

LSI概論(3)

論理回路と論理ゲート

2進数と論理回路

- コンピュータ内部の演算は“1”と“0”の2値で行われる(2進数が基本)
 - 電源電圧(Vdd)になっている場合“1”
 - GND電位(0V)になっている場合“0”
- 内部計算の例: 2進数の加算(正の数)
 - $0 + 0 = 0$
 - $0 + 1 = 1$
 - $1 + 0 = 1$
 - $1 + 1 = 10$ (桁上げが発生)
- 上記演算をLSI中で実行するため論理回路を使用する
 - OR, AND, NOT回路 etc.
- 負の数, 浮動小数点など数の一般表現は, 後で講義

否定(NOT)

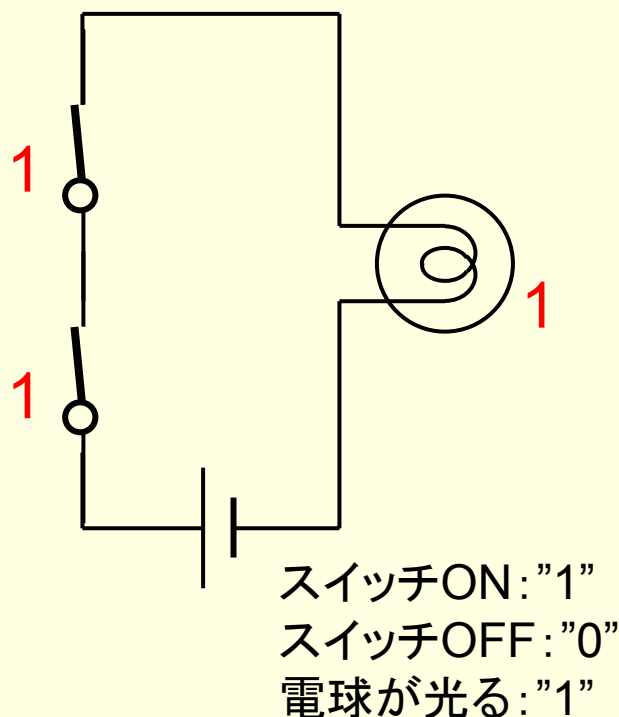
- インバータで実現される論理：否定 $Z = \overline{A}$
 - 入力が”1”の時,出力Zは”0”
 - 入力が”0”の時,出力Zは”1”
 - F,A,Bは”1”と”0”の値しかとらない論理変数

A	$F = \overline{A}$
0	1
1	0

論理積(AND)

■ $F = A \cdot B$ または $A \times B$ または $A \cap B$

- 入力A,Bの両方が"1"の時,出力Zは"1"
- F,A,Bは"1"と"0"の値しかとらない論理変数

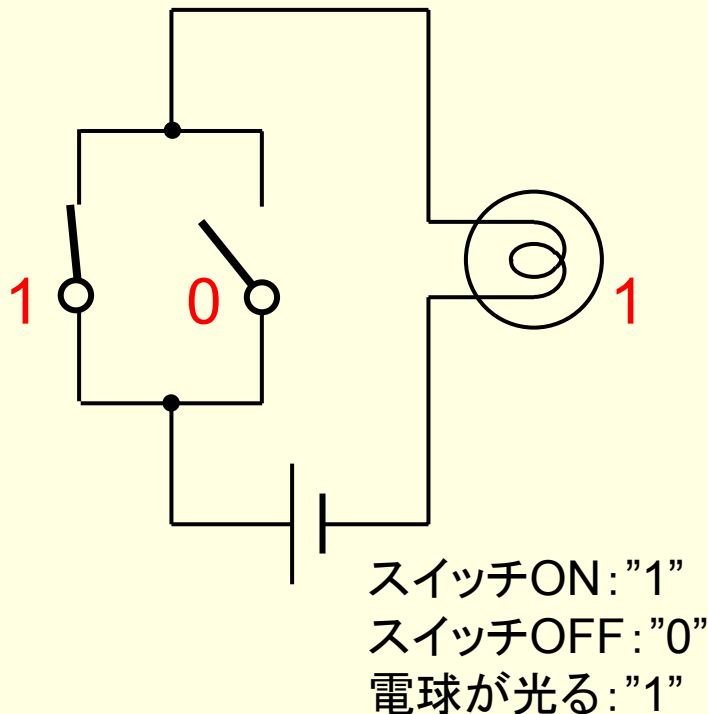


A	B	$F = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

論理和(OR)

■ $F=A+B$ または $A \cup B$

- 入力A,Bのどちらかが"1"の時,出力Zは"1"
- F,A,Bは"1"と"0"の値しかとらない論理変数

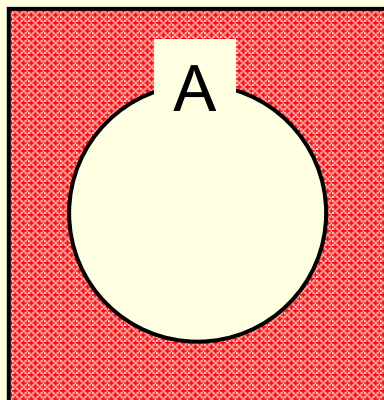


A	B	$F=A+B$
0	0	0
0	1	1
1	0	1
1	1	1

ベン図

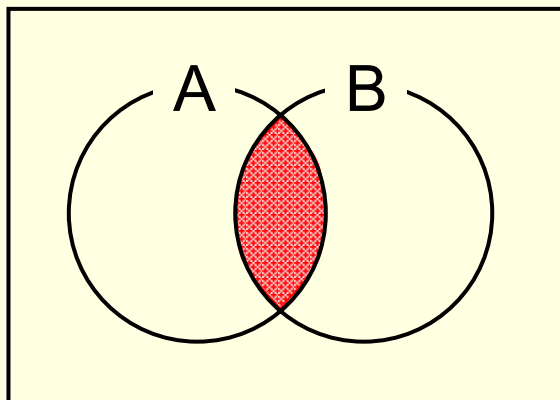
- NOT, AND, ORを用いてすべての論理を表すことができる。(後述)
- 論理をわかりやすくするためベン図が使用される

