

F1: Introduktion Digitalkonstruktion II, 4p

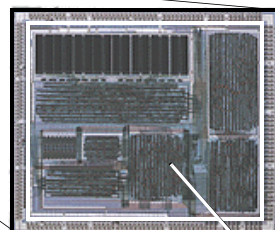
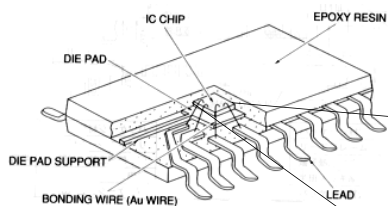
- **Översikt:**

- **Föreläsare:** Bengt Oelmann
- **Kurslitteratur:** "Principles of CMOS VLSI Design - A systems Perspective"
- **Föreläsningar:** 16
- **Räkneövningar:** 8
- **Laborationer:** 4
- **Målsättning:**
 - ge grundläggande kunskaper om strukturerad VLSI konstruktion
 - ge grundläggande kunskaper om simulering och konstruktionsverifiering
 - ge grundläggande kunskaper om testbarhet
 - kännedom om verktyg för VLSI konstruktion

- **Målsättning med F1:**

- **Ge en överblick över vilka grundkomponenter som bygger upp ett digitalt CMOS system och hur man konstruerar ett system med dessa.**
- **Innehåll**
 - Exempel på VLSI system baserat på CMOS
 - CMOS switchen
 - Kombinatorisk logik
 - Minneselement
 - Beskrivningsmetoder för CMOS system

Integrerad krets



padframe

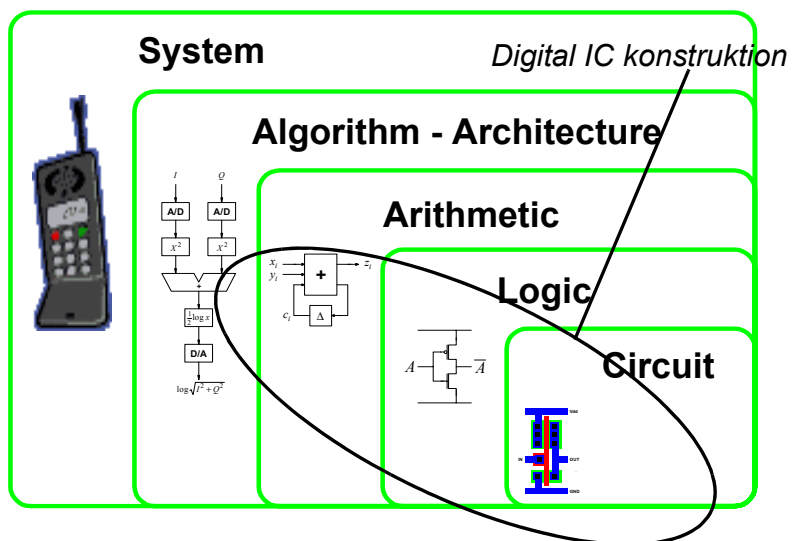
core

Padframe består av anslutningspunkter för in- och utsignaler samt matningsspänning till kretsen.

Core är kretsar för att utföra den funktion man önskar.

- i grunden uppbyggd av transistorer
- typiska byggblock är logiska grindar RAM- och ROM-minnen

Digital IC konstruktion



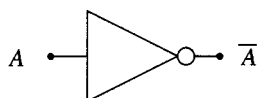
Exempel på VLSI system baserat på CMOS

- Mikroprocessorer
- Komplexa digitala kretsar med hög beräkningskapacitet
- FPGA:er (FPGA= Field Programmeble Gate Arrays)
- digitala ASIC:s (ASIC=Application Specific Integrated Circuits)
- Analoga och mixed analoga digitala kretsar med moderata krav på frekvens
- Exempel: Digital Equipments ALPHA processor

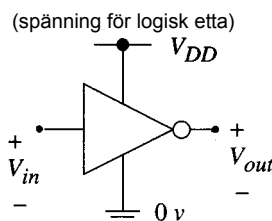
Alpha 21164 Microprocessor Fact Sheet	
Clock Speed	600 MHz
Manufacturing process	0.35 μm
Die size	208 mm ²
Transistor count	9.6 Million
CPU Voltage	2.4 V
I/O Voltage	3.3 V
Power	45 W
Package	499-pin PGA (57x57 mm)

