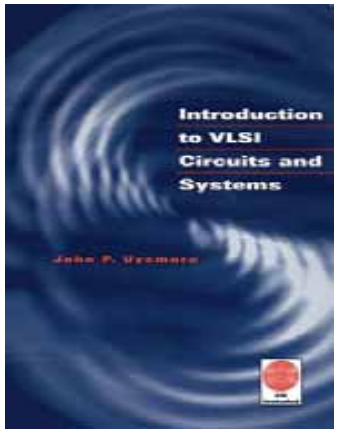
國立勤益科技大學
National Chin-Yi University of Technology

Introduction to VLSI Circuits and Systems
積體電路概論

Chapter 02
Logic Design with MOSFETs



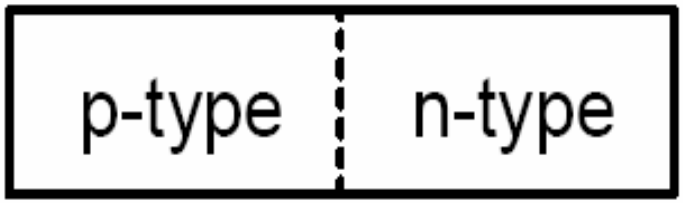
賴秉樑
Dept. of Electronic Engineering
National Chin-Yi University of Technology
Fall 2007

Outline

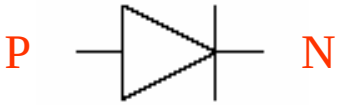
- The Fundamental MOSFETs
- Ideal Switches and Boolean Operations
- MOSFETs as Switches
- Basic Logic Gates in CMOS
- Complex Logic Gates in CMOS
- Transmission Gate Circuits
- Clocking and Dataflow Control

p-n Junction

- A junction between *p-type* and *n-type* semiconductor forms a **diode**
- Current flows only in one direction

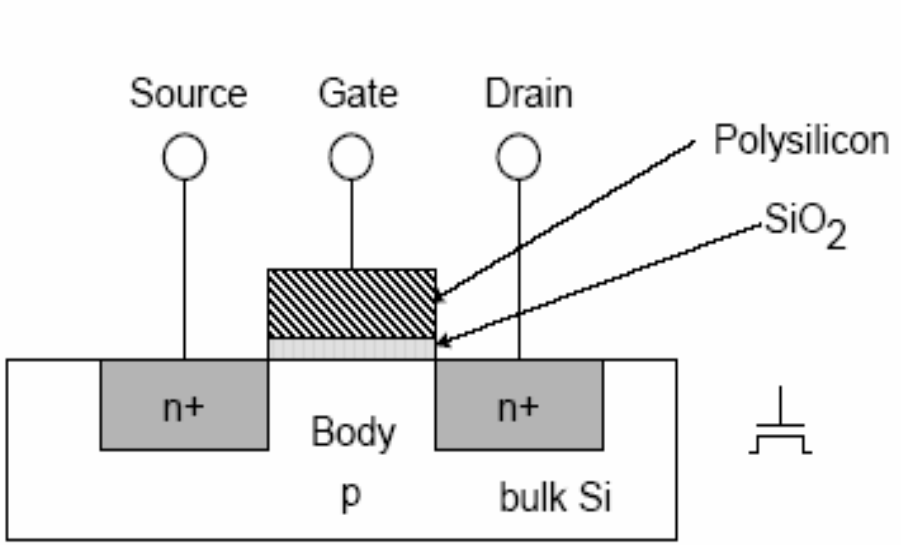


anode cathode



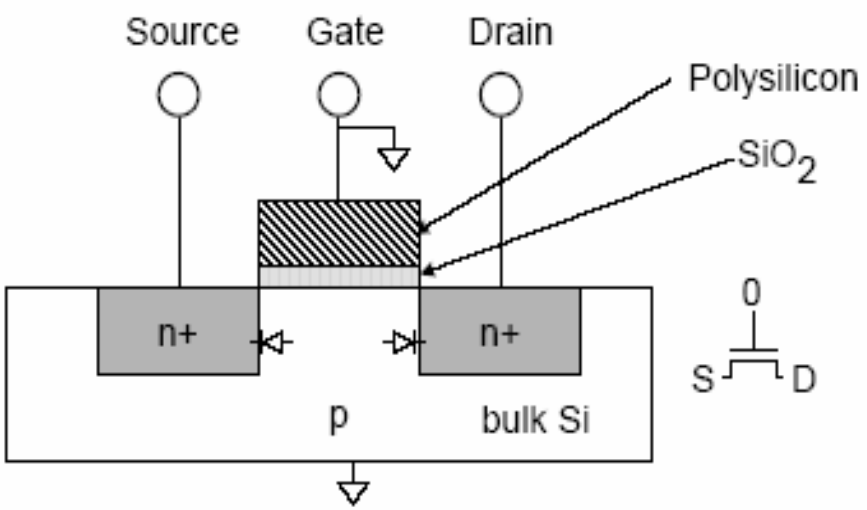
nMOS Transistor

- Four terminals: gate (G), source (S), drain (D), body (B)
- Gate–oxide–body stack looks like a capacitor
  - Gate and body are conductors
  - SiO_2 (oxide) is a very good insulator
  - Called metal – oxide – semiconductor (MOS) capacitor
  - Even though gate is no longer made of metal



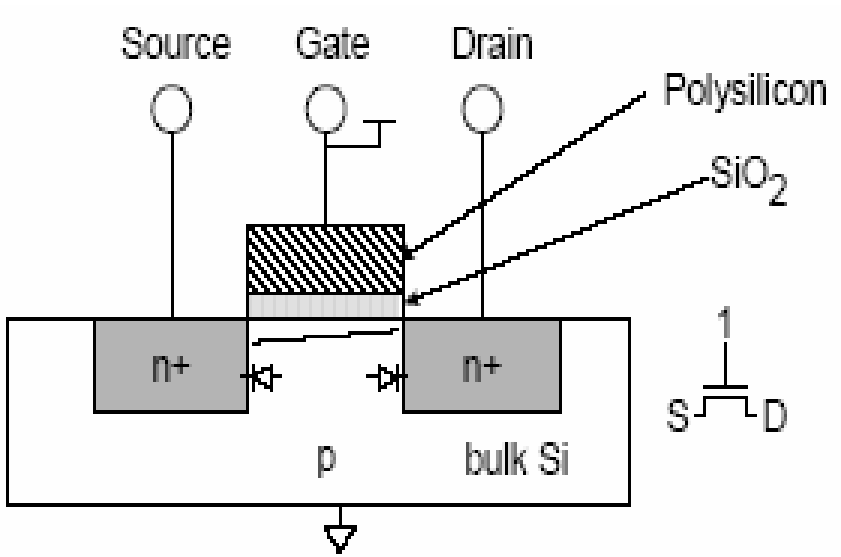
nMOS Operation (1/2)

- Body is usually tied to ground (0 V)
- When the gate is at a low voltage
  - P-type body is at low voltage
  - Source-body and drain-body diodes are OFF
  - No current flows, transistor is OFF



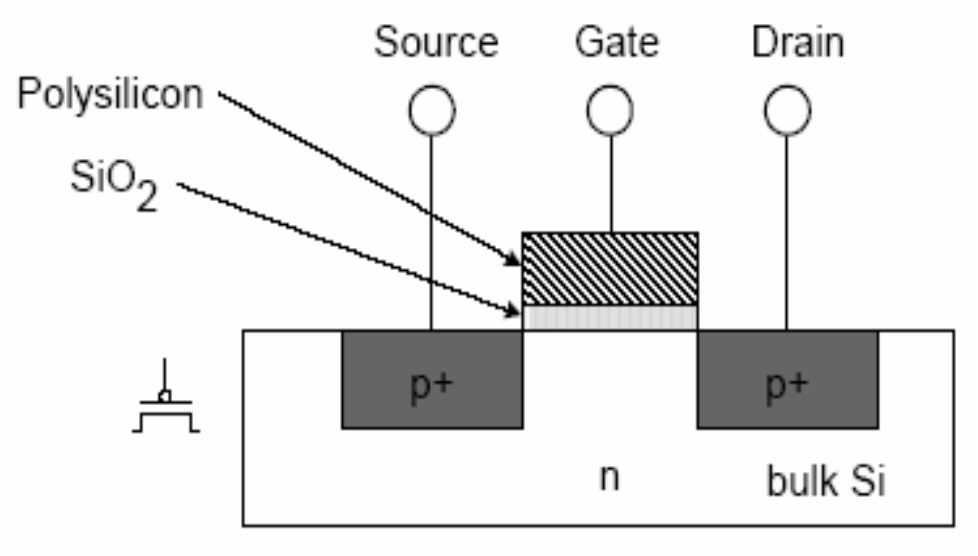
nMOS Operation (2/2)

- When the gate is at a high voltage
  - Positive charge on gate of MOS capacitor
  - Negative charge attracted to body
  - Inverts a channel under gate to n-type
  - Now current can flow through n-type silicon from source through channel to drain, transistor is ON



pMOS Transistor

- Similar, but doping and voltages reversed
  - Body tied to high voltage (VDD)
  - Gate “low”: transistor ON
  - Gate “high”: transistor OFF
  - Bubble indicates inverted behavior

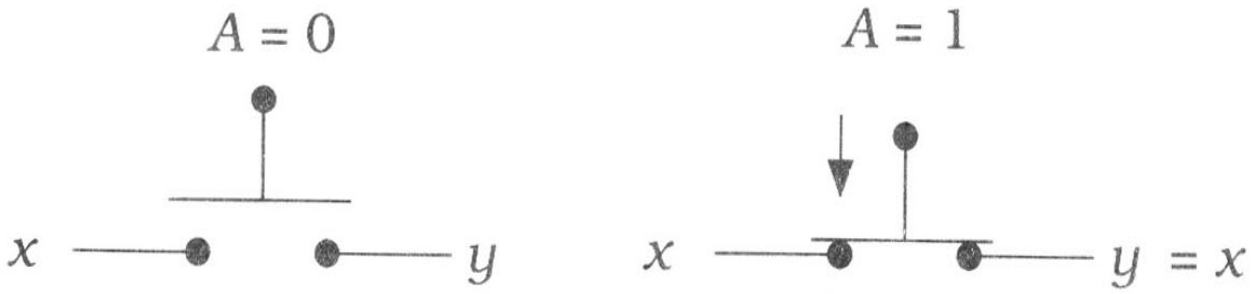


Outline

- The Fundamental MOSFETs
- Ideal Switches and Boolean Operations
- MOSFETs as Switches
- Basic Logic Gates in CMOS
- Complex Logic Gates in CMOS
- Transmission Gate Circuits
- Clocking and Dataflow Control

Ideal Switches (1 / 3)

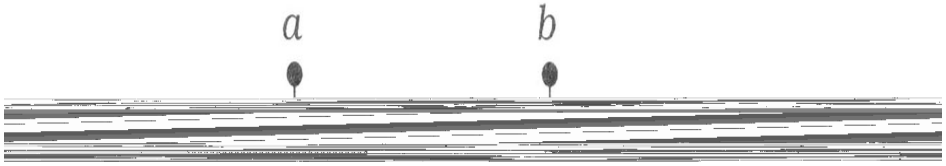
- CMOS integrated circuits use bi-directional devices called MOSFETs as logic switches
  - » Controlled switches, e.g, assert-high and assert-low switches
- An assert-high switch is showing in Figure 2.1



(a) Open (b) Closed

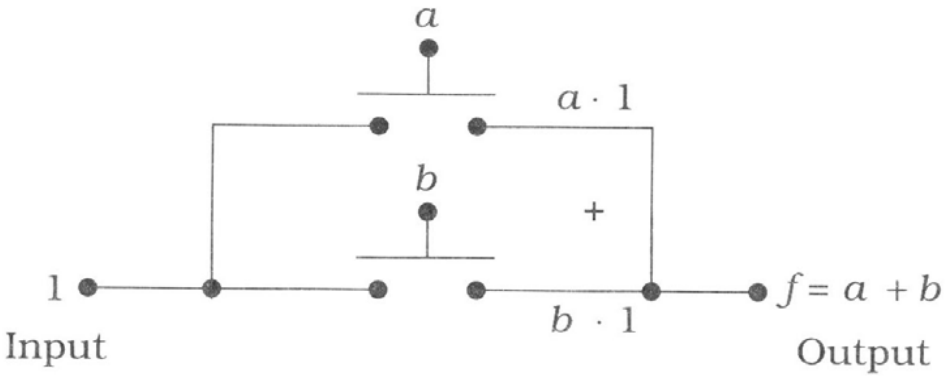
Figure 2.1 Behavior of an assert-high switch