NOT



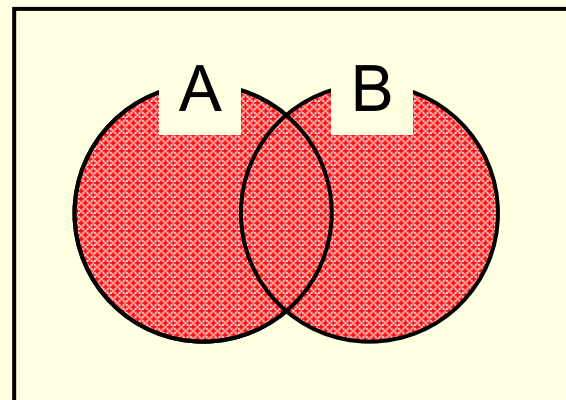
$$F = \overline{A}$$

AND



$$F = A \cdot B$$

OR



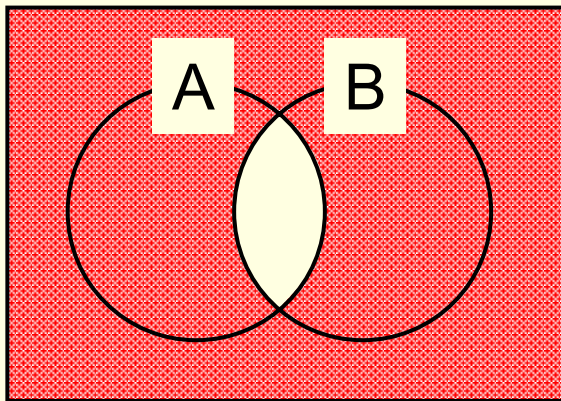
$$F = A + B$$

NANDとNOR

■ NAND: ANDの否定

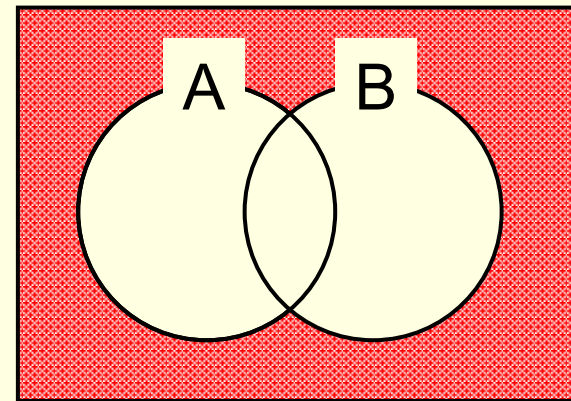
■ NOR: ORの否定

NAND



$$\begin{cases} F = \overline{A \cdot B} \\ F = \overline{A} + \overline{B} \end{cases}$$

NOR



$$\begin{cases} F = \overline{A + B} \\ F = \overline{A} \cdot \overline{B} \end{cases}$$

■ ド・モルガンの法則

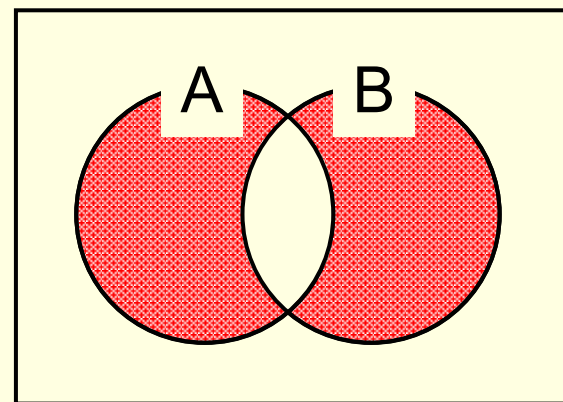
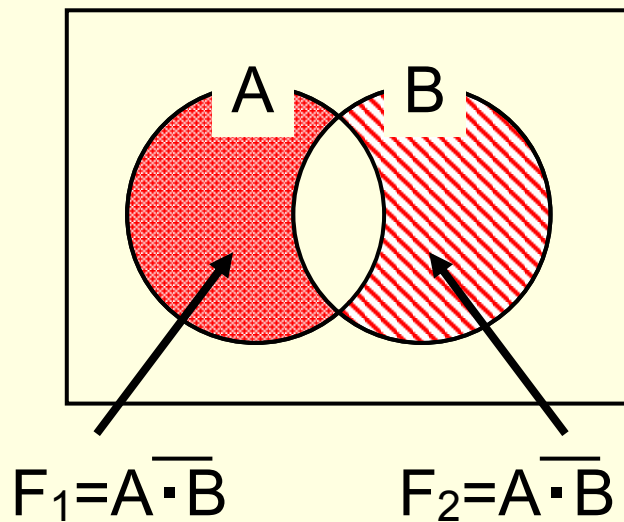
重要

$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

排他的OR (XOR: exclusive OR)

- A,Bいずれかの入力のうち一方が1のとき出力1



$$F = A \cdot \overline{B} + \overline{A} \cdot B$$

$Z = A \oplus B$
と書くこともある

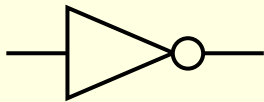
A	B	$F = A \cdot B$
0	0	0
0	1	1
1	0	1
1	1	0

* 3入力以上の場合, 奇数個の変数が1のときに1になる関数.

