

集積システム入門

高木 茂孝

本稿では、マイクロプロセッサの構成を例に取り、集積回路上に実現されるシステムとはどのようなものであるかを概説する。

1 論理変数と論理関数

1.1 データの表示方法

一般にマイクロプロセッサでは、“0”または“1”からなる数値列によって命令やデータを表示する。例えば、数値データの場合、2進数表示を用いれば、10進数9は1001と表される。この数値列の各桁は“0”か“1”かという二者択一の情報を有している。この情報の単位のことをビットと呼ぶ。1001は10進数9の4ビット表示であり、8ビット表示の場合は00001001となる。

数値データ以外の例として英数字を“0”または“1”の数値列によって表した例として表1に示すASCIIコードがある。

表 1: ASCII コード

文字	対応コード	文字	対応コード
0	0110000	A	1000001
1	0110001	B	1000010
2	0110010	C	1000011
⋮	⋮	⋮	⋮

1.2 論理演算

“0”か“1”かのいずれかの値を取る変数を論理変数と呼ぶ。ある論理変数を他の論理変数

表 2: 否定 (NOT)

論理値 (X)	演算結果 (Y)
0	1
1	0

に変換する操作を論理演算と呼ぶ。さらに、ある論理演算に関して与えられた論理変数とその演算結果を定義した表を真理値表と呼ぶ。

否定演算の真理値表を表2に示す。

論理変数を X 、演算結果を Y とすると、否定演算を

$$Y = \overline{X} \quad (1)$$

と表し、その記号による表現は図1である。

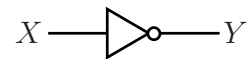


図 1: 否定演算の記号

表 3: 論理積 (AND)

論理値 1 (X ₁)	論理値 2 (X ₂)	演算結果 (Y)
0	0	0
0	1	0
1	0	0
1	1	1

論理積演算の真理値表を表3に示す。

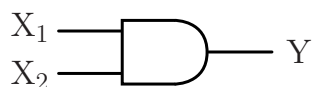


図 2: 論理積演算の記号

論理変数を X_1 と X_2 , 演算結果を Y とすると, 論理積演算を

$$Y = X_1 \cdot X_2 \quad (2)$$

と表し, その記号による表現は図 2 である.

表 4: 論理和 (OR)

論理値 1 (X_1)	論理値 2 (X_2)	演算結果 (Y)
0	0	0
0	1	1
1	0	1
1	1	1

論理和演算の真理値表を表 4 に示す.

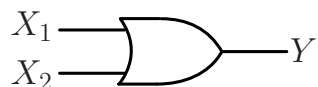


図 3: 論理和演算の記号

論理変数を X_1 と X_2 , 演算結果を Y とすると, 論理和演算を

$$Y = X_1 + X_2 \quad (3)$$

と表し, その記号による表現は図 3 である.

2 トランジスタと論理回路

2.1 論理値の回路的表現

集積回路上で論理演算を行うため, 論理値“0”や“1”を電圧の高低に対応させる. 対応方法を表 5 に示す.

一般に, 低電位とは接地電位 (0V) 付近を指し, 高電位とは電源電圧付近を指す.

表 5: 論理値と電位の対応

正論理:	0 → 低電位	1 → 高電位
負論理:	0 → 高電位	1 → 低電位

2.2 トランジスタの記号

集積回路で良く用いられるトランジスタとして MOS トランジスタがある. MOS トランジスタには電氣的極性が異なる 2 種類のトランジスタがあり, それぞれ n チャネル MOS トランジスタ, p チャネル MOS トランジスタと呼ばれている.