Ideal Switches (2 / 3)



$g = (a.\ 1).\ b = (a.\ 1).\ b$

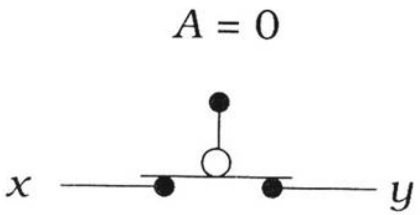
Figure 2.2 Series-connected switches



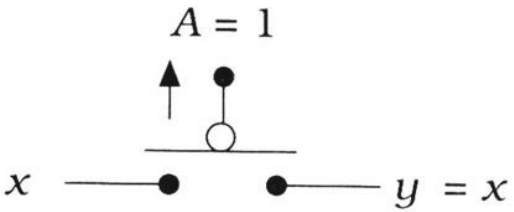
$g = (a.\ 1) + (b.\ 1) = a + b$

Figure 2.4 Parallel-connected switches

Ideal Switches (3 / 3)



(a) Closed



(b) Open

Figure 2.5 An assert-low switch

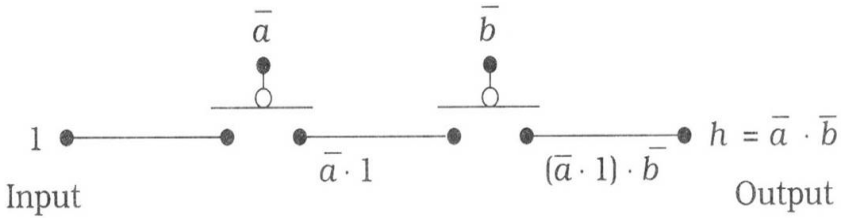


Figure 2.6 Series-connected complementary switches

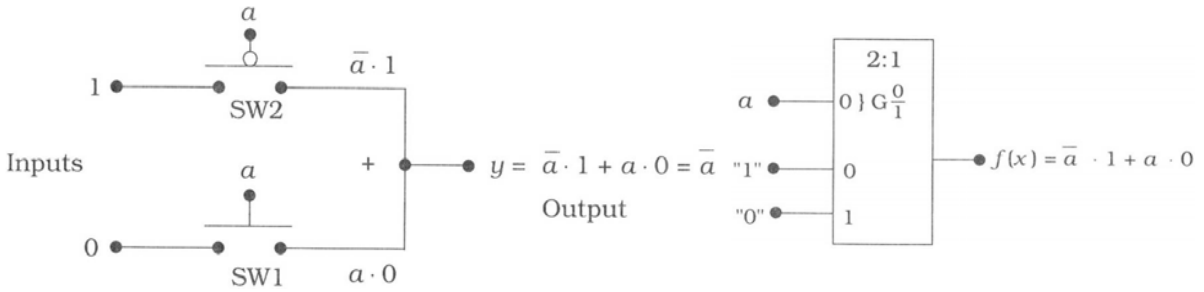


Figure 2.7 An assert-low switch

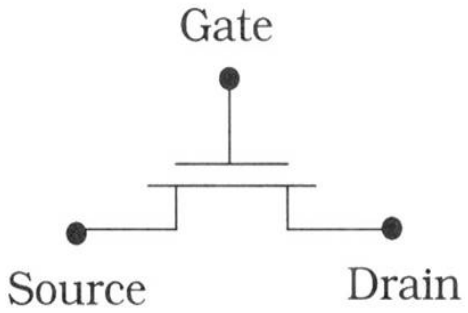
Figure 2.8 A MUX-based NOT gate

Outline

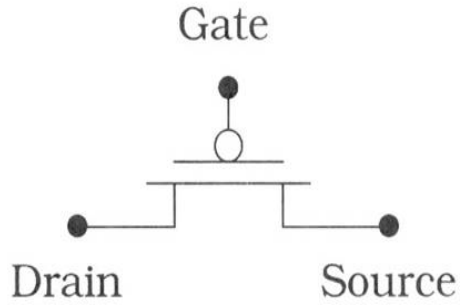
- The Fundamental MOSFETs
- Ideal Switches and Boolean Operations
- MOSFETs as Switches
- Basic Logic Gates in CMOS
- Complex Logic Gates in CMOS
- Transmission Gate Circuits
- Clocking and Dataflow Control

MOSFET as Switches

- MOSFET: Metal-Oxide-Semiconductor Field-Effect Transistor
- nFET: an n-channel MOSFET that uses negatively charged electrons for electrical current flow
- pFET: a p-channel MOSFET that uses positive charges for current flow
- In many ways, MOSFETs behave like the *idealized switches* introduced in the previous section
- The voltage applied to the gate determines the current flow between the source and drain terminals



(a) nFET symbol



(b) pFET symbol

Figure 2.9 Symbols used for nFETs and pFETs

MOSFET as Switches

- Early generations of silicon MOS logic circuits used both *positive* and *negative supply voltages* as Figure 2.10 showing
- In modern designs require only a single positive voltage V_{DD} and the ground connection, e.g. V_{DD} = 5 V and 3.3 V or lower
- The relationship between logic variables *x* and it's voltages V_x

$$0 \le V_x \le V_{DD} \qquad (2.14)$$

$$\begin{cases} x = 0 \ \textit{means that} \ V_x = 0V \\ x = 1 \ \textit{means that} \ V_x = V_{DD} \end{cases} \qquad (2.15)$$

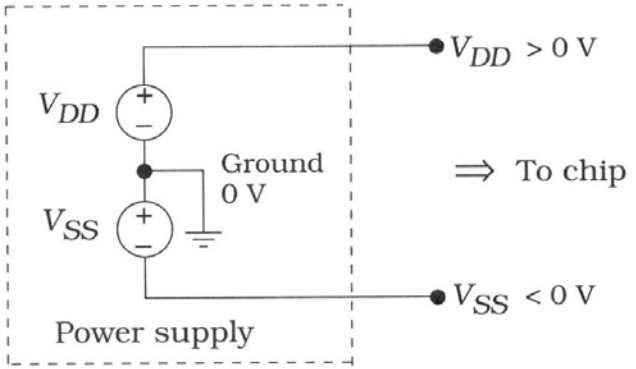
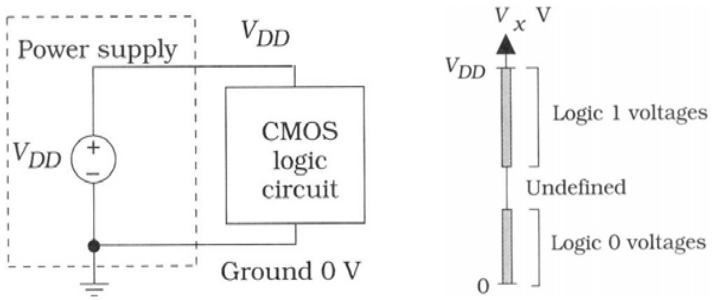


Figure 2.10 Dual power supply voltages

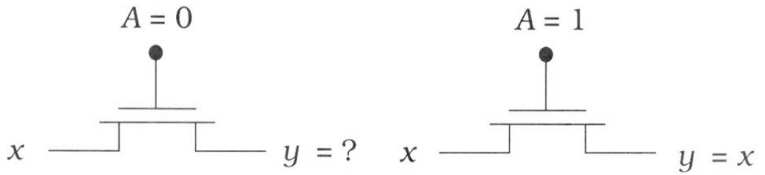


(a) Power supply connection (b) Logic definitions

Figure 2.11 Single voltage power supply

Switching Characteristics of MOSFET

- In general,
  - Low voltages correspond to logic 0 values
  - High voltages correspond to logic 1 values
  - » The transition region between the highest logic 0 voltage and the lowest logic 1 voltage is undefined



(a) Open (b) Closed

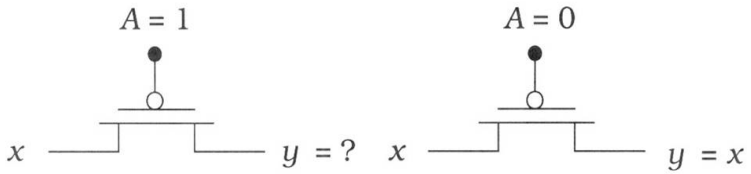
Figure 2.12 nFET switching characteristics

- nFET

$$y = x \cdot A \quad \textit{which is valid iff} \quad A = 1 \qquad (2.16)$$

- pFET

$$y = x \cdot \overline{A} \quad \textit{which is valid iff} \quad A = 0 \qquad (2.17)$$



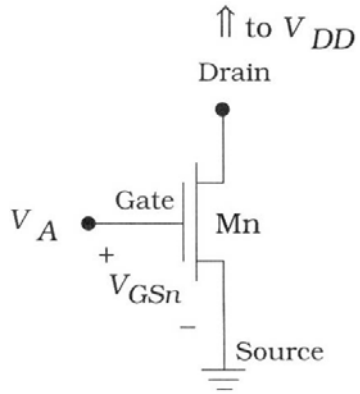
(a) Open (b) Closed

Figure 2.13 pFET switching characteristics

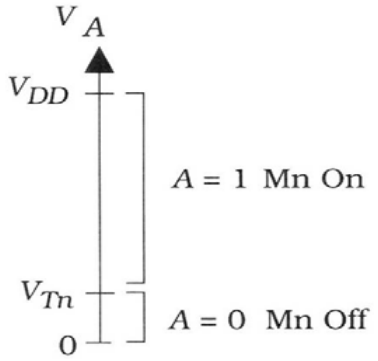
nMOS FET Threshold Voltages

- An nFET is characterized by a threshold voltage V_{Tn} that is positive, typical is around V_{Tn} = 0.5 V to 0.7 V
- If $V_{GSn} \le V_{Tn}$, then the transistor acts like an open (**off**) circuit and there is no current flow between the drain and source
- If $V_{GSn} \ge V_{Tn}$, then the nFET drain and source are connected and the equivalent switch is closed (**on**)
- Thus, to define the voltage V_A that is associated with the binary variable A

$$V_A = V_{GSn} \qquad (2.20)$$



(a) Gate-source voltage



(b) Logic translation

Figure 2.14 Threshold voltage of an nFET

pMOS FET Threshold Voltages

- An pFET is characterized by a threshold voltage V_{Tp} that is negative, typical is around V_{Tp} = –0.5 V to –0.8 V
  - » If $V_{SGp} \le |V_{Tp}|$, then the transistor acts like an open (**off**) switch and there is no current flow between the drain and source
  - » If $V_{SGp} \ge |V_{Tp}|$, then the pFET drain and source are connected and the equivalent switch is closed (**on**)
- Thus, to the applied voltage V_A we first sum voltage to write

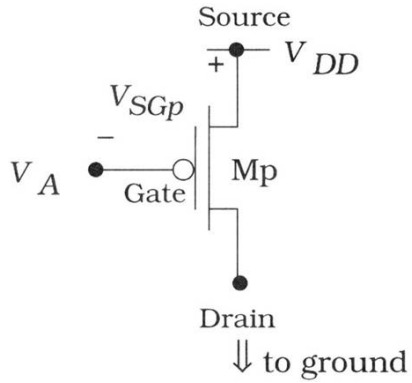
$$V_A + V_{SGp} = V_{DD} \qquad (2.23)$$

$$\Rightarrow V_A = V_{DD} - V_{SGp} \qquad (2.24)$$

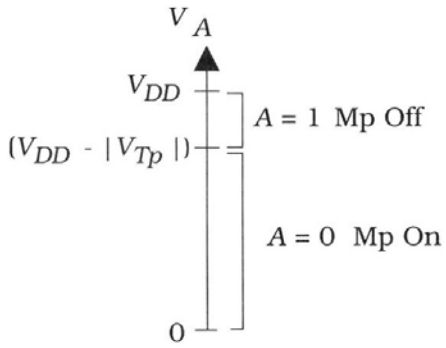
$$V_{DD} - |V_{Tp}| \qquad (2.25)$$

$$\begin{cases} V_A = 0\,V \\ V_A = V_{DD} \end{cases} \qquad (2.26)$$

Note that the transition between a logic 0 and a logic 1 is at (2.25) !