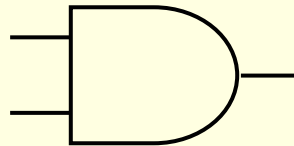
MIL記号

- 論理回路図中では以下のMIL(military standard)記号が使用される。

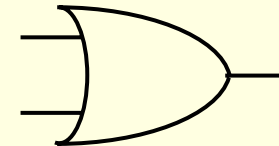
NOT



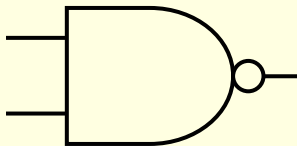
AND



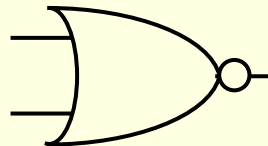
OR



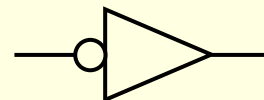
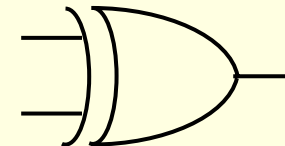
NAND



NOR



XOR

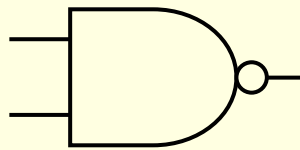


NOTの別表現

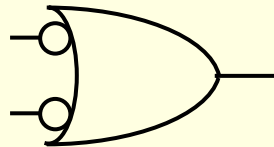
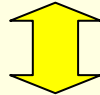
MIL記号を用いたド・モルガンの法則

- 論理回路図で下記の変換はよく使用する

NAND



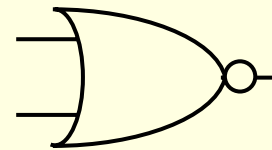
$\overline{A \cdot B}$



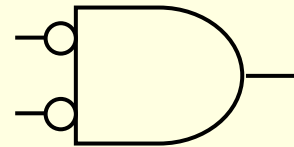
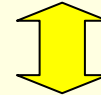
$\overline{A} + \overline{B}$

$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

NOR



$\overline{A + B}$

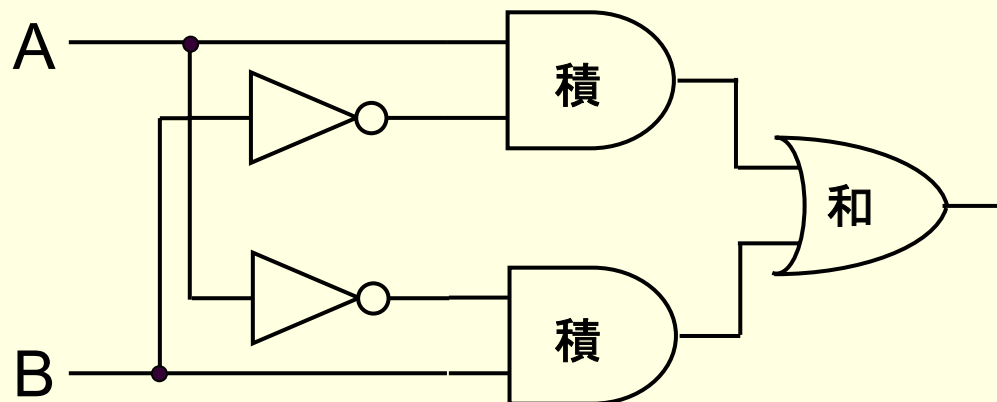


$\overline{A} \cdot \overline{B}$

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

論理関数の完全系(1)

- 任意の論理式はOR,AND,NOTの組み合わせで表現できる
- 論理は以下の積和形, 和積形の2種類の両表現ある
 - 積和形: $F = A_1 \cdot B_1 \cdot C_1 + A_2 \cdot B_2 \cdot C_2 + A_3 \cdot B_3 \cdot C_3$
 - 和積形: $F = (A_1 + B_1 + C_1) \cdot (A_2 + B_2 + C_2) \cdot (A_3 + B_3 + C_3)$
- たとえば積和型であらわしたXORは以下のとおり
 - $F = A \cdot \overline{B} + \overline{A} \cdot B$
 - MIL記号では以下のとおり



論理関数の完全系(2)

■ 積和型論理(加法標準形)の作り方

- 入力すべての組み合わせに対して, $f_i=1$ or 0 を定義する
- $F=m_1f_1+m_2f_2+m_3f_3+m_4f_4+m_5f_5+m_6f_6+m_7f_7+m_8f_8$

● 例: AND

$$f_1 \sim f_7 = 0, f_8 = 1 \\ \Rightarrow F = A \cdot B \cdot C$$

A	B	C	m_i	F_i	
0	0	0	$\overline{A} \cdot \overline{B} \cdot \overline{C}$	f_1	0
0	0	1	$\overline{A} \cdot \overline{B} \cdot C$	f_2	0
0	1	0	$\overline{A} \cdot B \cdot \overline{C}$	f_3	0
0	1	1	$\overline{A} \cdot B \cdot C$	f_4	0
1	0	0	$A \cdot \overline{B} \cdot \overline{C}$	f_5	0
1	0	1	$A \cdot \overline{B} \cdot C$	f_6	0
1	1	0	$A \cdot B \cdot \overline{C}$	f_7	0
1	1	1	$A \cdot B \cdot C$	f_8	1