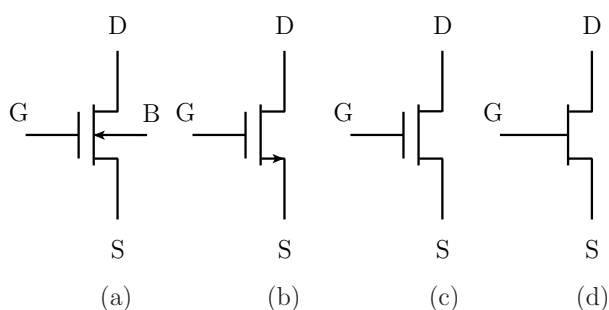


図 4: トランジスタの記号
(n チャネル MOS トランジスタ)

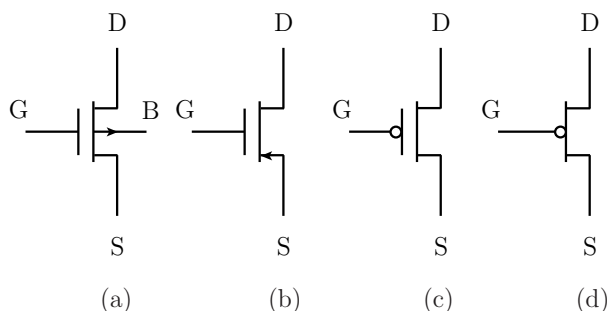


図 5: トランジスタの記号
(p チャネル MOS トランジスタ)

MOS トランジスタの記号としては, 図 4 と図 5 に示すように, 様々な種類が用いられている. 本稿では, 図 (c) の記号を用いることにする.

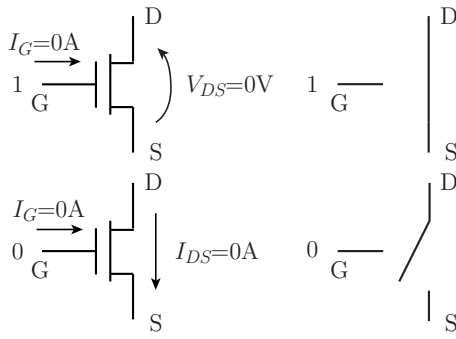


図 6: n チャネル MOS トランジスタのスイッチ動作

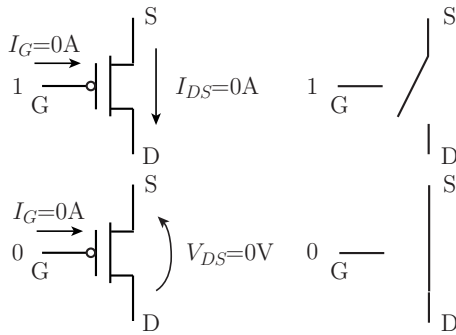


図 7: p チャネル MOS トランジスタのスイッチ動作

2.3 トランジスタのスイッチ動作

MOS トランジスタはゲート電位の高低に応じて開閉するスイッチと考えることができる。図 6 と図 7 に n チャネル MOS トランジスタ及び p チャネル MOS トランジスタのスイッチとしての動作を示す。

2.4 基本論理回路

トランジスタを組み合わせることにより、論理演算を実行する回路、すなわち論理回路を構成することができる。以下では、正論理に基づき、論理回路の動作を説明する。

図 8(a) に示す回路は否定演算を行う回路であり、NOT 回路と呼ばれている。その動作は、図 6 や図 7 に示される MOS トランジスタのスイッチ動作を考慮すれば、図 8(a) のゲ-

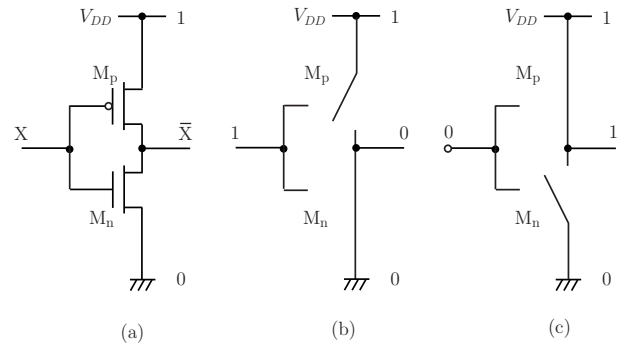


図 8: NOT 回路とその動作

トに論理値 “1” を与えた場合は、図 8(b) となり、論理値 “0” を与えた場合は、図 8(c) となる。従って、図 8(a) の回路は、表 2 に示す演算を実行する回路であることがわかる。