(a) Source-gate voltage



(b) Logic translation

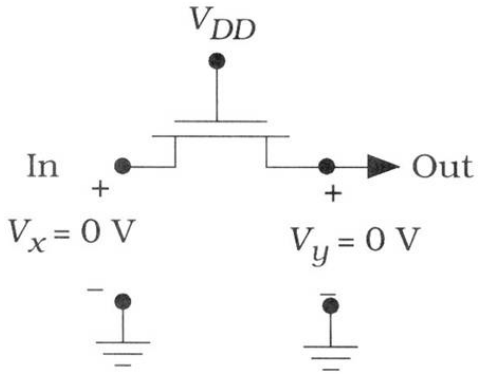
Figure 2.15 pFET threshold voltage

nFET Pass Characteristics

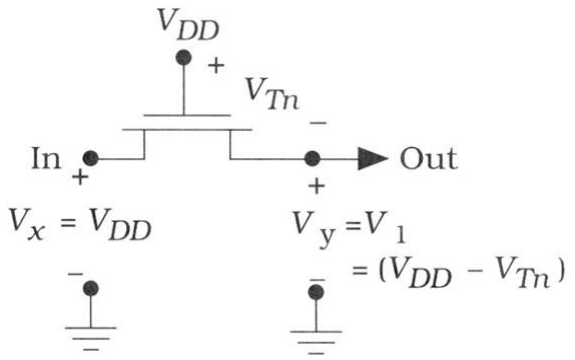
- An ideal electrical switch can pass any voltage applied to it
- As Figure 2.16(b), the output voltage V_y is reduced to a value

$$V_1 = V_{DD} - V_{Tn} \quad (2.27) \quad since \quad V_{GSn} = V_{Tn}$$

  - » Which is less than the input voltage VDD, called **threshold voltage loss**
- Thus, we say that the nFET can only pass a **weak logic 1**; in other word, the nFET is said to pass a **strong logic 0** ➔ can pass a voltage in the range [0, V_1]



(a) Logic 0 transfer



(b) Logic 1 transfer

Figure 2.16 nFET pass characteristics

pFET Pass Characteristics

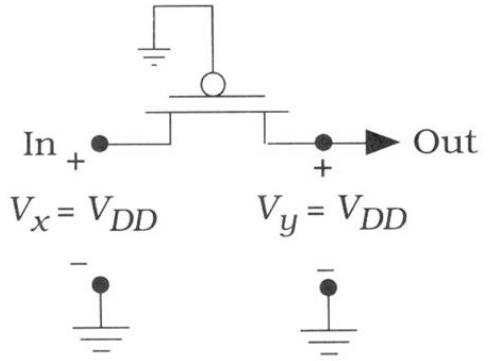
- Figure 2.17(a) portrays the case where V_x = V_{DD} corresponding to a logic 1 input. The output voltage is

$$V_y = V_{DD} \quad (2.29)$$,which is an ideal logic 1 level

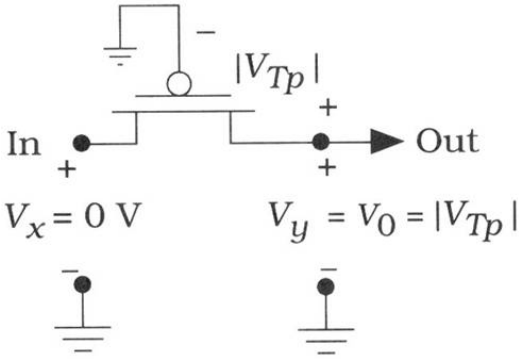
- Figure 2.17(b), the transmitted voltage can only drop to a minimum value of

$$V_y = |V_{Tp}| \quad (2.30) \quad since \quad V_{SGp} = |V_{Tp}|$$

- The results of the above discussion
  - » nFETs pass strong logic 0 voltages, but weak logic 1 values
  - » pFETs pass strong logic 1 voltages, but weak logic 0 levels
  - » Use pFETs to pass logic 1 voltages of V_{DD}
  - » Use nFETs to pass logic 0 voltages of V_{SS} = 0 V



(a) Logic 0 transfer



(b) Logic 1 transfer

Figure 2.17 pFET pass characteristics

Outline

- The Fundamental MOSFETs
- Ideal Switches and Boolean Operations
- MOSFETs as Switches
- Basic Logic Gates in CMOS
- Complex Logic Gates in CMOS
- Transmission Gate Circuits
- Clocking and Dataflow Control

Basic Logic Gates in CMOS

- Digital logic circuits are nonlinear networks that use transistors as electronic switches to divert one of the supply voltages V_{DD} or 0 V to the output
- The general switching network

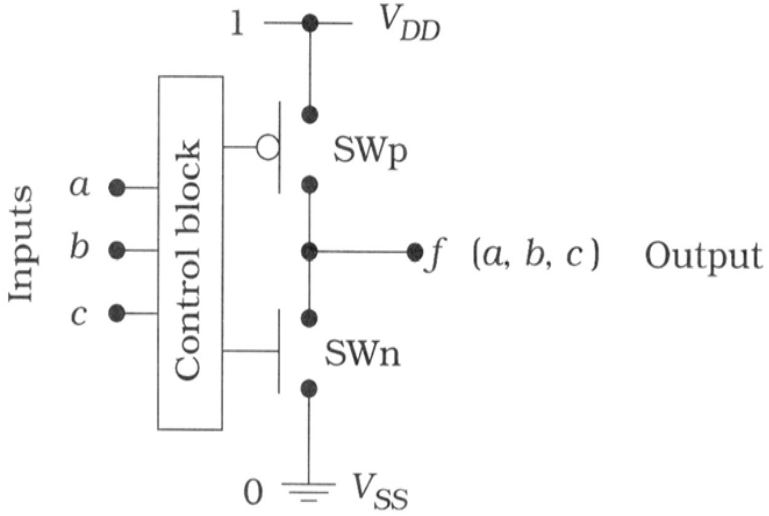
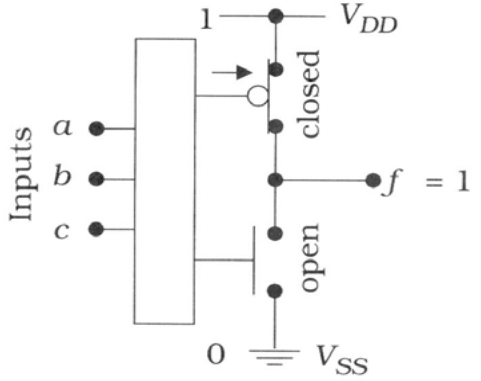
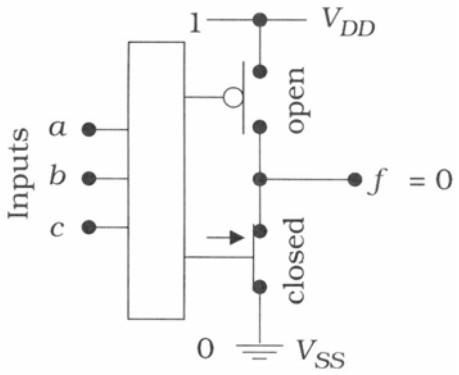


Figure 2.18 General CMOS logic gate



(a) $f = 1$ output



(b) $f = 0$ output

Figure 2.19 Operation of a CMOS logic gate

The NOT Gate (1 / 2)

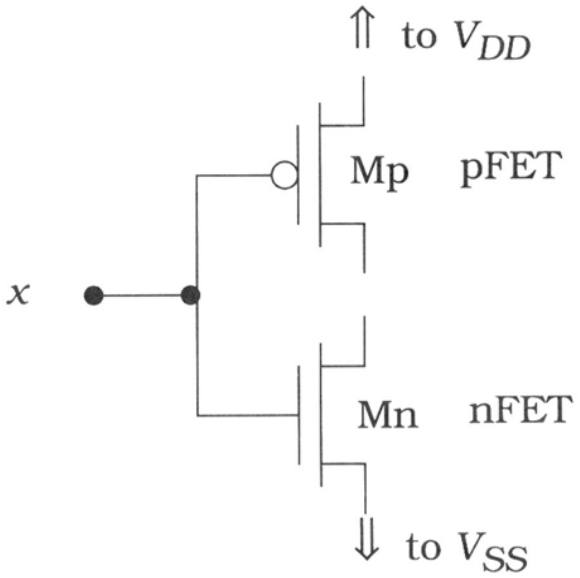
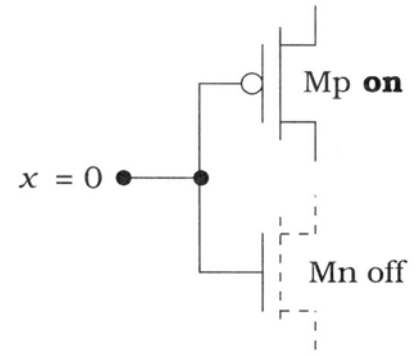
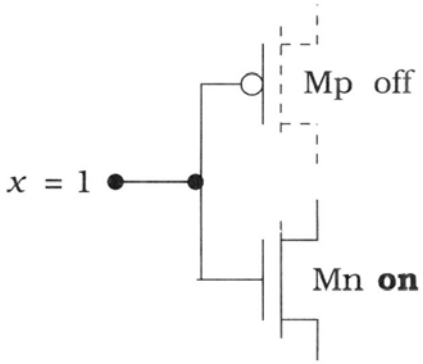


Figure 2.20 A complementary pair

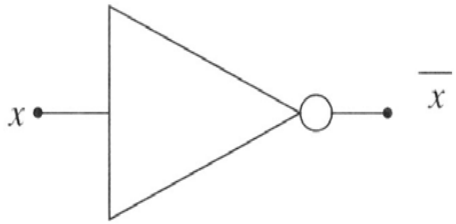


(a) x = 0 input



(b) x = 1 input

Figure 2.21 Operation of the complementary pair



(a) Logic symbol

x	$\overline{x}$
0	1
1	0

(b) Truth Table

Figure 2.22 NOT gate

The NOT Gate (2 / 2)

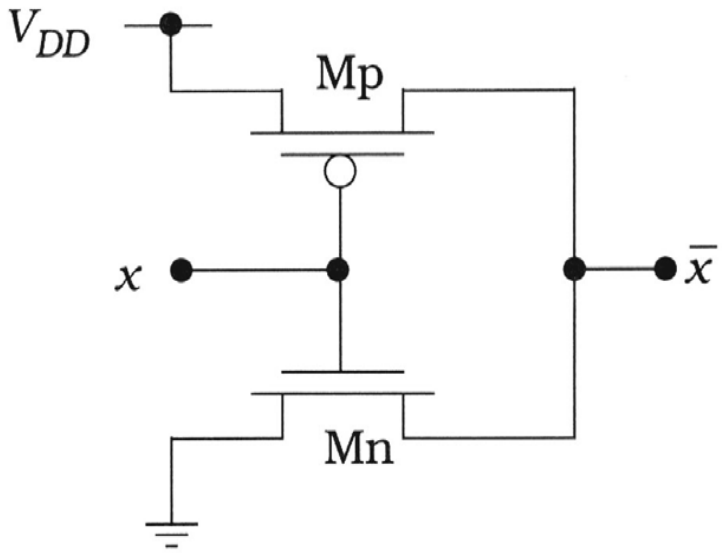
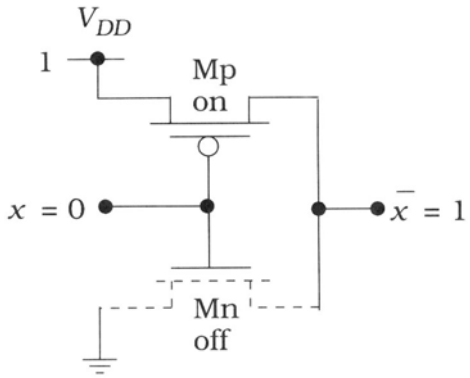
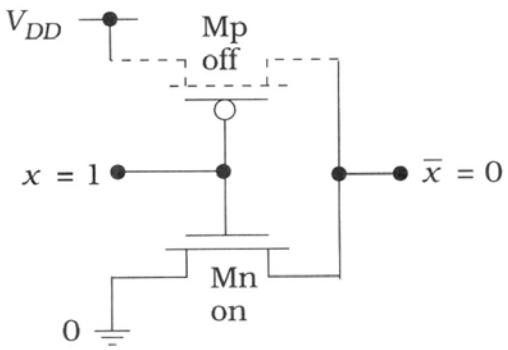