論理関数の完全系(3)

■ 和積型論理(乗法標準形)の作り方

- 反転論理 $\overline{F}$ に対して $\overline{f_i}$ の積和型論理を作る

$$\overline{F} = m_1 \overline{f_1} + m_2 \overline{f_2} + m_3 \overline{f_3} + m_4 \overline{f_4} + m_5 \overline{f_5} + m_6 \overline{f_6} + m_7 \overline{f_7} + m_8 \overline{f_8}$$

- $F = \overline{\overline{F}}$ としドモルガンの定理を使用すると, 和積型の論理式ができる

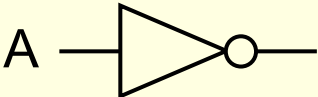
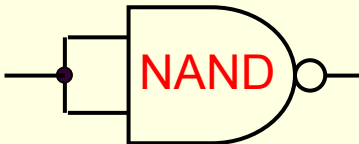
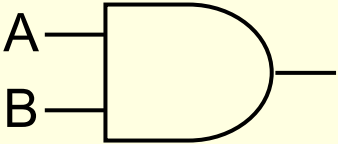
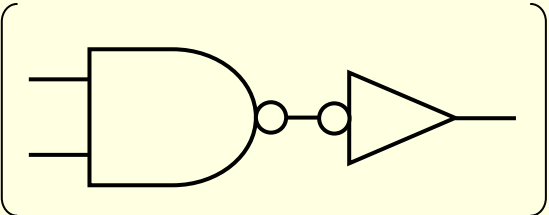
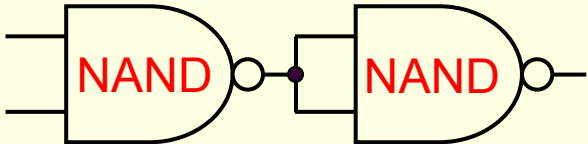
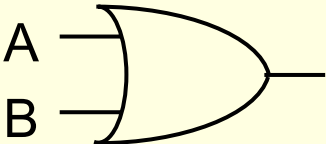
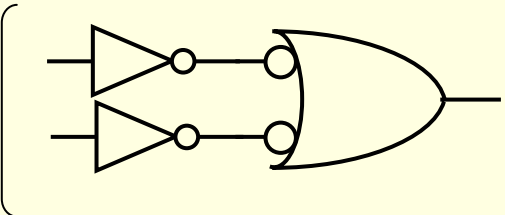
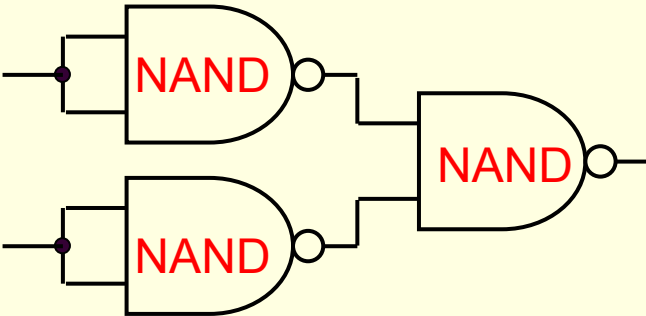
- 例: OR

$$\begin{aligned} f_1 = 0, f_2 \sim f_8 = 1 \\ \Rightarrow F = \overline{\overline{F}} = \overline{\overline{A \cdot B \cdot C}} \\ = A + B + C \end{aligned}$$

A	B	C	m_i	F_i	
0	0	0	$\overline{A} \cdot \overline{B} \cdot \overline{C}$	f_1	0
0	0	1	$\overline{A} \cdot \overline{B} \cdot C$	f_2	1
0	1	0	$\overline{A} \cdot B \cdot \overline{C}$	f_3	1
0	1	1	$\overline{A} \cdot B \cdot C$	f_4	1
1	0	0	$A \cdot \overline{B} \cdot \overline{C}$	f_5	1
1	0	1	$A \cdot \overline{B} \cdot C$	f_6	1
1	1	0	$A \cdot B \cdot \overline{C}$	f_7	1
1	1	1	$A \cdot B \cdot C$	f_8	1

論理関数の完全系(4)

- 任意の論理式はNANDまたは, NORの組み合わせで表現できる

NOT 	
AND 	 
OR 	 

論理代数の基本法則(1)

■ 同一則

- $A + A = A$
- $A \cdot A = A$

■ 相補則

- $A + \overline{A} = 1$
- $A \cdot \overline{A} = 0$

■ 吸収則

- $0 + A = A$
- $1 + A = 1$
- $0 \cdot A = 0$
- $1 \cdot A = A$

論理代数の基本法則(2)