図 9 と図 10 はそれぞれ NAND 回路、NOR 回路と呼ばれている。NAND 回路は否定論理積回路、NOR 回路は否定論理和回路と呼ばれることもある。

図 8 と同様に、これらの回路を解析すると、表 6 と表 7 の真理値表が得られる。これらの真理値表から、NAND 回路は、二つの論理変数の論理積演算を行った結果の否定を出力し、NOR 回路は、二つの論理変数の論理和演算を行った結果の否定を出力する回路であることがわかる。従って、これらの回路の出力 (Y) に NOT 回路を付加すれば、AND 回路や OR 回路を実現することもできる。

表 6: 否定論理積 (NAND)

論理値 1 (X_1)	論理値 2 (X_2)	演算結果 (Y)
0	0	1
0	1	1
1	0	1
1	1	0

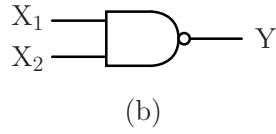
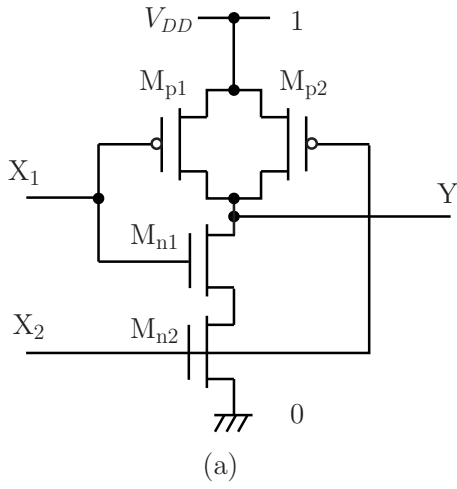


図 9: NAND 回路とその記号

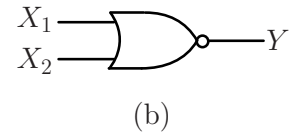
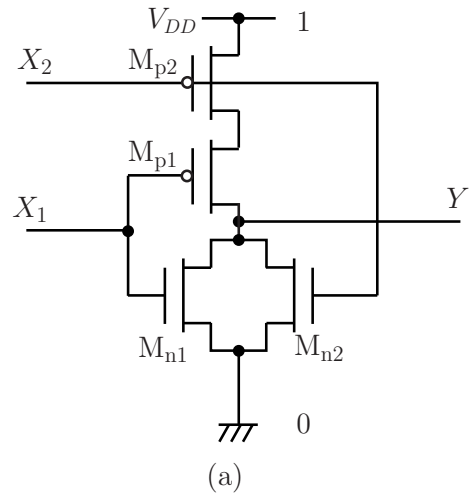


図 10: NOR 回路とその記号

表 7: 否定論理和 (NOR)

論理値 1 (X_1)	論理値 2 (X_2)	演算結果 (Y)
0	0	1
0	1	0
1	0	0
1	1	0

3 組み合わせ論理回路

現在の状態に依らず，現在の論理変数の値のみによって演算結果が定まる回路を組み合わせ論理回路と呼ぶ．ここでは，簡単な組み合わせ論理回路の例について説明する．

3.1 マルチプレクサ

複数のデータから一つのデータを選択する回路をマルチプレクサと呼ぶ．図 11(a) は論理変数 X_1 と X_2 の一方を選択する回路である． SEL が “1” の場合に， $Y = X_1$ となり，論理

変数 X_1 が選ばれる．また， SEL が “0” の場合， $Y = X_2$ となり， X_2 が選ばれる．図 11(b) にマルチプレクサの記号を示す．

3.2 加算器

1 ビットの 2 進数 A と B の和を S ，繰り上がりを C とすると， A と B の加算は表 8 となる．表 8 に示される演算を実行する回路が図 12 である．

表 8: 半加算器の真理値表

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

一般に， n ビットの 2 進数の加算を考えた場合，下位の桁の加算からの繰り上がりを考慮しなければならない．繰り上がりを考慮して