Elektroniska logiska grindar

Elektronisk krets som beteer sig som en logisk grind



logisk symbol



elektriska parametrar

logiska variabler ersätts med spänningar

A	$\bar{A}$
0	1
1	0

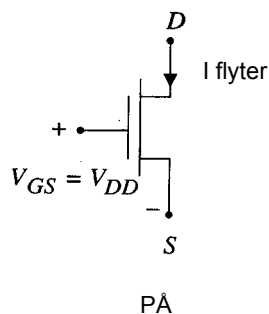
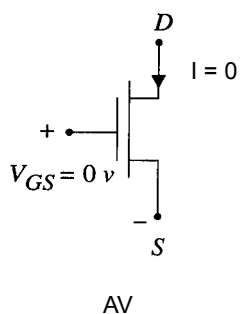
icke-operatorns sanningstabell

V_{in}	V_{out}
0v	V_{DD}
V_{DD}	0v

ersatt med spänningar

Ström genom en MOSFET av n-typ (nMOS)

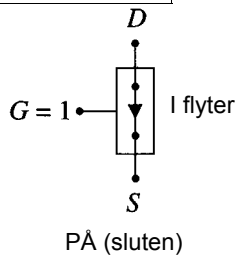
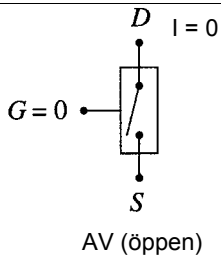
- nMOS har tre anslutningar:
 - Gate (G)
 - Source (S)
 - Drain (D)
- Spänningen V_{GS} kontrollerar strömmen i transistor
- I digitala kretsar är transistorns tillstånd AV ($V_{GS}=0V$) och PÅ ($V_{GS}=V_{DD}$) av intresse.



nMOS som en kontrollerbar switch

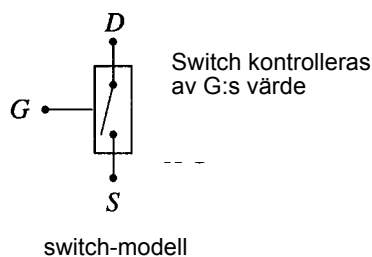
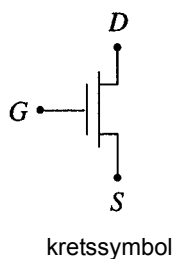
- Inför en ideal switch-modell för nMOS
- Definiera switchens funktion som:

spänning V_{GS}	logiskt värde	switchens tillstånd
0 V	0	AV
V_{DD}	1	PÅ



Switch-modell för nMOS

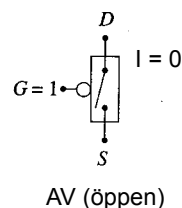
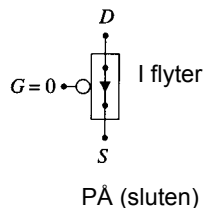
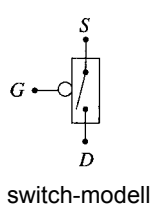
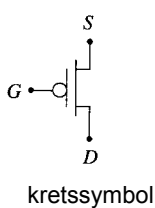
- Switchen kallas för
 - spänningskontrollerad switch (då spänningar används)
 - logikkontrollerad switch (då logiska värden används)



Ström genom en MOSFET av p-typ (pMOS)

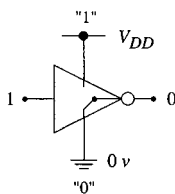
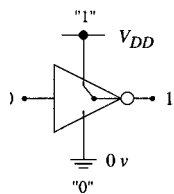
- Den andra typen av transistor kallas för pMOS
- är den logiska och elektriska komplementet till nMOS, d.v.s:
 - polariteten för spänningarna och strömriktningen i transistorn är omvända

spänning V_{GS}	logiskt värde	switchens tillstånd
0 V	0	PÅ
V_{DD}	1	AV



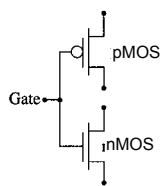
Komplementära par av nMOS och pMOS

• Konstruera en inverterare ...