■ 2重否定

- $\overline{\overline{A}} = A$

■ 交換則

- $A + B = B + A$

- $A \cdot B = B \cdot A$

■ 結合則

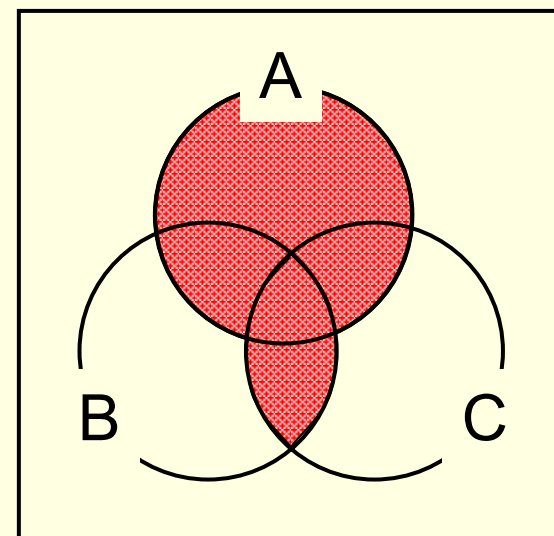
- $A + (B + C) = (A + B) + C$

- $A \cdot (B \cdot C) = (A \cdot B) \cdot C$

■ 分配則

- $A \cdot B + A \cdot C = A \cdot (B + C)$

- $(A + B) \cdot (A + C) = A + B \cdot C$



算術演算では理解
しづらいが,上記の
ベン図により明らか

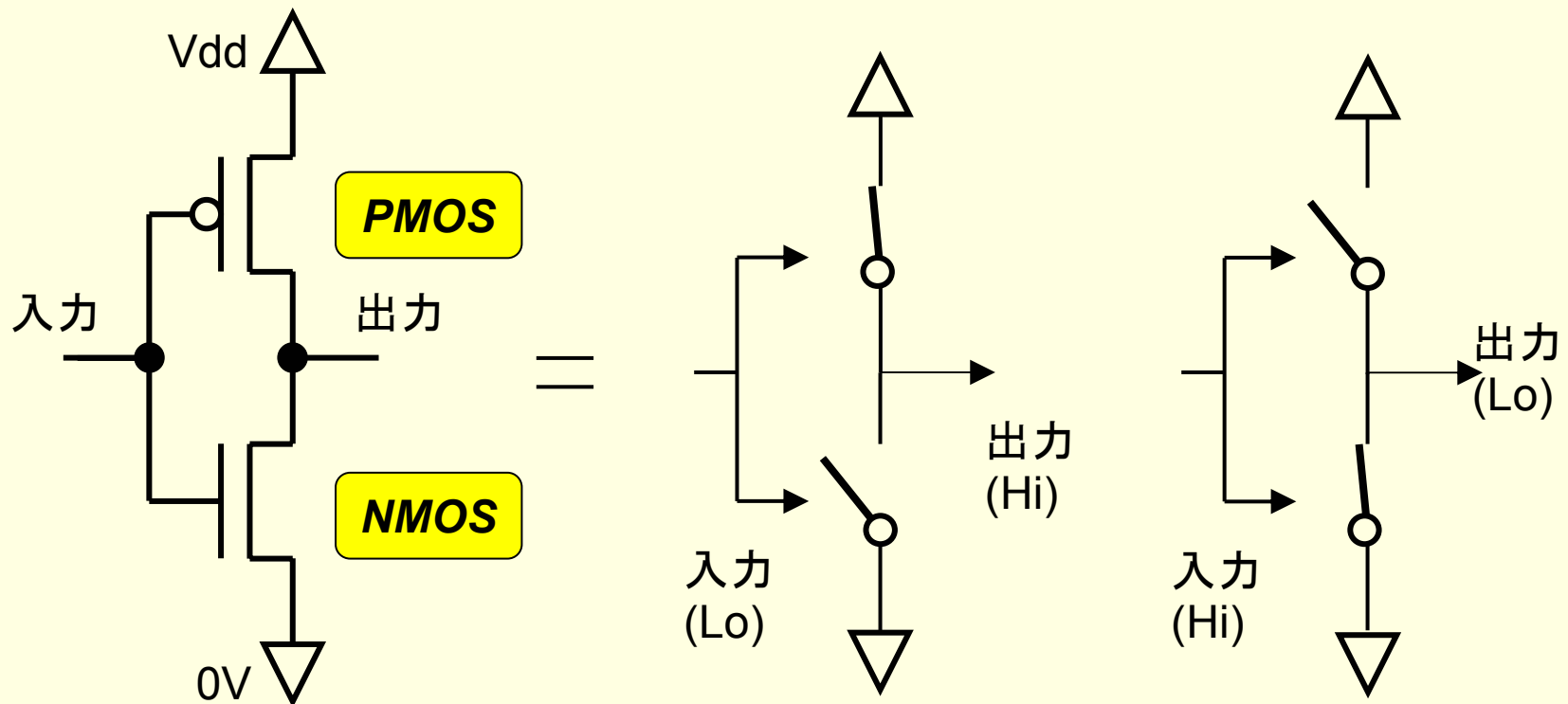
練習問題

- NOT,AND,ORをNORゲートのみを用いてMIL記号で表現せよ
- XOR(排他的OR)をNANDゲートのみを用いてMIL記号で表現せよ
- 下記の論理式を簡単にせよ
 1. $B\{A(A+B)+B \cdot C\}$
 2. $(A+\bar{B})(A+\bar{A} \cdot \bar{B})$

＊ 式の変形では一番簡単になっているか判断困難
⇒カルノー図, クワイン・マクラスキーの方法など
(論理回路の基礎: 田丸啓吉著: 工学図書etc.参照)

CMOSインバータのデジタル理解(復習)

- 入力電圧により, NMOS, PMOSTランジスタのいずれかが相補的(**C**omplimentary)にONしている⇒**CMOS回路**
 - 入力Hi(V_{dd})のときONするスイッチ: NMOS
 - 入力Lo(0V)のときONするスイッチ: PMOS