Figure 2.23 CMOS not gate



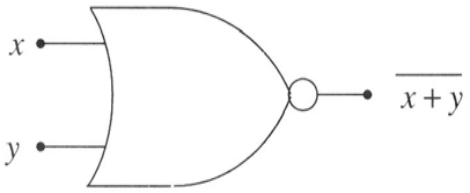
(a) x = 0 input



(b) x = 1 input

Figure 2.24 Operation of the CMOS NOT gate

The NOR Gate (1 / 2)

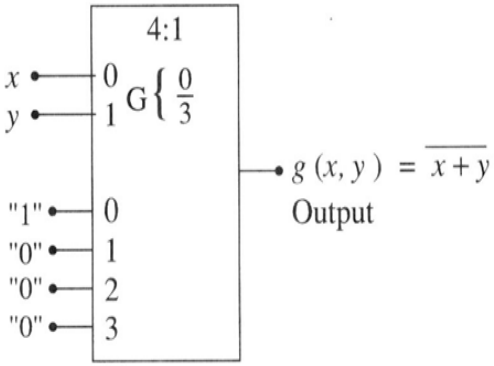


(a) Logic symbol

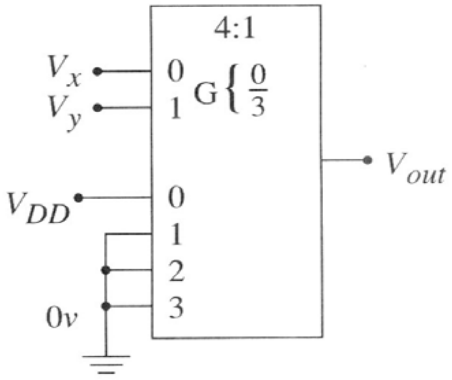
x	y	$\overline{x+y}$
0	0	1
0	1	0
1	0	0
1	1	0

(b) Truth Table

Figure 2.25 NOR logic gate



(a) Logic diagram



(b) Voltage network

Figure 2.26 NOR2
using a 4:1 multiplexor

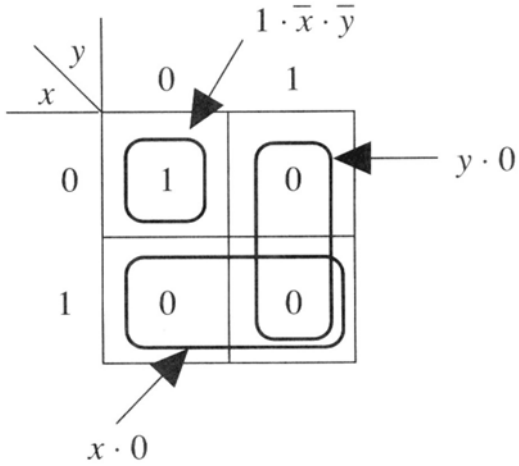


Figure 2.27 NOR2 gate
Karnaugh map

NOR (2 / 2)

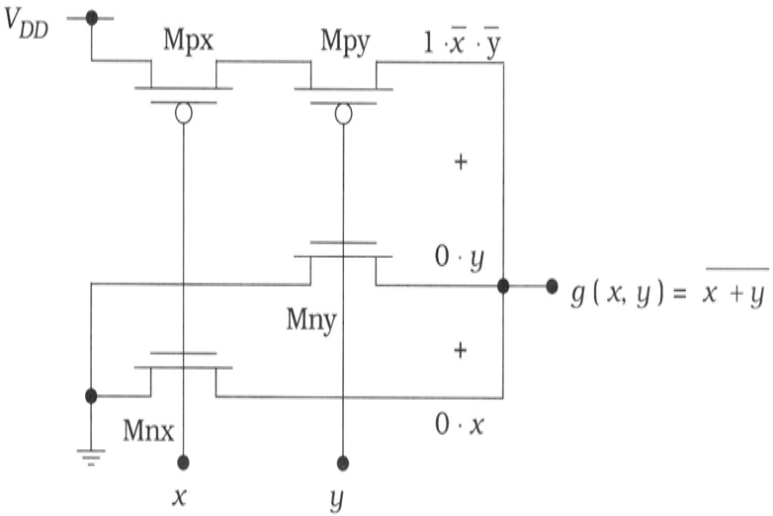
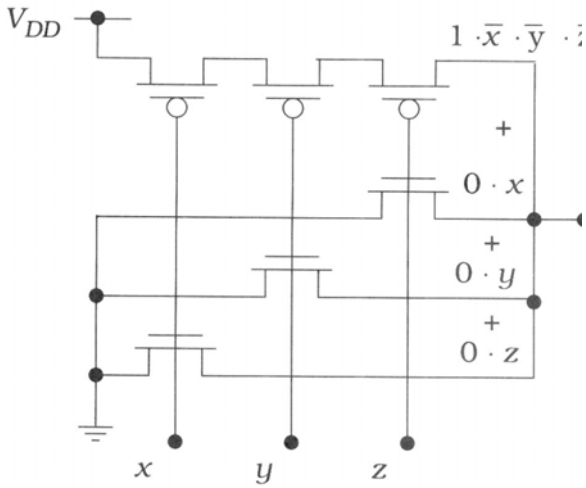


Figure 2.28 NOR2 in CMOS

x	y	Mpx	Mpy	Mnx	Mny	g
0	0	on	on	off	off	1
0	1	on	off	off	on	0
1	0	off	on	on	off	0
1	1	off	off	on	on	0

Figure 2.29 Operational summary of the NOR2 gate



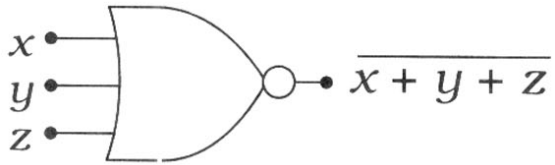
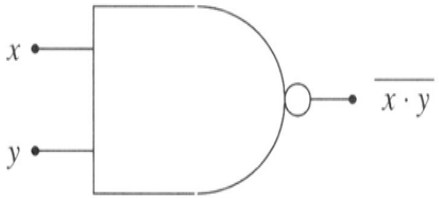


Figure 2.30 NOR3 in CMOS

NAND (1/2)

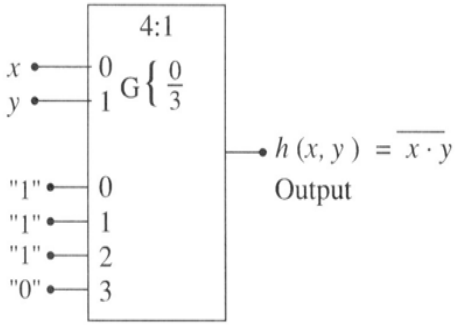


(a) Logic symbol

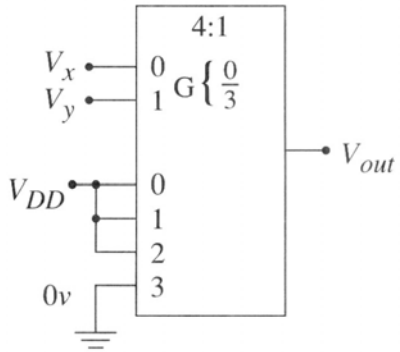
x	y	$\overline{x \cdot y}$
0	0	1
0	1	1
1	0	1
1	1	0

(b) Truth Table

Figure 2.31 NAND2 logic gate



(a) Logic diagram



(b) Voltage network

Figure 2.32 NAND2 using 4:1 multiplexor

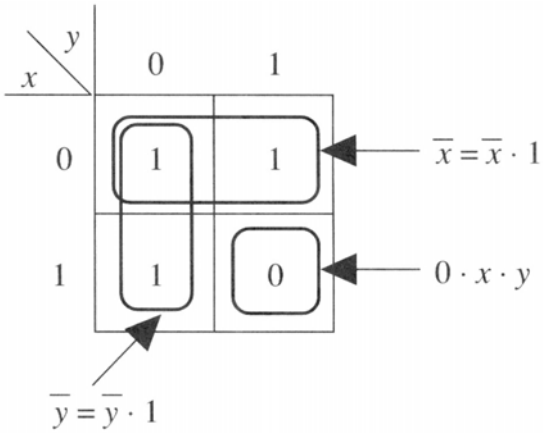


Figure 2.33 NAND2 K-map

NAND (2/2)

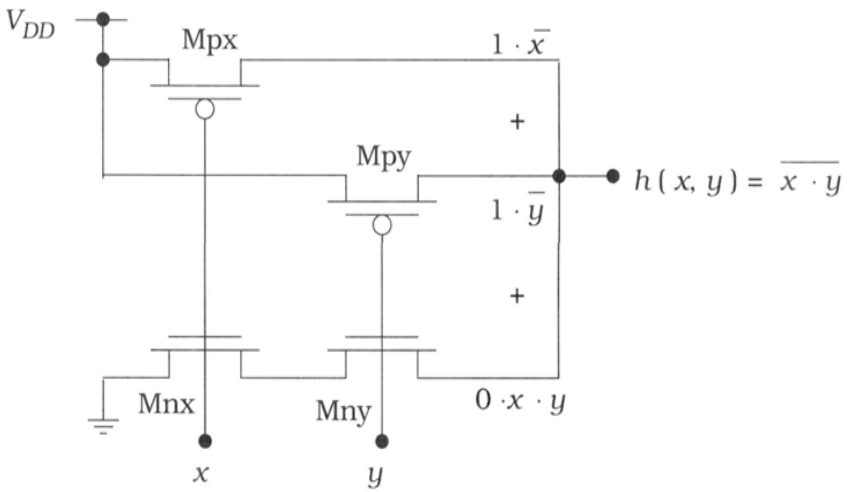
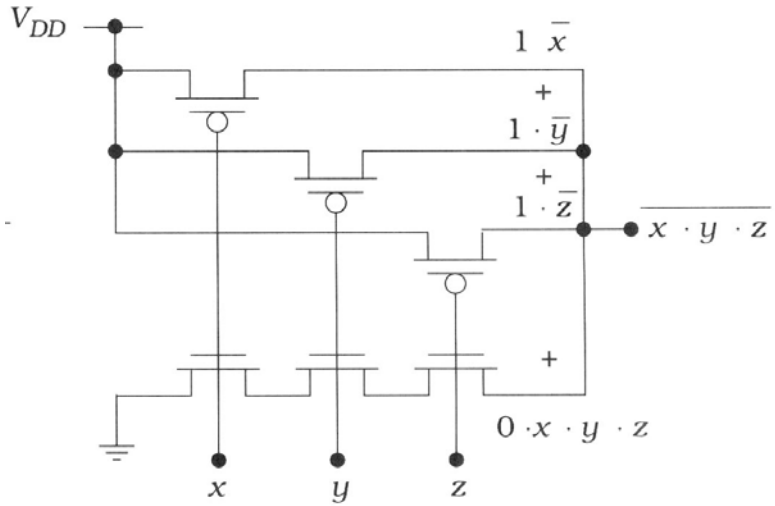


Figure 2.34 CMOS NAND2 logic circuit

x	y	Mpx	Mpy	Mnx	Mny	h
0	0	on	on	off	off	1
0	1	on	off	off	on	1
1	0	off	on	on	off	1
1	1	off	off	on	on	0

Figure 2.35 Operational summary of the NAND2 gate



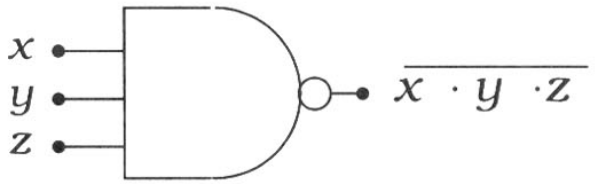


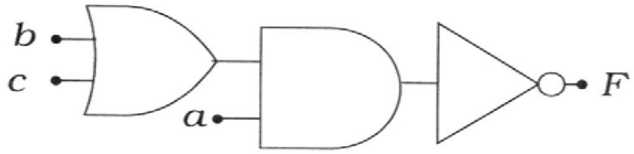
Figure 2.36 NAND3 in CMOS

Outline

- The Fundamental MOSFETs
- Ideal Switches and Boolean Operations
- MOSFETs as Switches
- Basic Logic Gates in CMOS
- Complex Logic Gates in CMOS
- Transmission Gate Circuits
- Clocking and Dataflow Control

Complex Logic Gate (1 / 3)

- Complex or combinational logic gates
  - » Useful in VLSI system-level design
  - » Consider a Boolean expression $F(a,b,c) = \overline{a \cdot (b + c)}$



$$F(a,b,c) = \overline{a \cdot (b + c)}$$
$$= \bar{a} + \overline{(b + c)}$$
$$= [\bar{a} + (\bar{b} \cdot \bar{c})] \cdot 1 \qquad (2.50)$$