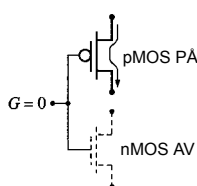


0 motsvarar 0V
1 motsvarar V_{DD}

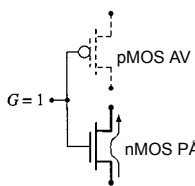
• ... med MOS transistorer



nMOS/pMOS par



låg gate-spänning



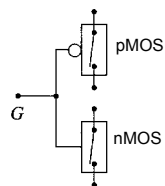
hög gate-spänning

Komplementärt par:

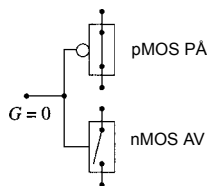
- en nMOS och en pMOS med sammankopplade gate-elektroder
- Den gemensamma G-signalen kontrollerar den logiska grinden
- då en transistor är PÅ så är den andra AV.

MOSFET par som inverterare

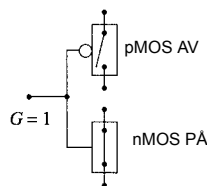
- CMOS är förkortning av Complementary MOSFET
- Digitalt beteende på CMOS krets



pMOS/nMOS par

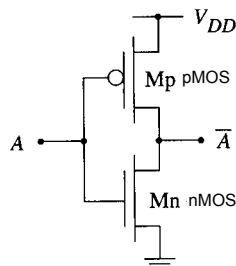


(b) $G = 0$ input



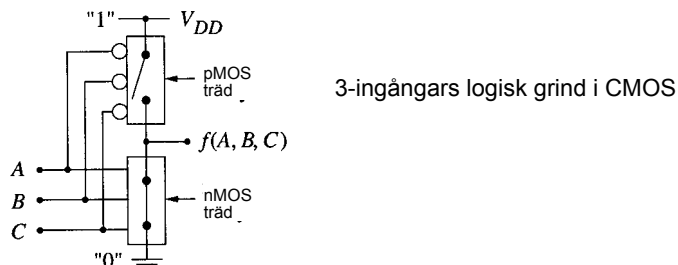
(c) $G = 1$ input

• CMOS inverterare



Logiska grindar i CMOS

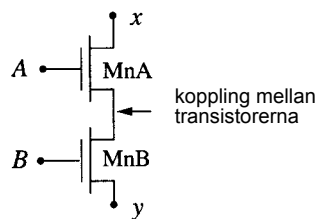
- ICKE-operationen kan utökas till t.ex NOR- och NAND-operationer
- Grundläggande struktur:



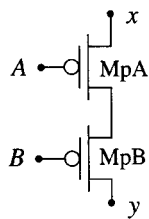
- För en sådan krets måste ett av de två följande påståenden vara sant för samtliga kombinationer av insignaler:
 - pMOS trädets är AV (öppet) och nMOS trädets är PÅ (slutet)
 - pMOS trädets är PÅ (slutet) och nMOS trädets är AV (öppet)
- $f(A,B,C)$ tilldelas endast ett av värdena 1 eller 0.

13 (29)

Seriekoppling av MOS transistorer



x kopplas till y
om och endast om
 $A=1$ OCH $B=1$



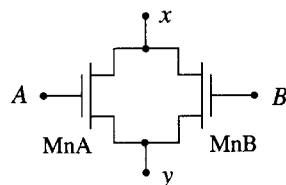
x kopplas till y
om och endast om
 $A=0$ OCH $B=0$

- Seriekoppling verkar som en AND-funktion

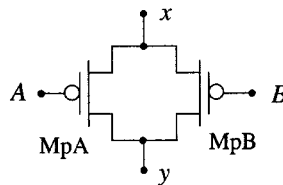
- nMOS:
Om både $A=1$ OCH $B=1$,
då kopplas x till y
- pMOS:
Om både $A=0$ OCH $B=0$,
då kopplas x till y

14 (29)