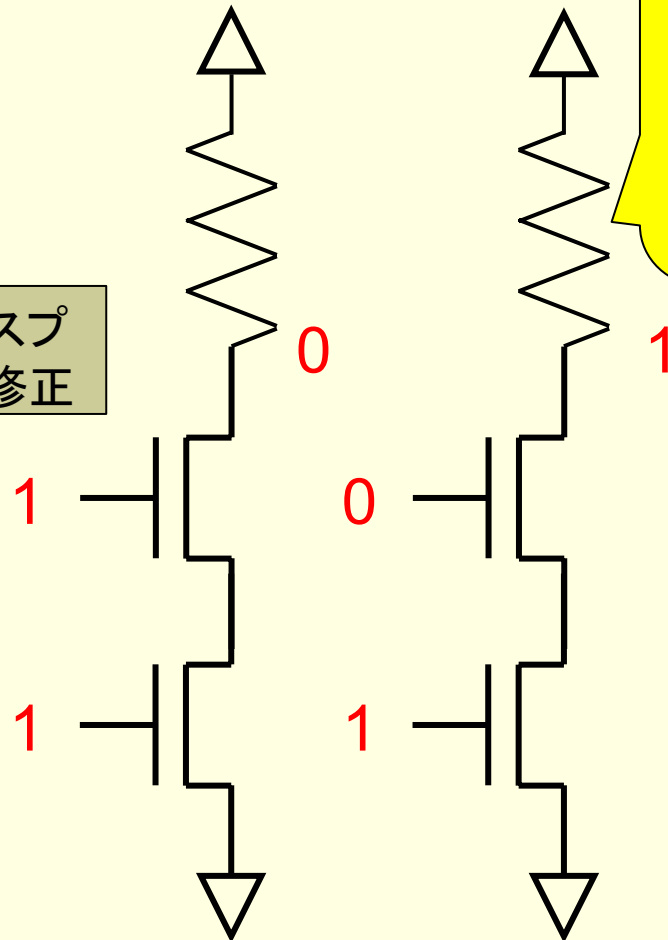
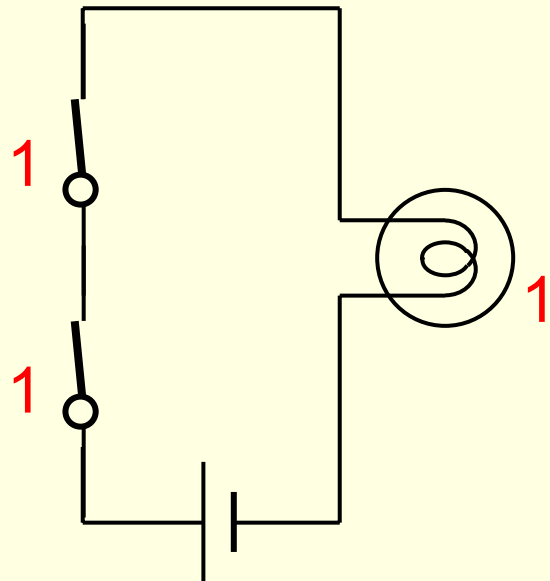


