

§ 1. イントロダクション

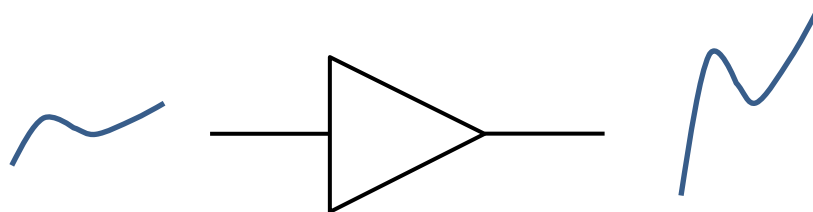
<なぜデジタル電子回路を学ぶのか？>

デジタル化により、大きく社会は変化し、これからも大きく変化しようとしている。その仕組みを理解するのが、この講義の目的である。

なぜ、デジタル化するのか？ それは、情報がコンピュータで処理できるからである。また、コンピュータを使うと、仕事の効率を飛躍的に向上させることができ、大幅なコストダウンが可能となる。これが、デジタル化の motivation である。

<アナログとデジタル>

アナログ (analogue) とは？ ana + logue 相似に増幅される信号から名前がつけられた
相似 相似の 言葉



デジタル (digital) とは？

digitus (指：ラテン語) → 数字で表された (とびとびの)

<アナログメディア (機器) とデジタルメディア (機器) >

	アナログ	デジタル
音楽記録	レコード盤	CD
録音	カセット (磁気) テープ	デジタルプレーヤー (iPod)
写真	フィルムカメラ	デジカメ
録画	磁気テープ (VHS)	HD, DVD, Blu-ray
電話	アナログ電話 (固定電話)	携帯電話, IP 電話
テレビ	アナログテレビ	デジタルテレビ (地デジ)
文書伝送	手紙	電子メール
新聞	新聞紙	インターネット, 電子リーダー
本	本	電子ブック

<デジタルの優位性>

1. 複写が完全
2. 長期保存が可能：情報が劣化しにくい（全く劣化しない訳ではない）
3. 通信が容易
4. 検索が容易
5. 処理や加工が可能
6. 記憶容量が大きくなる（記録媒体が小さくなるため）

<デジタル化（離散化）における問題点>

1. 標本化（サンプリング）密度（速度）の問題：時間的にとびとびにサンプル化すると，エリザベス女王の問題が発生する（映画やTVで回転するプロペラや扇風機が逆回転して見える現象）
2. 量子化誤差：標本化の精度が不足すると情報が欠落する
3. 直感的把握に難点がある：アナログ時計など
4. 一度に大量のコピーができるため，セキュリティ的に脆弱である。

§ 2. ブール代数

デジタル信号を扱うための論理回路の信号は，ほとんどの場合，二つの値で表す。

これを二値論理という。論理回路の信号を，三つ以上の値で表す方法もあり，フラッシュメモリなどで応用されているが，これは例外的な場合である。

よって，論理回路の信号は，ブール代数によって扱われる。

論理変数 A を考える。 A は，0 または 1 の値をとる。論理変数 A, B に対し，次の演算を考える。

1. 否定 (NOT) $\bar{A}$
2. 論理和 (OR) $A + B$
3. 論理積 (AND) $A \cdot B$

これらの演算は，次の関係を満たすものと「定義」されている。このような表を真理値表 (Truth table) という（電圧で書いたものは動作表という）。

A	$\bar{A}$				A	B	$A+B$	A	B	$A \cdot B$
0	1	0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	1	0	0	1	0
		1	0	1	1	0	1	0	0	0
		1	1	1	1	1	1	1	1	1

これらより、次の法則が成り立つことが分かる（簡単に証明することができる）。

1. 交換則 $A+B=B+A$, $A \cdot B=B \cdot A$
2. 結合則 $A+(B+C)=(A+B)+C$, $A \cdot (B \cdot C)=(A \cdot B) \cdot C$
3. 分配則 $A \cdot (B+C)=A \cdot B+A \cdot C$,
4. 吸収則 $A+0=A$, $A+1=1$, $A \cdot 0=0$, $A \cdot 1=A$ （和の吸収則に注意）
5. ド・モルガンの法則 $\overline{A+B}=\bar{A} \cdot \bar{B}$, $\overline{A \cdot B}=\bar{A}+\bar{B}$ （論理の置き換えに使用する）