Parallellkoppling av MOS transistorer



x kopplas till y
om antingen
 $A=1$ ELLER $B=1$



x kopplas till y
om antingen
 $A=0$ ELLER $B=0$

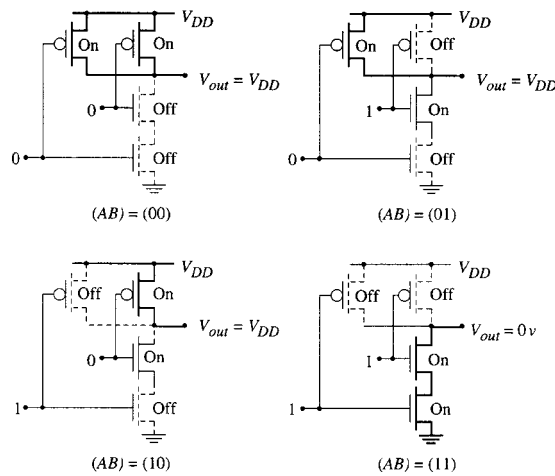
- **Parallellkoppling verkar som en OR-funktion**

- **nMOS:**
Om $A=1$ ELLER $B=1$,
då kopplas x till y
- **pMOS:**
Om $A=0$ ELLER $B=0$,
då kopplas x till y

Syntes av grundläggande grindar i CMOS

- De grundläggande grindarna i CMOS är NOT, NAND och NOR
- Följande observationer kan göras:
 - seriekopplade nMOS transistorer kan användas för AND funktionen
 - parallellkopplade nMOS transistorer kan användas för OR funktionen
 - Vi kan göra ekvivalenta påståenden för pMOS transistorer. Hänsyn måste tas till att de är aktivt låga. Med deMorgans relation kan vi skriva om
 - seriekopplingen som $\overline{A \cdot B} = \overline{A} + \overline{B}$
 - parallellkopplingen som $\overline{A + B} = \overline{A} \cdot \overline{B}$
 - seriekopplade pMOS transistorer kan användas för NOR funktionen
 - parallellkopplade pMOS transistorer kan användas för NAND funktionen

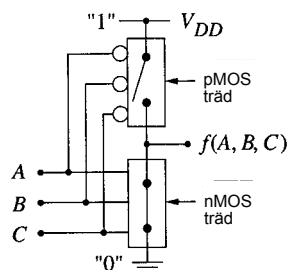
Exempel: 2-ingångars NAND grind (NAND2)



17 (29)

Syntes av komplexa logiska grindar

- Komplexa funktioner kan byggas upp av ett antal kaskadkopplade enkla funktioner som NAND, NOR och NOT
- Med komplexa CMOS grindar får man en lösning som är snabbare och mer kompakt (färre transistorer)
- En komplex grind har samma karakteristik som en enkel grind
 - Komplementära par av MOSFET transistorer används där varje ingång är kopplad till en nMOS och en pMOS
 - Den konstrueras enligt den generella modellen:



18 (29)

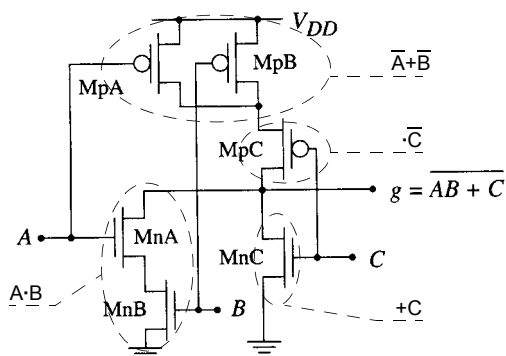
Syntes av komplexa CMOS grindar, metod 1

- Utgå från det logiska uttrycket
- Bilda nMOS trädet genom att bilda serie- och parallellkopplingar utifrån AND- och OR funktioner
- Bilda pMOS trädet genom inversen av nMOS trädet
- Exempel: logisk funktion $Z = \overline{AB + C}$

$$Z_n = AB + C$$

Z_p är inversen av Z_n , alltså:

$$Z_p = \overline{AB + C} = (\overline{A + B})\overline{C}$$



19 (29)

Syntes av komplexa CMOS grindar, metod 2

- Utgå från Karnaugh diagram
- Bilda nMOS-träd
 - ringa in '0'-orna och skriv upp uttrycket på summa-av-produktform (SOP)
- Bilda pMOS-träd
 - ringa in '1'-orna och skriv upp uttrycket på summa-av-produktform (SOP)
 - invertera alla ingående invariabler i SOP-uttrycket
- Rita up transistorschema for n- och pMOS träd enligt uttrycket