論理ゲートをMOSトランジスタで実現

■ AND論理とNMOSスイッチによるNAND

A	B	$F=A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

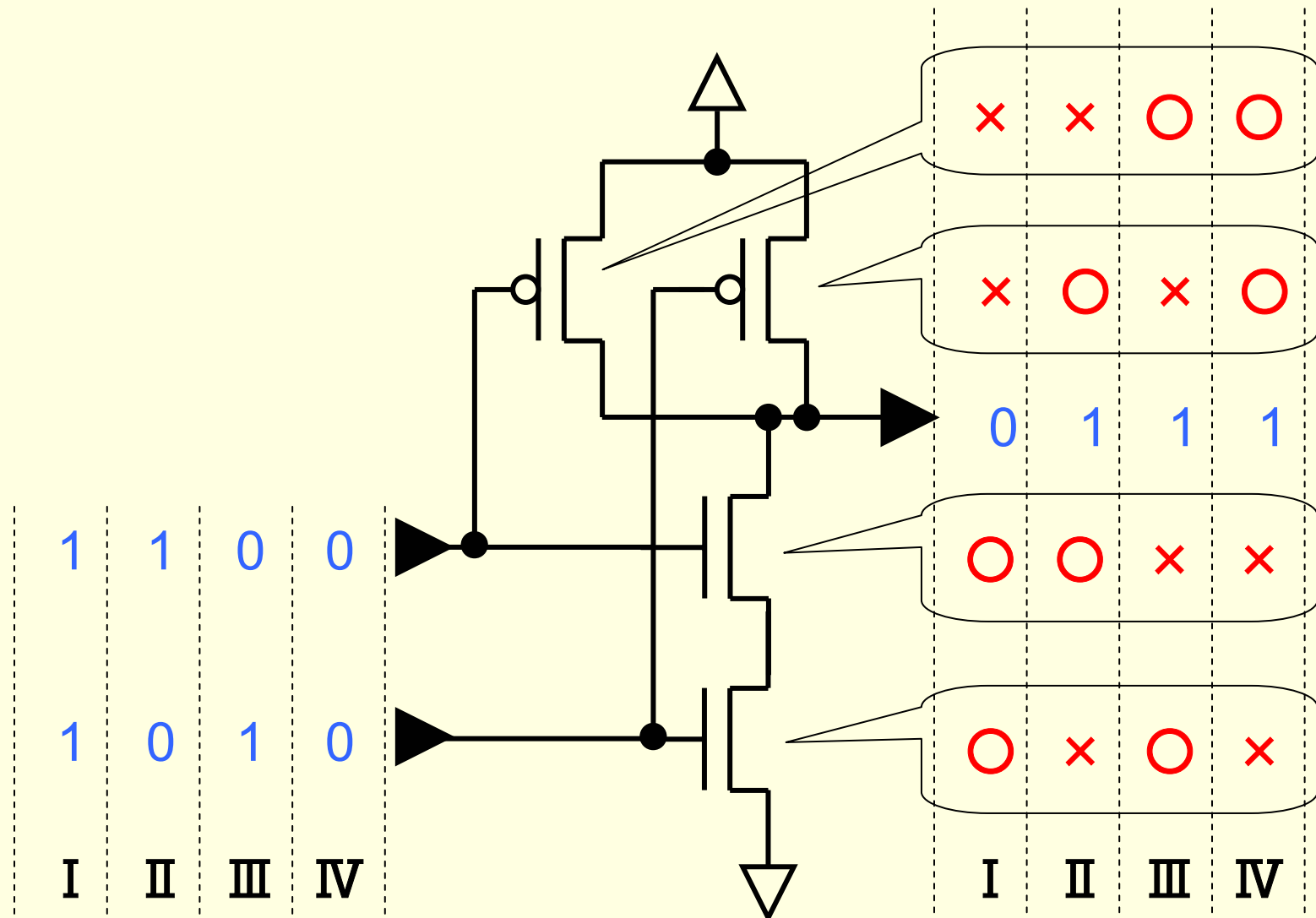
ミス
プリ修正



NMOSトランジスタと比較して
十分高い抵抗
⇒PMOSスイッチを使用する

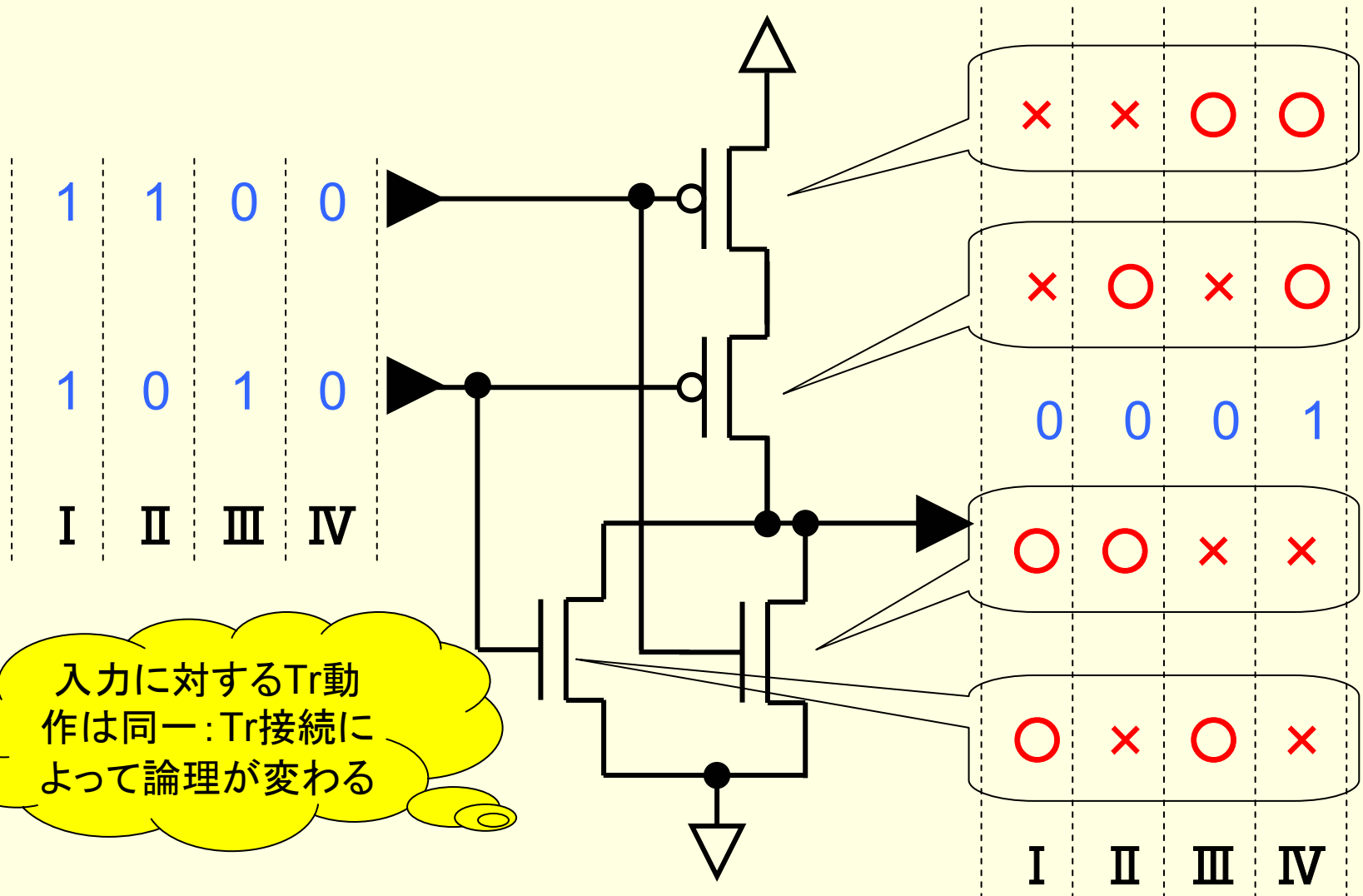
CMOS回路によるNAND

■ NMOSを直列, PMOSを並列に接続



CMOS回路によるNOR

■ NMOSを並列, PMOSを直列に接続



多入力の論理

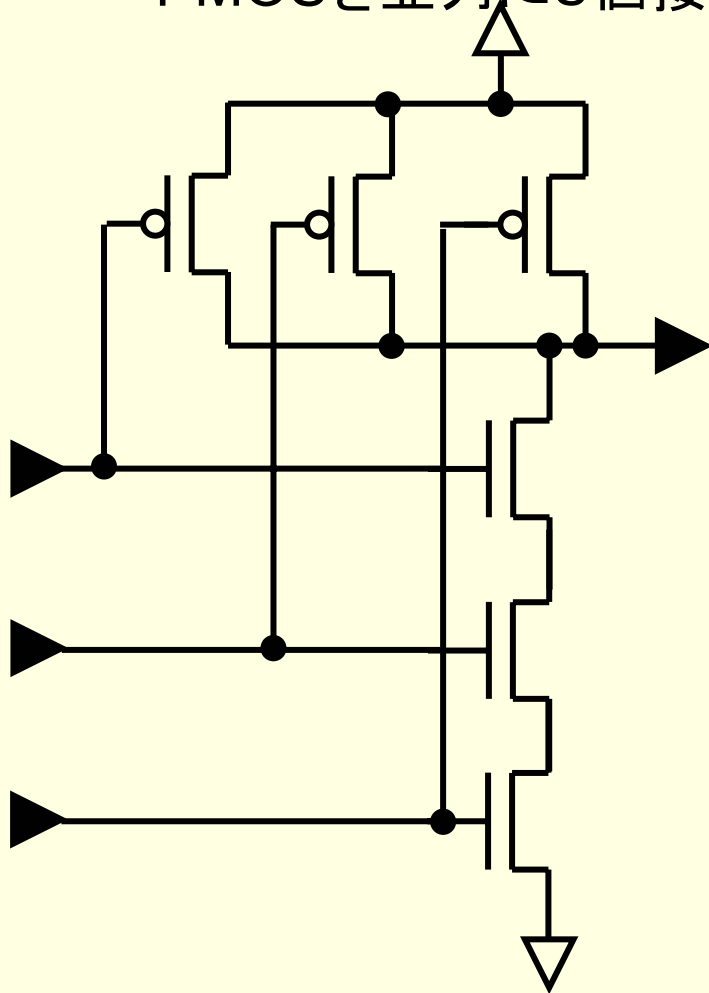
■ 3入力の場合

A	B	C	AND	OR	NAND	NOR	XOR
0	0	0	0	0	1	1	0
0	0	1	0	1	1	0	1