  - » Expanding by simply ANDing the result with a logical 1

$$F = \bar{a} \cdot 1 + (\bar{b} \cdot \bar{c}) \cdot 1 \qquad (2.51)$$

Complex Logic Gate (2 / 3)

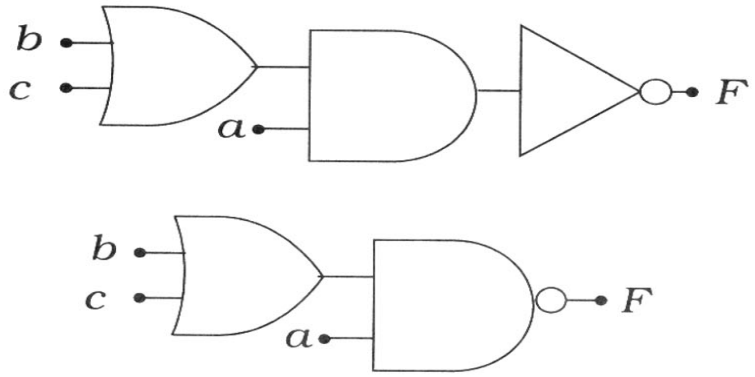


Figure 2.37 Logic function example

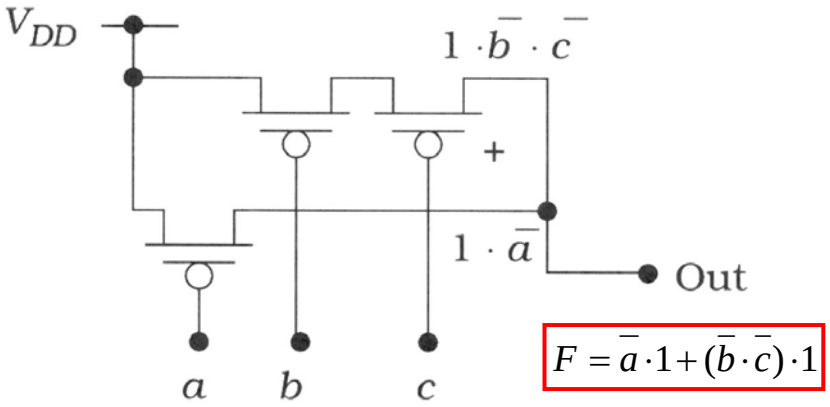


Figure 2.38 pFET circuit for *F* function from equation (2.51)

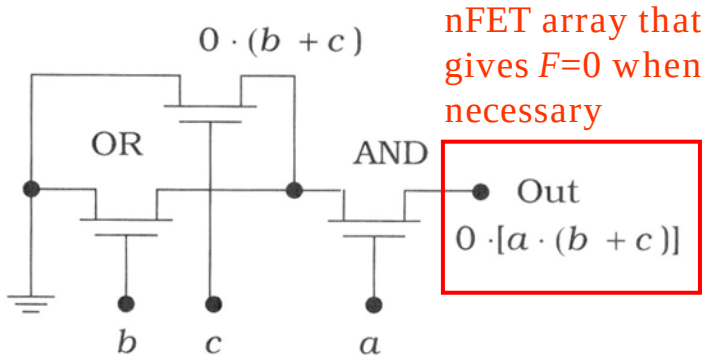


Figure 2.39 nFET circuit for F

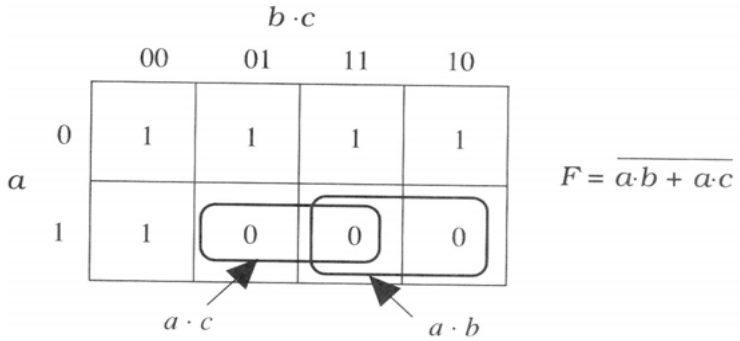


Figure 2.40 Karnaugh for nFET circuit

Complex Logic Gate (3/3)

- The characteristics of Complementary CMOS
  - 對於CMOS電路而言，由於完全對稱的結構，若是輸入電壓有0 ~ V_{DD}的全擺幅(full swing) ，則輸出訊號也一定擁有V_{DD} ~ 0 (反相) 的全擺幅。同時當輸入電壓是0或是V_{DD}時，電路也沒有直流功率消耗。
  - 由於輸入及輸出訊號均有全擺幅，任何閘級均能任意銜接，不需考慮電壓準位問題。
  - 也由於輸入及輸出訊號均有全擺幅，製程變動也不會影響此特性。所以縱然製造後製程特性可能與設計時稍有不同，對於一般的CMOS電路，或許會影響其速度或功率消耗等電性的表現，卻不會影響其應有的功能(function)。此點對於所謂的超大型積體電路而言是一項大利多，因為它為大型電路之量產(mass production)特性提供了一個絕佳的支持－即可靠度(reliability)。
  - 更由於上述特性，一般在設計時均會考慮環境因素的變動範圍(variation corners)，讓設計除了滿足功能要求外，也讓電性的表現甚至比規格 (specifications)還嚴一點，亦即留有寬限(margin)。因此製造後，非但功能一定可以達成，電性也能滿足規格。

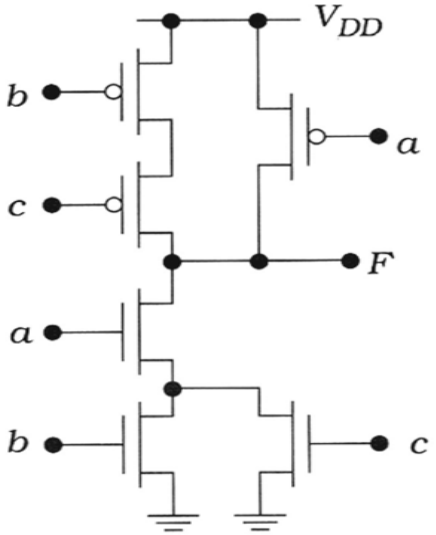


Figure 2.41 Finished complex CMOS logic gate circuit

Structured Logic Design (1 / 4)

- CMOS logic gates are intrinsically inverting
  - » Output always produces a *NOT operation* acting on the input variables

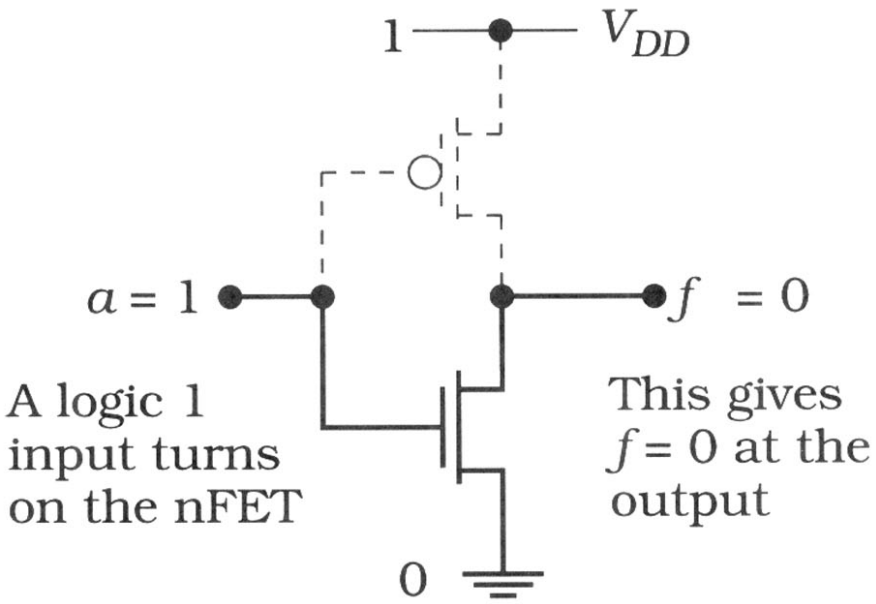
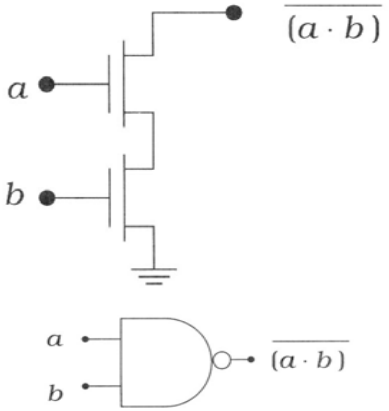
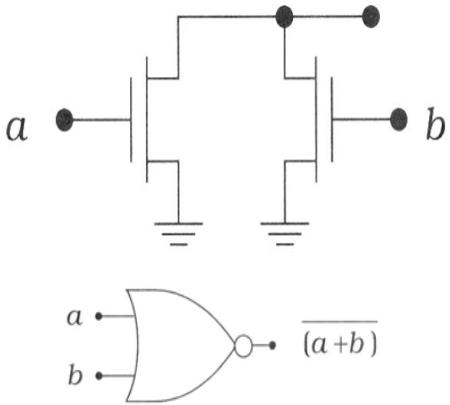


Figure 2.42 Origin of the inverting characteristic of CMOS gates

Structured Logic Design (2 / 4)



(a) Series-connected nFETs



(b) Parallel-connected nFETs

Figure 2.43 nFET logic formation

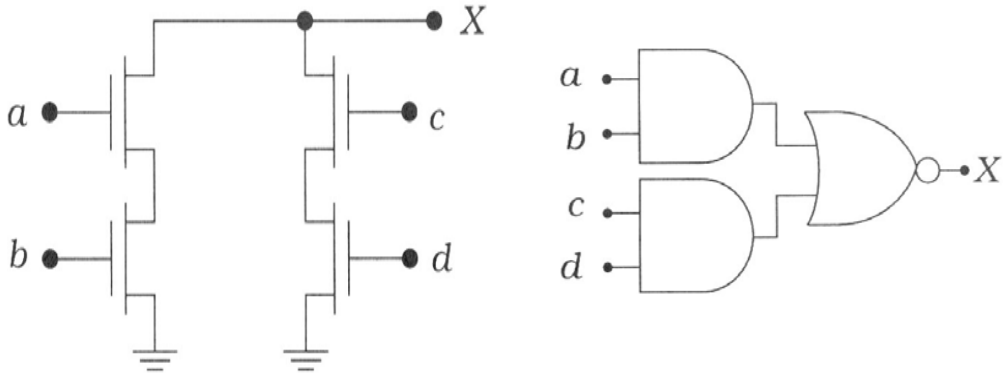


Figure 2.44 nFET AOI circuit

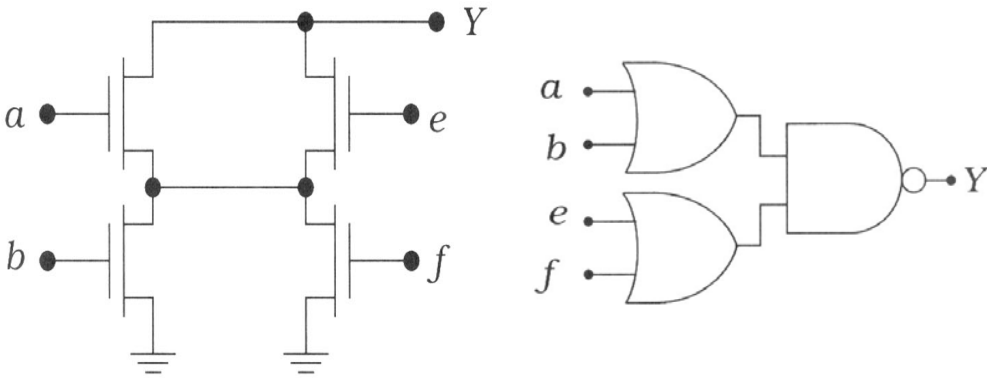
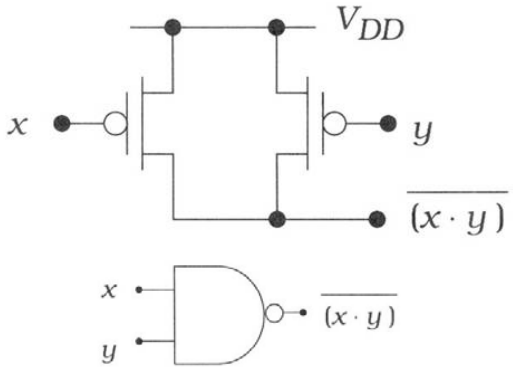
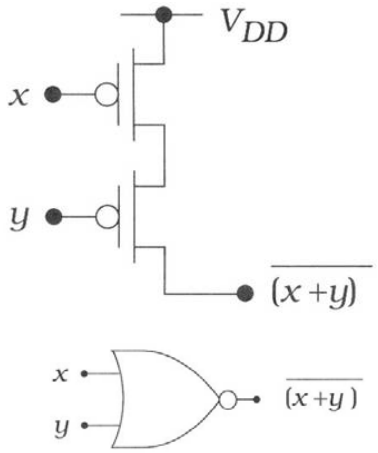


