data selector 17
decoder 31, 82
demultiplexer 17
die 21
digital circuits VI
DIL 9
diode clamp 82
diode transistor logic 2
diode transistor logic with zener diode 25
disable 52, 85
divide by N 47
divide by N counter 48
drain 90
drivers 83
DTL 7
DTLZ 25
dual in line 9

edge triggered 39
emitter coupled logic 26
enable 52
encoder 31
erasable field programmable 129
exclusive OR 12
EXOR 12

fan out 59, 72
FET 90
field programmable 129
first in first out 123
flip-flop 82
full adder 14
fusible links 129

gate 15, 37, 82, 90
GND 9

half adder 14
high level logic 25
high noise immunity logic 25
high speed TTL 75
HiNIL 25
HLL 25
hold time 43

inclusive OR 12
inhibit 52, 86
input/output bus 18
inverter 9
I/O-bus 18

Johnson counter 55, 56

langsam störsicher Logik 25

large scale integration 70
last in first out 123
least significant bit 16
level operated 41
line driver 82, 83, 86
line receiver 82, 83, 86
load 53
low power Schottky 76
low power TTL 76
LSL 25

main frame memories 116
mask programmed 129
master slave 39
medium scale integration 70
memory 116
modulo N 47
monostable 82
MOS 90
most significant bit 16
MSB 16
multiplexer 17, 82

NAND-gate 9
noise immunity 74
noise immunity margin 74
noise margin 74
NOR 8
NOT 3
N-type epitaxial layer 20

open collector 11, 59
OR 3
output high 70
output loading capability 59, 72
output low 70

party line 87
PIPO 33
PISO 33
positive going edge 41
preparatory inputs 38
preset 41
programmed logic 128, 129
propagation delay 69
P-type isolation 20
pull down 81
pull up 80

RAM 118
random access memory 118
read 118
read only memory 118, 124
read only storage 124
refresh 94
reset 38
resistor transistor logic 8
ring counter 53
rippel count 53

rippel counter 48
ROM 118
RTL 8
SBD 78
Schottky TTL 76
second source 110
self alignment 130
sequential circuit 30
serial input 45
set 38
set-up time 43
shift register 82
silicon on sapphire 102
sink 58
sinking logic 58
SIPO 33
SISO 33
small scale integration 82
source 58, 90
sourcing logic 58
speed-power product 76
SSI 82
stand by 139
store 116
strobe 86
surface barrier diode 78
switchtail ring 55
synchronous counter 48

toggle 37, 39
totem-pole 64
trade offs 23
transition operated 39, 41
transmission gate 105
tristate 83, 85
truth table VI
twinned pairs 86

unit load 72
UP/DOWN-counter 49

 V_{CC} 9

wafer 21
walking ring counter 53
wired-AND 11, 64
wired logic 128, 129
wired-OR 11
word 116
worst case 71
write 118

XOR 12

yield 21, 63

Digitala kretsar

är andra delen i en serie elektronikböcker som kan användas vid undervisningen på gymnasieskolan, för introduktiv högskoleundervisning och för fortbildning inom industrin.

Delarna Pulskretsar och Digitala kretsar ersätter den tidigare läroboken "Elektronik, 1 Digitala kretsar".

I serien ingår även delarna Analoga kretsar och Effektkretsar.

 ESSELTE STUDIUM

ISBN 91-24-26209-9

Best.nr 24-26209-9

(24-26209-9) B