0	1	0	0	1	1	0	1
0	1	1	0	1	1	0	0
1	0	0	0	1	1	0	1
1	0	1	0	1	1	0	0
1	1	0	0	1	1	0	0
1	1	1	1	1	0	0	1

CMOS回路による3入力NAND, NOR

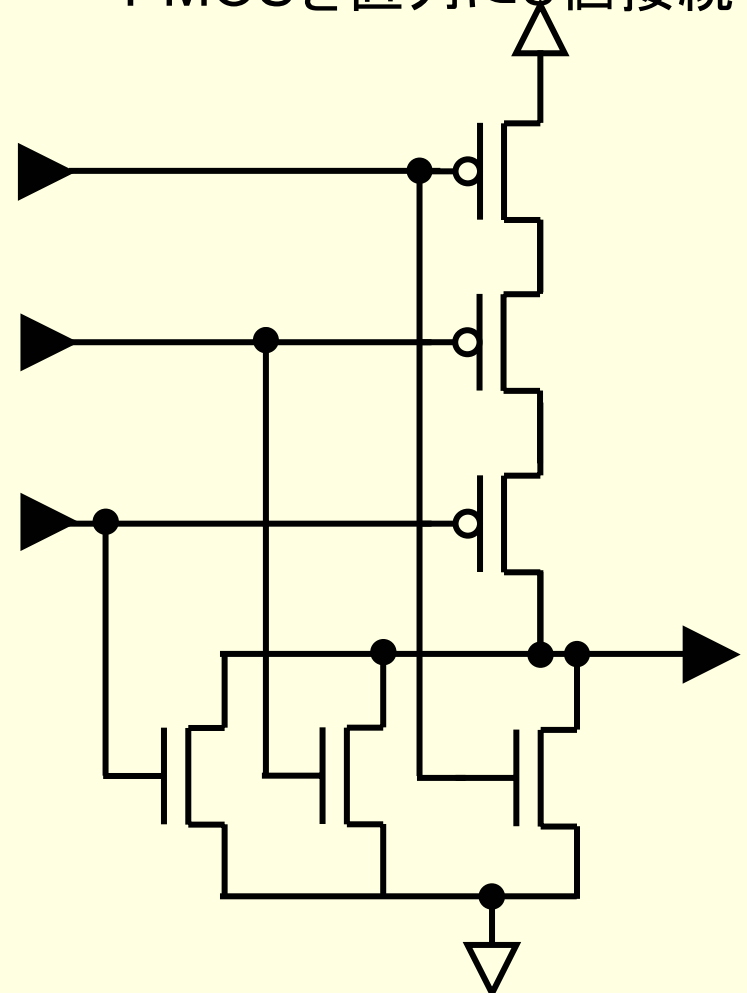
■ 3入力NAND

- NMOSを直列に3個, PMOSを並列に3個接続

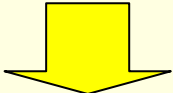


■ 3入力NOR

- NMOSを並列に3個, PMOSを直列に3個接続



注意

- NANDとNORどちらで論理構成するのがよいか
 - $I_d(\text{NMOS}) > I_d(\text{PMOS})$ のため, できるだけNMOSを多用するほうがよい
 - ゲート入力数はどこまで増やせるか
 - スイッチングの特性で4入力程度までが実用範囲
- 
- LSI設計においてTr特性の理解が重要！