	BC			
	00	01	11	10
A	0	1	0	0
1	1	0	0	0

$\overline{B}\overline{C}$ C AB $\overline{A}\overline{C}$

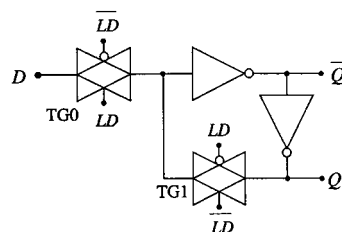
$$Z_n = AB + C$$

$$Z_p = \overline{A}\overline{C} + \overline{B}\overline{C} = (\overline{A+B})\overline{C}$$

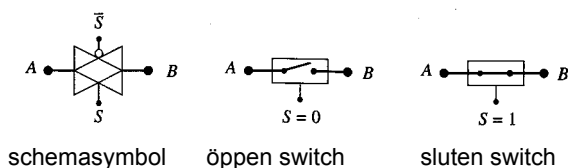
20 (29)

Minneselement (latch)

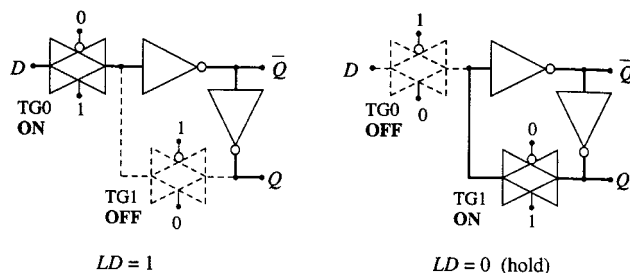
- I CMOS byggs en latch upp av
 - Transmissiongrindar (TG) som kontrollerar dataflödet
 - Inverterare som utgör ett bistabilt minneselement



Load-signalen LD styr TG0 och TG1 så att en, och endast en, alltid är öppen.



Latchens funktion

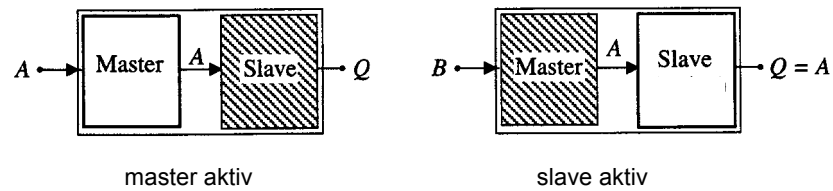


$LD=1$: Latchen är transparent. Signalen D passerar TG0 och genom inverterarna. TG1 är öppen och det tidigare Q-värdet återförs ej.

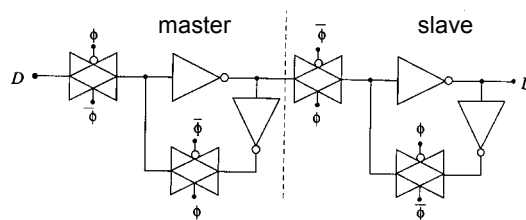
$LD=0$: Latchen låser Q-värdet (eng. latch mode /hold). Signalen D spärras av TG0. Q-värdet återförs genom TG1.

Minneselement (D Flip-Flop)

- Master-slave DFF



- Implementering -- två kaskadkopplade D-latchar



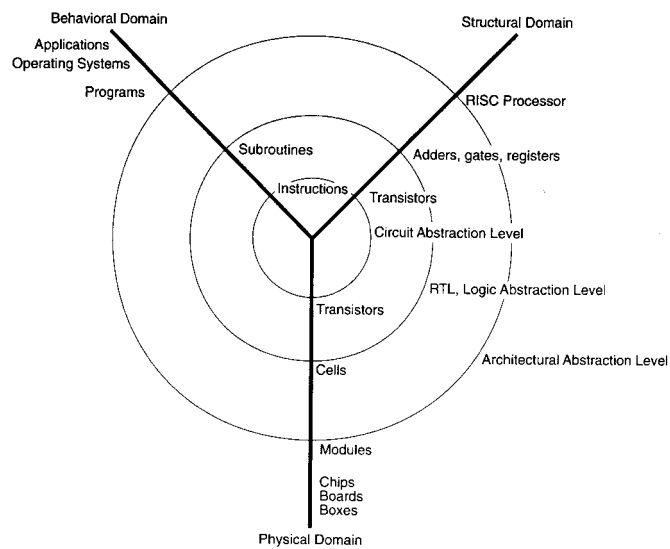
23 (29)