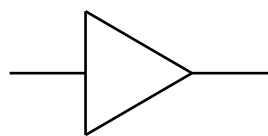
（証明）

A	B	$\bar{A}$	$\bar{B}$	$A+B$	$\overline{A+B}$	$\bar{A}+\bar{B}$	$A \cdot B$	$\overline{A \cdot B}$	$\bar{A} \cdot \bar{B}$
0	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0	1	0
1	0	0	1	1	0	1	0	1	0
1	1	0	0	1	0	0	1	0	0

（第1回目の講義はここまで）

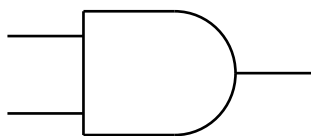
§ 3. 論理回路の記号

論理回路は、通常、MIL 記号（Military Standard：米国陸軍規格）で表される。

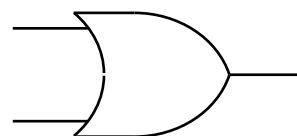


バッファ

(論理値は変わらないがレベル保持) $Y = A \cdot B$

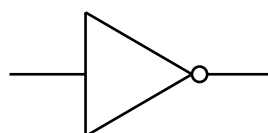


AND ゲート



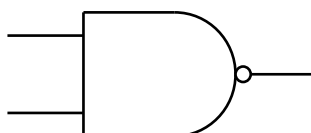
OR ゲート

$Y = A + B$



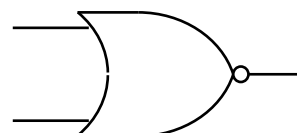
NOT ゲート

$Y = \bar{A}$



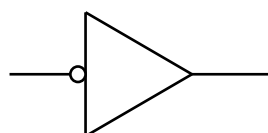
NAND ゲート

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



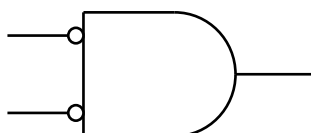
NOR ゲート

$Y = \overline{A + B}$



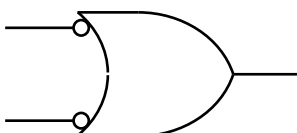
NOT ゲート

$Y = \bar{A}$



NOR ゲート

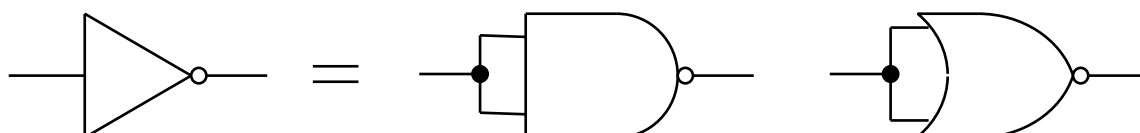
$Y = \overline{A \cdot B} = \overline{A + B}$



NAND ゲート

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

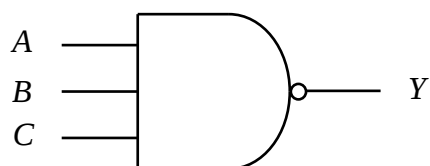
このように、基本的な論理演算機能をもつものをゲートと呼ぶ。また、NOT 回路は、NAND ゲートもしくは NOR ゲートを用いて、以下のように表すことができる。



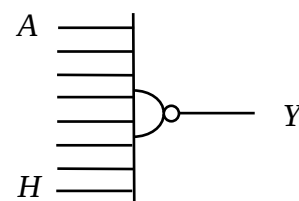
デジタル LSI の規模はゲート数、もしくはトランジスタ (FET) 数で表現する。

すべての論理回路は、これらのゲートを用いて作成することができる。ド・モルガンの法則を用いると、AND ゲートを OR ゲートで、OR ゲートを AND ゲートで実現することができる。すなわち、NAND ゲートもしくは NOR ゲートのみで、すべての論理回路を構成することができる。