Figure 2.45 nFET OAI circuit

Structured Logic Design (3 / 4)

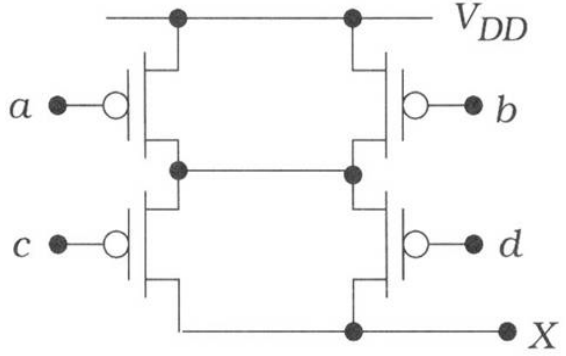


(a) Parallel-connected pFETs

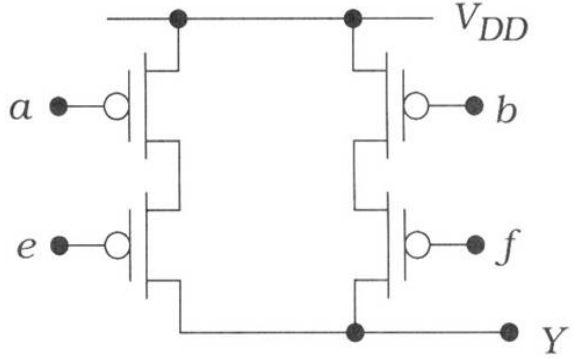


(b) Series-connected pFETs

Figure 2.46 pFET logic formation



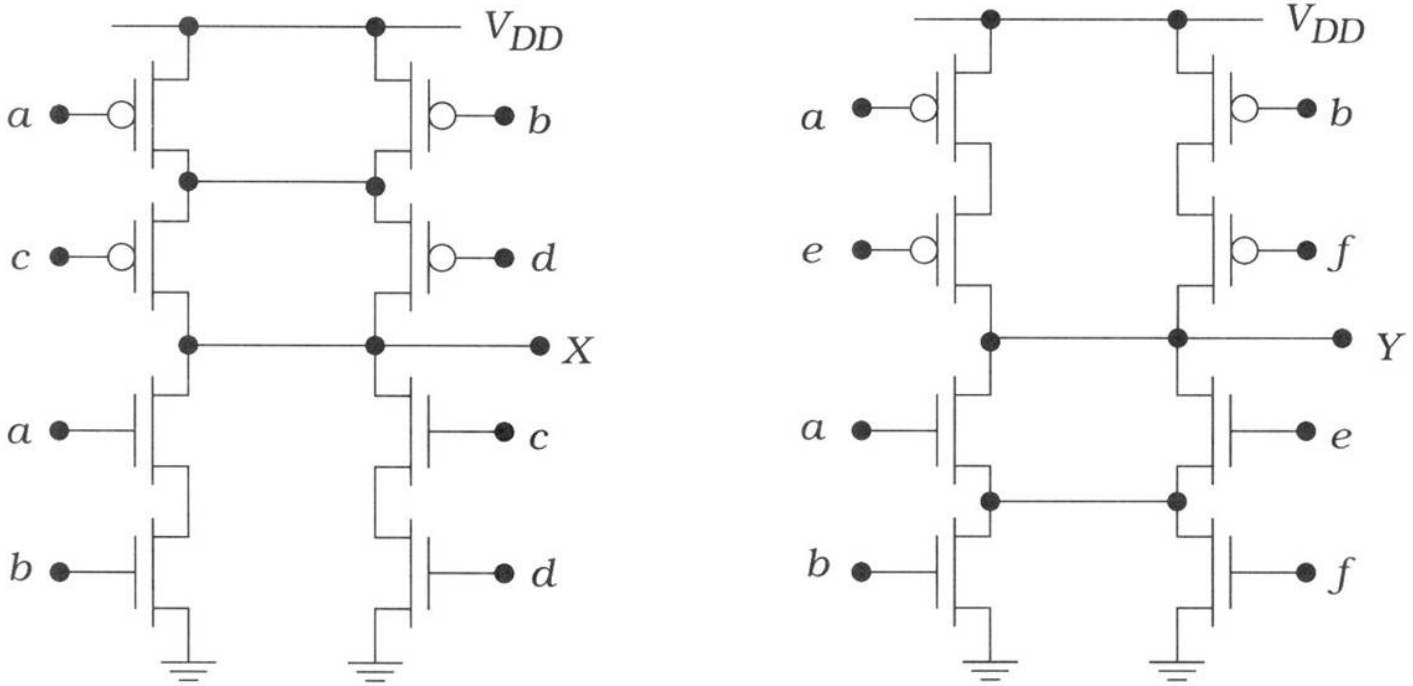
(a) pFET AOI circuit



(b) pFET OAI circuit

Figure 2.47 pFET arrays for AOI and OAI gates

Structured Logic Design (4/4)

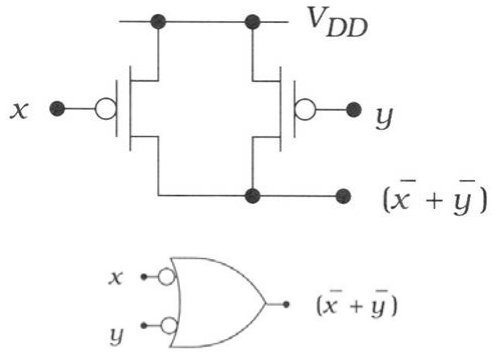


(a) AOI circuit

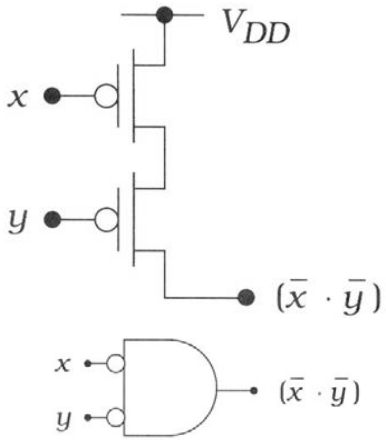
(b) OAI circuit

Figure 2.48 Complete CMOS AOI and OAI circuits

Bubble Pushing

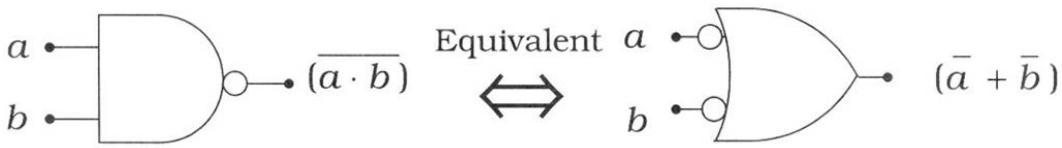


(a) Parallel-connected pFETs



(b) Series-connected pFETs

Figure 2.51 Assert-low models for pFETs



(a) NAND - OR

(b) NOR - AND

Figure 2.52 Bubble pushing using DeMorgan rules

XOR and XNOR Gates

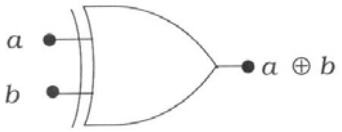
- An important example of using an AOI circuit is constructing Exclusive-OR (XOR) and Exclusive-NOR circuits

$$a \oplus b = \bar{a} \cdot b + a \cdot \bar{b} \quad (2.71)$$

$$\overline{a \oplus b} = a \cdot b + \bar{a} \cdot \bar{b} \quad (2.72)$$

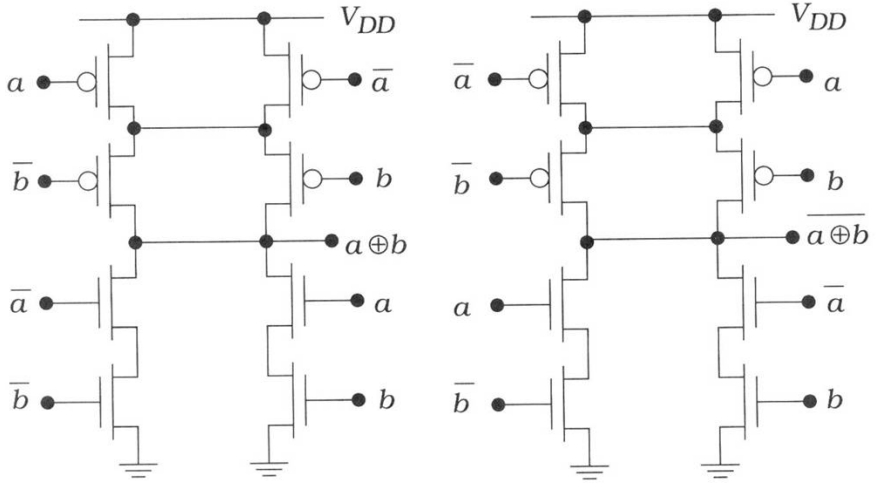
$$\Rightarrow a \oplus b = \overline{(\overline{a \oplus b})} = \overline{a \cdot b + \bar{a} \cdot \bar{b}} \quad (2.73)$$

$$\Rightarrow \overline{a \oplus b} = \overline{\bar{a} \cdot b + a \cdot \bar{b}} \quad (2.74)$$



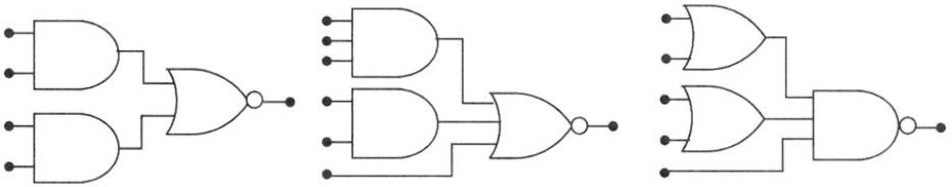
a	b	a ⊕ b
0	0	0
0	1	1
1	0	1
1	1	0

Figure 2.56 XOR



(a) Exclusive-OR (b) Exclusive-NOR

Figure 2.57 AOI XOR and XNOR gates



(a) AOI22 (b) AOI321 (c) AOI221

Figure 2.58 General naming convention

Outline

- The Fundamental MOSFETs
- Ideal Switches and Boolean Operations
- MOSFETs as Switches
- Basic Logic Gates in CMOS
- Complex Logic Gates in CMOS
- Transmission Gate Circuits
- Clocking and Dataflow Control