Beskrivningsmetoder för CMOS system

- Genom att införa olika nivåer av abstraktioner kan komplexa konstruktioner byggas.
- Ett elektroniskt system kan beskrivas i 3 olika domäner:
 - Beteende (behavioral) - specificerar vad systemet gör
 - Struktur (structural) - specificerar hur enheterna i systemet är sammankopplade
 - fysisk (physical) - specificerar hur strukturerna ska byggas
- Vart och ett av dessa domäner kan beskrivas i ett antal abstraktionsnivåer

24 (29)

Y-diagram



25 (29)

Beteende beskrivning

- Beskriver hur en speciell krets ska uppföra sig för givna indata
- Kretsens beteende kan t.ex ges i form av:
 - Booliska uttryck
 - Tabeller
 - Algoritmer skrivna i något HDL (Hardware Description Language) språk
- Exempel på olika abstraktionsnivåer
 - Algoritm:


```
if (d<0) d=d+a;
else d = d+b;
```
 - Logiskt beteende:

$$S = \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC$$

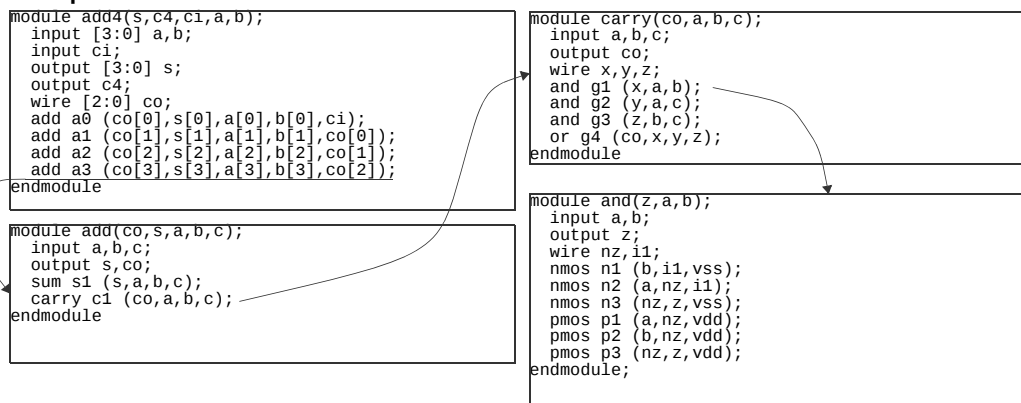
$$CO = AB + AC + BC$$
 - Elektriskt beteende:

$$I_{DS} = \frac{kW}{2L}(V_{GS} - V_T)^2(1 + \lambda V_{DS})$$

26 (29)

Strukturell beskrivning

- Specificerar hur komponenter ska kopplas samman för att det önskade beteendet ska fås
- Kretsens struktur ges i form av en lista av komponenter och hur dessa är sammankopplade (nätlista)
- De lägre abstraktionsnivåerna avslöjar mer detaljer om implementeringen av kretsen
- Exempel:



27 (29)

Fysisk beskrivning

- Den fysiska specifikationen beskriver hur en struktur ska byggas
 - I en IC process beskrivs kretsen genom en fotomask
 - Material
 - Dimensioner
 - Placering
 - Text en metalledare:
 - längd: 100 μm
 - bredd: 3 μm
 - material: aluminium
 - startkoordinat: (3.00, 3.00), slutkoordinat: (70.00, 5.00)

28 (29)

Summering

- Genom att betrakta en MOS transistor som en switch kan grundläggande logiska grindar tas fram
- Olika typer av sätt att beskriva elektroniksystem har diskuterats
- Många av de saker som har tagits upp, kommer att gås igenom mer i detalj i kommande föreläsningar