この他に、多入力のゲートも定義され、しばしば使用されている。



$Y = \overline{ABC}$

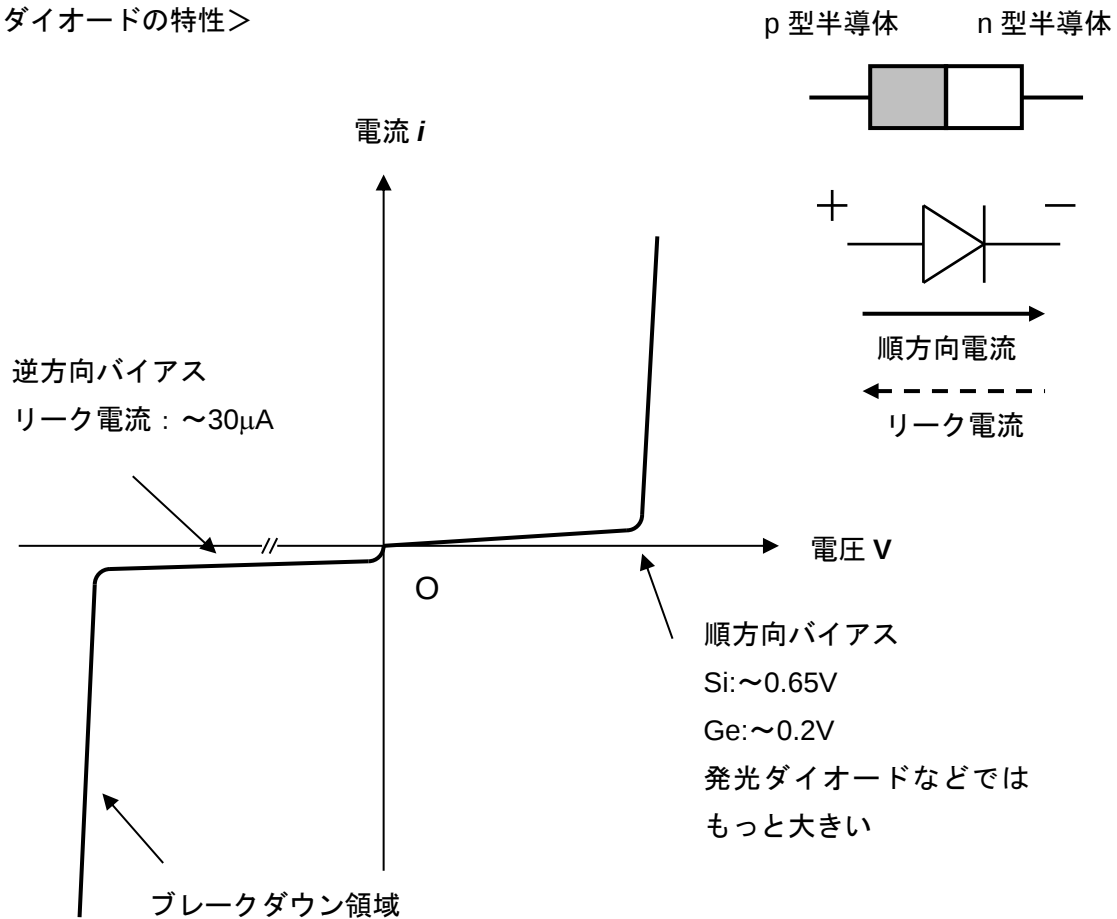


$Y = \overline{ABCDEFGH}$

§ 4. 論理回路の構成方法

論理回路は、非線形な電子回路。よって非線形な素子を使う必要がある。

<ダイオードの特性>



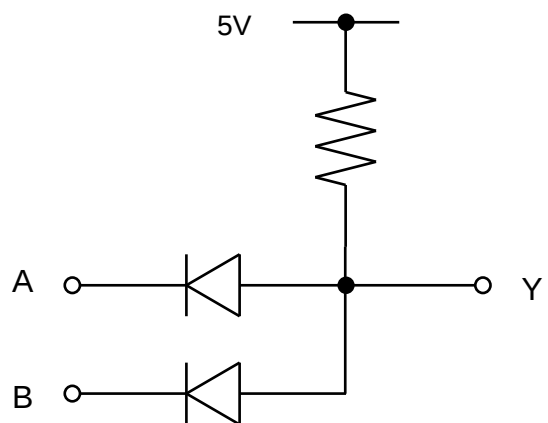
導通 (1~10mA) の電流を流したときに、約 0.7V の電圧降下がある。

<ダイオード, トランジスタ, 抵抗による構成>

電源電圧は、特に断らない限り 5V ($\pm 0.25V$ (5%)) とする。現在の論理回路の標準的電圧である。最近では、3.3V の電源電圧も広く使われている。