Transmission Gate Circuits

- A CMOS TG is created by connecting an nFET and pFET in parallel
  - Bi-directional
  - Transmit the entire voltage range [0, V_{DD}]

$$y = x \cdot s \quad iff \quad s = 1 \qquad (2.78)$$

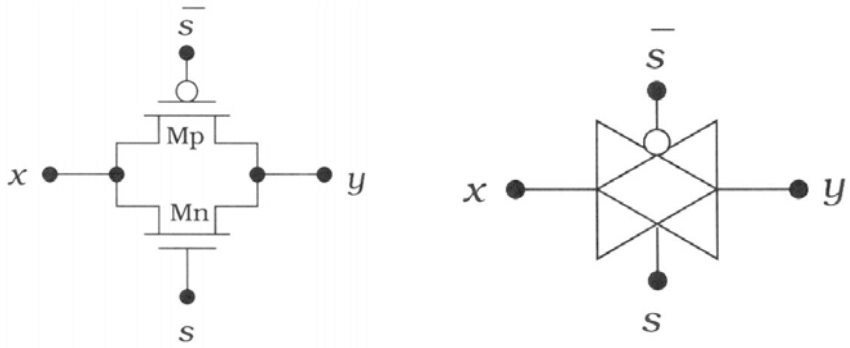
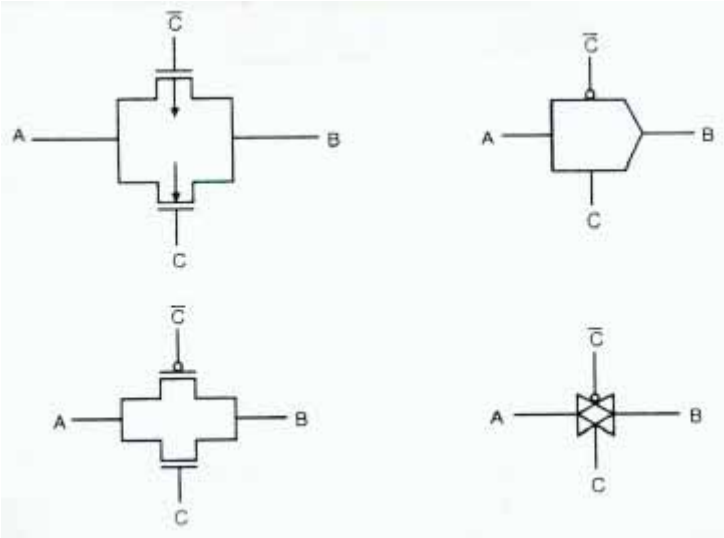


Figure 2.60 Transmission gate (TG)

Analysis of CMOS TG (1 / 4)

- Four representations of CMOS Transmission Gate (TG)



A : *Input*

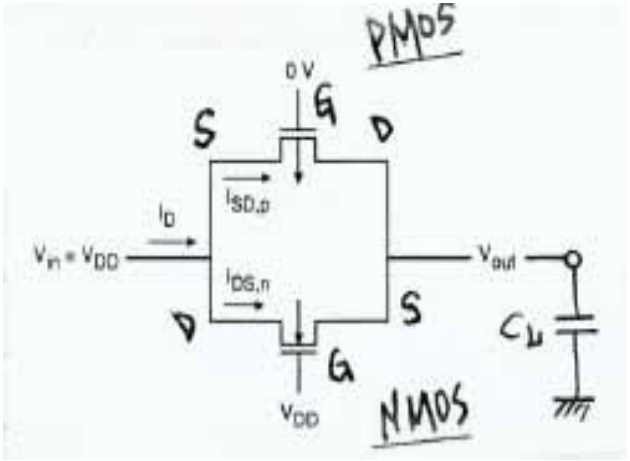
B : *Output*

C : *Control Signal*

$$C = \begin{cases} 0, Z\ (high\ impedance) \\ 1, B = A \end{cases}$$

Analysis of CMOS TG (2 / 4)

- Case (A) $V_{in}=V_{dd}$, $C=V_{dd}$ (Both transistors ON)



NMOS:

$$\begin{cases} V_{ds,n} = V_{DD} - V_{out} \\ V_{gs,n} = V_{DD} - V_{out} \end{cases}$$

NMOS operation:

$$\begin{cases} 1.\ Trun\ off\ , V_{out} > V_{DD} - V_{t,n} \\ 2.\ Saturation\ , V_{out} < V_{DD} - V_{t,n} \end{cases}$$

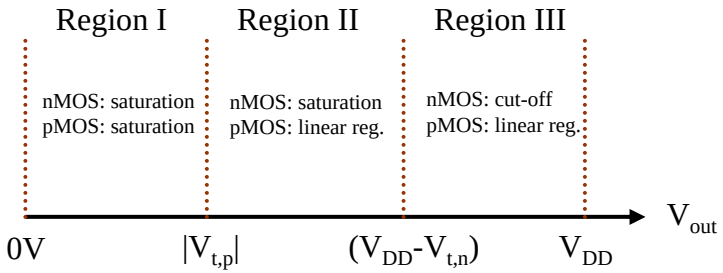
PMOS:

$$\begin{cases} V_{ds,p} = V_{out} - V_{DD} \\ V_{gs,p} = -V_{DD} \end{cases}$$

PMOS operation:

$$\begin{cases} 1.\ Saturation\ , V_{out} < |V_{t,p}| \\ 2.\ Linear\ , V_{out} > |V_{t,p}| \end{cases}$$

➔ *PMOS is always ON regardless of V_{out} Value*



Summary of operating regions of MOS

- Total Current from *I/P* to *O/P*: $I_D = I_{DS,n}+I_{SD,p}$
- Equivalent resistance of NMOS and PMOS

$$reg,_n = \frac{V_{DD}-V_{out}}{I_{ds,n}},\ reg,_p = \frac{V_{DD}-V_{out}}{I_{sd,p}}$$

➔ Equivalent R of TG = $reg,_n$ // $reg,_p$

Analysis of CMOS TG (3 / 4)

- Region (I): $V_{out} < |V_{t,p}|$ { NMOS: saturation; PMOS: saturation

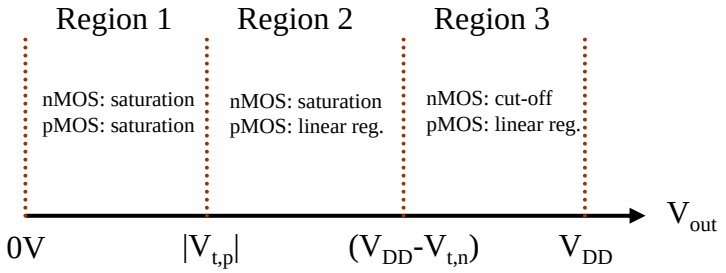
$$reg,_n = \frac{2(V_{DD} - V_{out})}{\beta_n (V_{DD} - V_{out} - V_{t,n})^2}$$

$$reg,_p = \frac{2(V_{DD} - V_{out})}{\beta_p (V_{DD} - |V_{t,p}|)^2}$$

- Region (II): $|V_{t,p}| < V_{out} < (V_{DD}-V_{t,n})$ { NMOS: saturation; PMOS: linear reg.

$$reg,_n = \frac{2(V_{DD} - V_{out})}{\beta_n (V_{DD} - V_{out} - V_{t,n})^2}$$

$$reg,_p = \frac{2(V_{DD} - V_{out})}{\beta_p [\underbrace{2(V_{DD} - |V_{t,p}|)}_{(V_{gs}-V_t)} \underbrace{(V_{DD} - V_{out})}_{V_{ds}} - \underbrace{(V_{DD} - V_{out})^2}_{(V_{ds})^2}]}$$



Note:

① NMOS source-to-substrate voltage
$= V_{SB,n} = V_{out} - 0 = V_{out}$
(*Body Effect*)

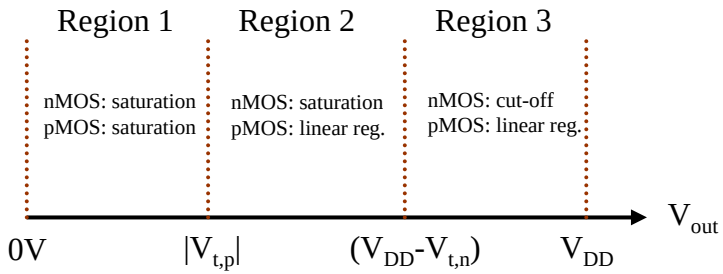
② PMOS source-to-substrate voltage
$= V_{SB,p} = 0 - 0 = 0$ (constant)

Analysis of CMOS TG (4 / 4)

- Region (III): $V_{out} > (VDD-|V_{t,p}|)$ { NMOS: cut-off; PMOS: linear reg.

$$reg,_n = \infty, \; open$$

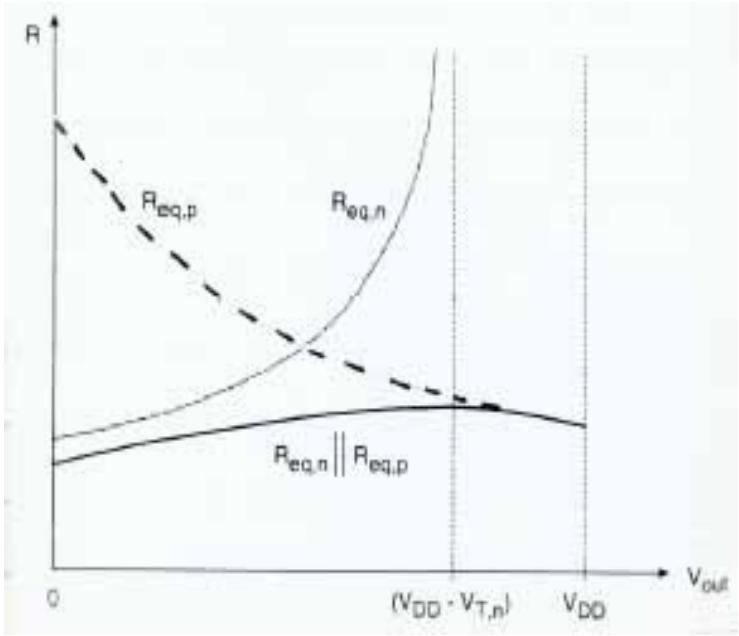
$$reg,_p = \frac{2}{\beta_p[2(V_{DD} - |V_{t,p}|) - (V_{DD} - V_{out})]} \quad (simplify)$$



- Total resistance of CMOS TG v.s. V_{out}

① Equivalent resistance of TG is relatively constant

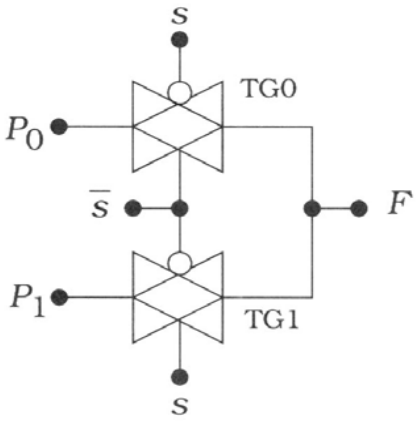
② Individual reg. of NMOS and PMOS are strongly dependent on V_{out}!!



Logic Design using TG (1 / 3)

- Multiplexors
  - » TG based 2-to-1 multiplexor

$$F = P_0 \cdot \overline{s} + P_1 \cdot s \qquad (2.79)$$



s	TG0	TG1	F
0	Closed	Open	P_0
1	Open	Closed	P_1

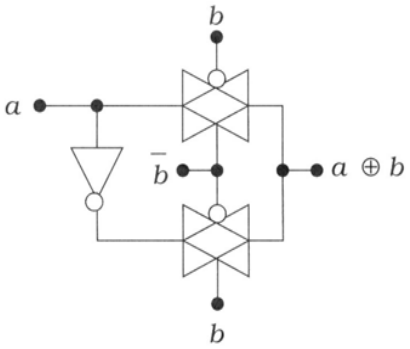
Figure 2.61 A TG-based 2-to-1 multiplexor

  - » The 2-to-1 extended to a 4:1 network by using the 2-bit selector word (s_1, s_o)

$$F = P_0 \cdot \overline{s_1} \cdot \overline{s_0} + P_1 \cdot \overline{s_1} \cdot s_0 + P_2 \cdot s_1 \cdot \overline{s_0} + P_3 \cdot s_1 \cdot s_0 \qquad (2.80)$$

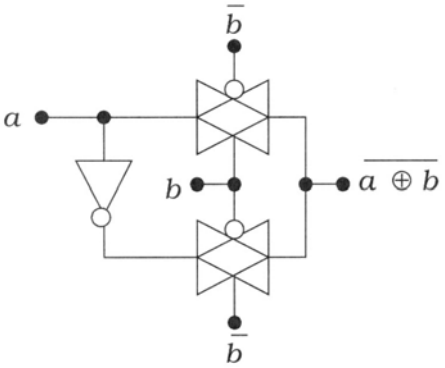
Logic Design using TG (2 / 3)

- TG based XOR/XNOR



$$a \cdot \bar{b} + \bar{a} \cdot b = a \oplus b \qquad (2.81)$$

(a) XOR circuit

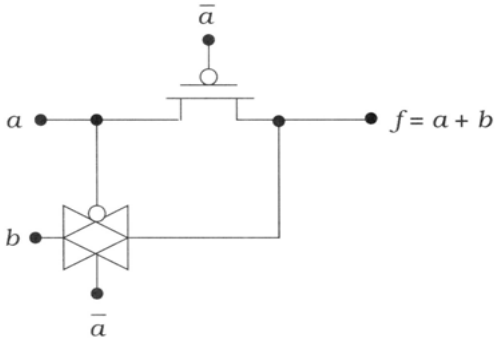


$$\overline{a \oplus b} = a \cdot b + \bar{a} \cdot \bar{b} = a \otimes b \qquad (2.82)$$