また、L, H の入力レベルはそれぞれ 0V (グランドもしくはアース), 5V (電源電圧) とする。

<AND 型回路>



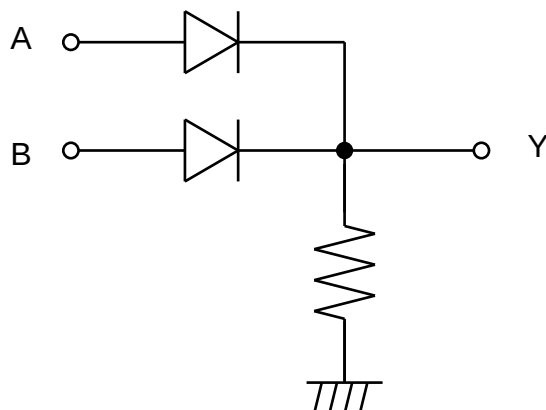
電気抵抗：旧と新

動作表

A	B	Y
0V	0V	0.7V
0V	5V	0.7V
5V	0V	0.7V
5V	5V	5V

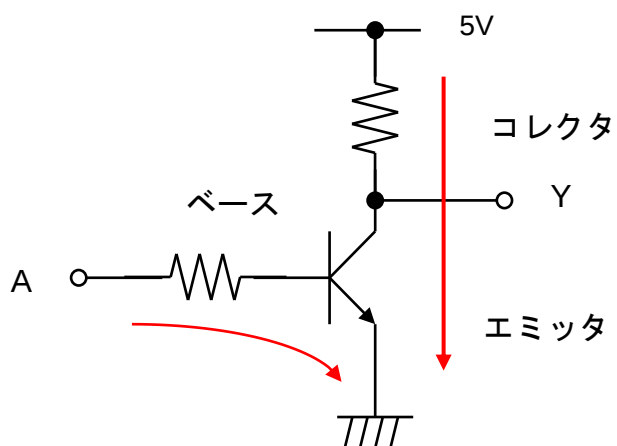
正論理 (0V (L) → 0, 5V (H) → 1) の時には AND 回路であるが, 負論理 (H→0, L→1) の時には, OR 回路となる.

<OR 型回路>



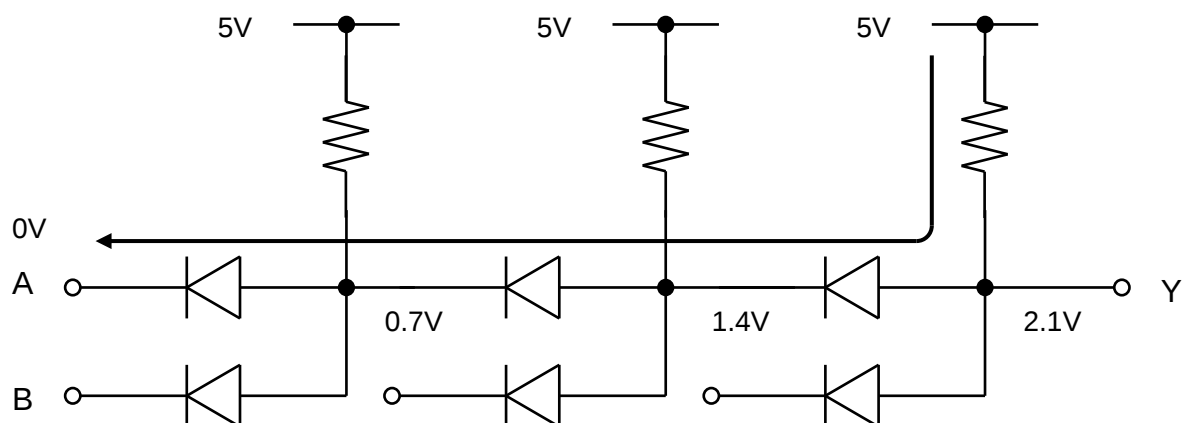
A	B	Y
0V	0V	0V
0V	5V	4.3V
5V	0V	4.3V
5V	5V	4.3V