(b) XNOR circuit

Figure 2.62 TG-based exclusive-OR and exclusive-NOR circuits

- TG based OR gate



$$\begin{aligned} f &= a \cdot (\bar{\bar{a}}) + \bar{a} \cdot b \\ &= a + \bar{a} \cdot b \\ &= a + b \end{aligned} \qquad (2.83)$$

Figure 2.63 A TG-based OR gate

Logic Design using TG (3/3)

- Alternate XOR/XNOR Circuits
  - » Mixing TGs and FETs which are designed for exclusive-OR and equivalence (XNOR) functions
  - » It's important in adders and error detection/correction algorithms

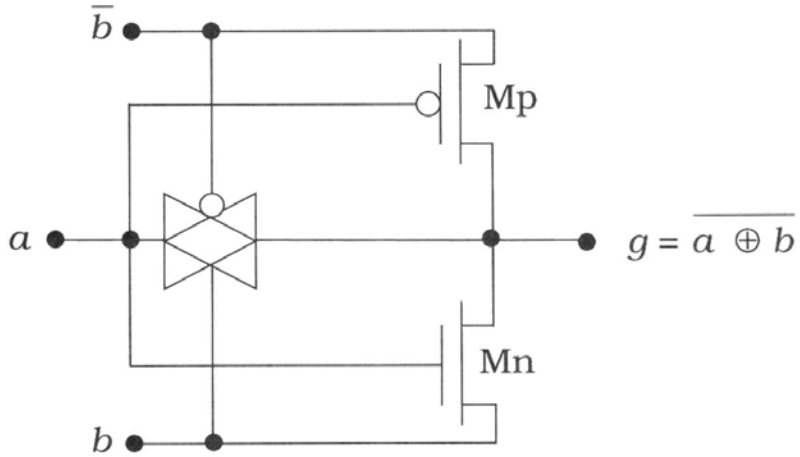


Figure 2.64 An XNOR gate that used both TGs and FETs

Outline

- The Fundamental MOSFETs
- Ideal Switches and Boolean Operations
- MOSFETs as Switches
- Basic Logic Gates in CMOS
- Complex Logic Gates in CMOS
- Transmission Gate Circuits
- Clocking and Dataflow Control

Clock and Dataflow Control

- *Synchronous* digital design using a clock signal
  - Simply, the switching characteristics of TGs

$$f = \frac{1}{T} \qquad (2.84)$$

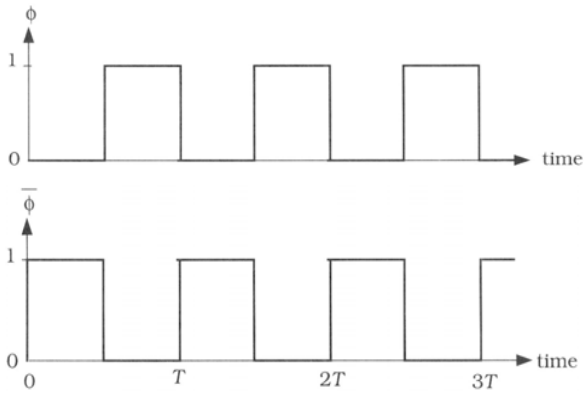
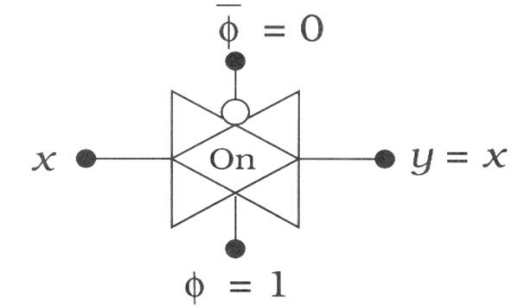


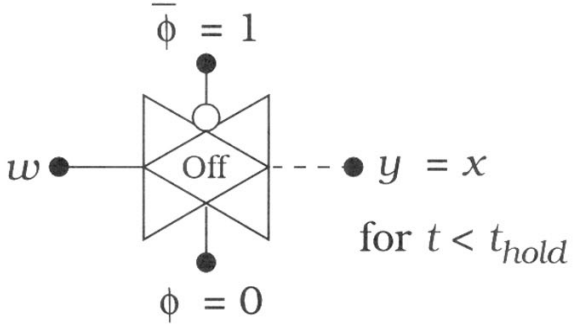
Figure 2.65 Complementary clocking signals

  - As Figure 2.66(b), when TG is off, the value of $y = x$ for a very short time t_{hold}. If we use a high-frequency clock then the periodic open-closed change occurs at every half clock cycle

$$\Rightarrow (T/2) < t_{hold}$$



(a) Closed switch



(b) Open switch

Figure 2.66 Behavior of a clocked TG

Clock and Dataflow Control Using TGs

- *Data Synchronization* using transmission gates
  - To use clocked TGs for data flow control, we place oppositely phased TGs at the inputs and outputs of logic blocks

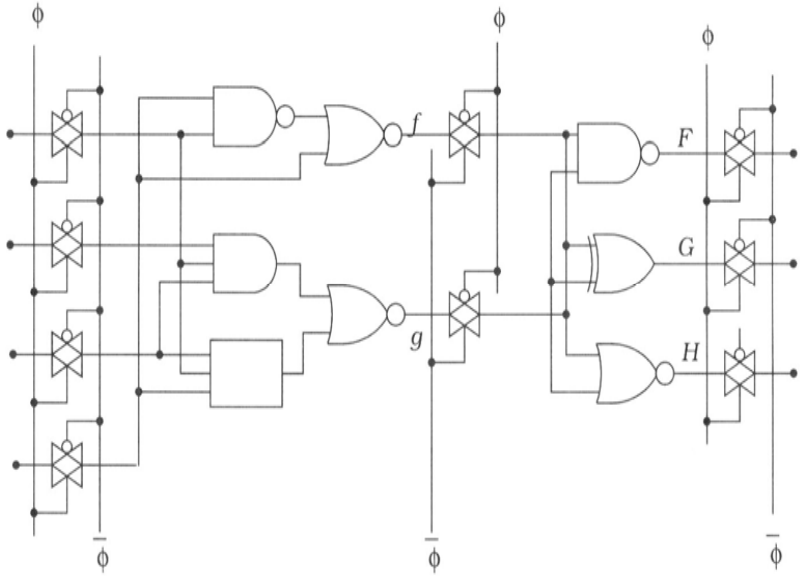


Figure 2.67 Data synchronization using transmission gates

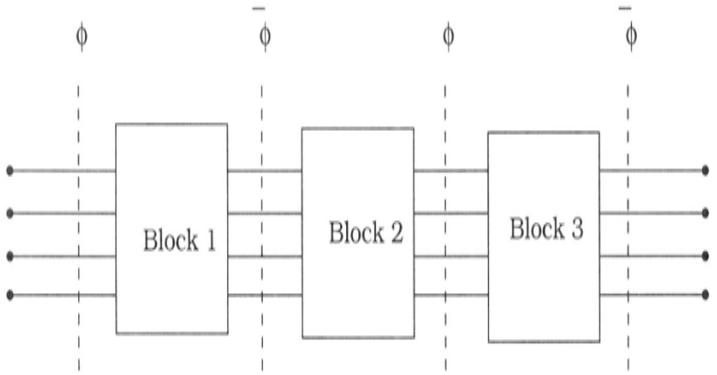
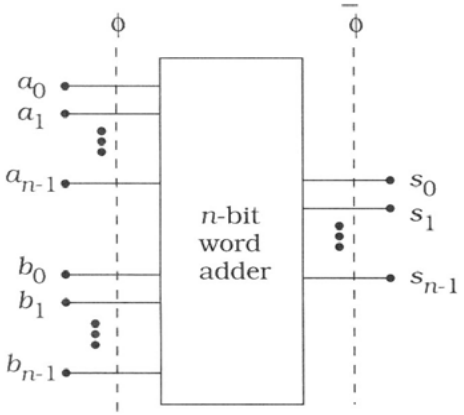


Figure 2.68 Block-level system timing diagram

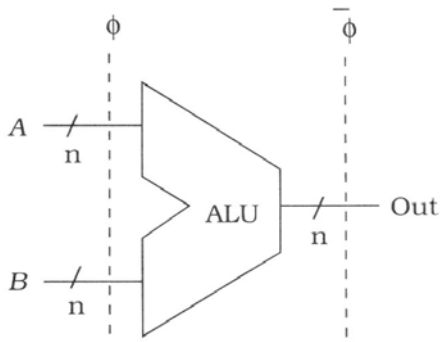
- In this scheme, data moves through a logic block every half cycle
- Since the logic blocks are arbitrary, it can be used as the basis for building very complex logic chains
- Synchronize the operations performed on each bit of an n-bit binary word

A Synchronized Word Adder

- In figure 2.69(a), the input word $a_{n-1}\ldots a_0$ and $b_{n-1}\ldots b_0$ are controlled by the $\phi - clock$ plane, while the sum $s_{n-1}\ldots s_0$ is transferred to the output when $\phi = 0$
  - » Every bit in a word is transmitted from one point to another at the same time, which allows us to track the data flow through the system
- In figure 2.69(b), a larger scale with the ALU (arithmetic and logic unit)
  - » Input A and B are "gated" into the ALU by the control signal $\phi - plane$
  - » The result word Out is transferred to the next stage when $\bar{\phi} = 1$, i.e., $\phi = 0$



(a) Clocked adder

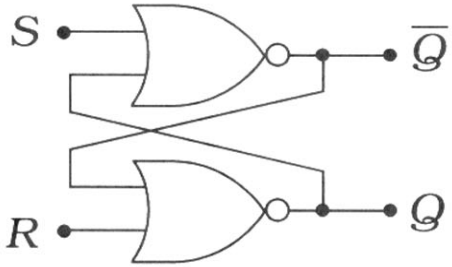


(b) Clocked ALU

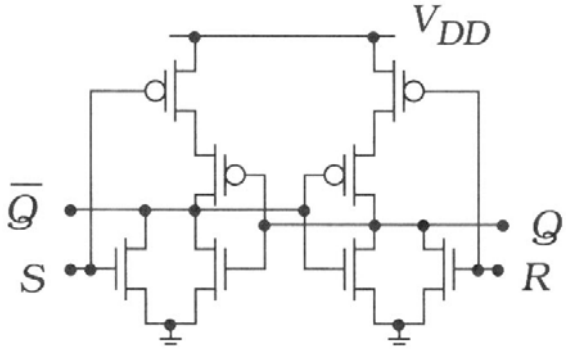
Figure 2.69 Control of binary words using clocking planes

Clock and Dataflow Control

- Clocked transmission gates synchronize the flow of signals, but the line themselves cannot store the values for times longer than t_{hold}

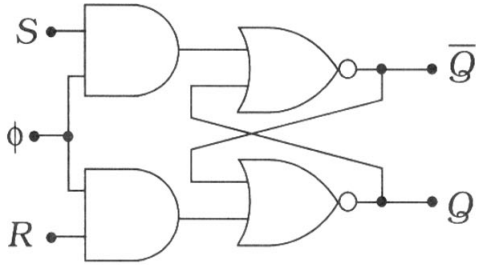


(a) Logic diagram

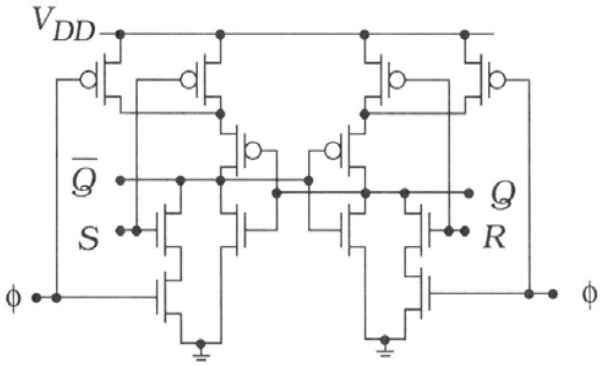


(b) CMOS circuit

Figure 2.70 SR latch



(a) Logic diagram



(b) CMOS circuit

Figure 2.71 Clocked SR latch