<NOT 型回路：エミッタ接地回路，ベース電流によるスイッチング>

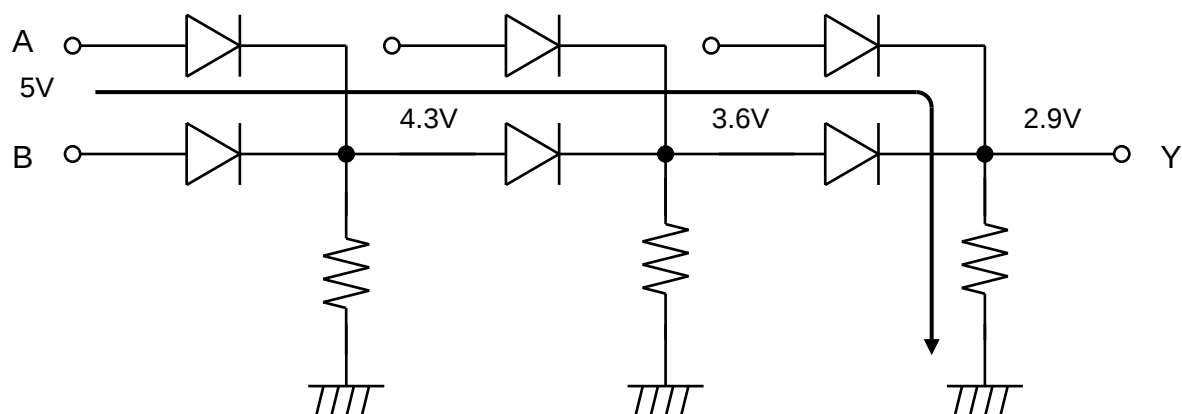


A	Y
0V	5V
5V	~0V

<これらの回路の問題点>：今回の授業ではやっていませんが、参考のため掲載します。

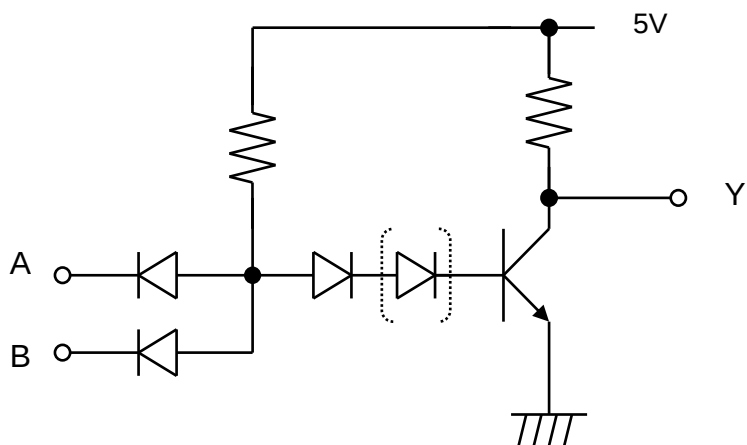


AND 型回路の L レベル出力は、1 段で 0.7V、2 段で 1.4V、3 段で 2.1V と、電圧レベルが上昇していく。よって、何段もつなぐと、L レベルと H レベルの区別がつかなくなる。



OR 型回路の H レベル出力は、1 段で 4.3V、2 段で 3.6V、3 段で 2.9V と、電圧レベルが低下していく。よって、何段もつなぐと、L レベルと H レベルの区別がつかなくなる。

<DTL による NAND 回路>



基本的には、AND 型回路と NOT 回路の組み合わせであるが、動作を安定させるために、間にダイオードを 2 個程度挿入して閾値を上昇させる。

この回路は、Diode Transistor Logic と呼ばれ、1960 年代に使用された。

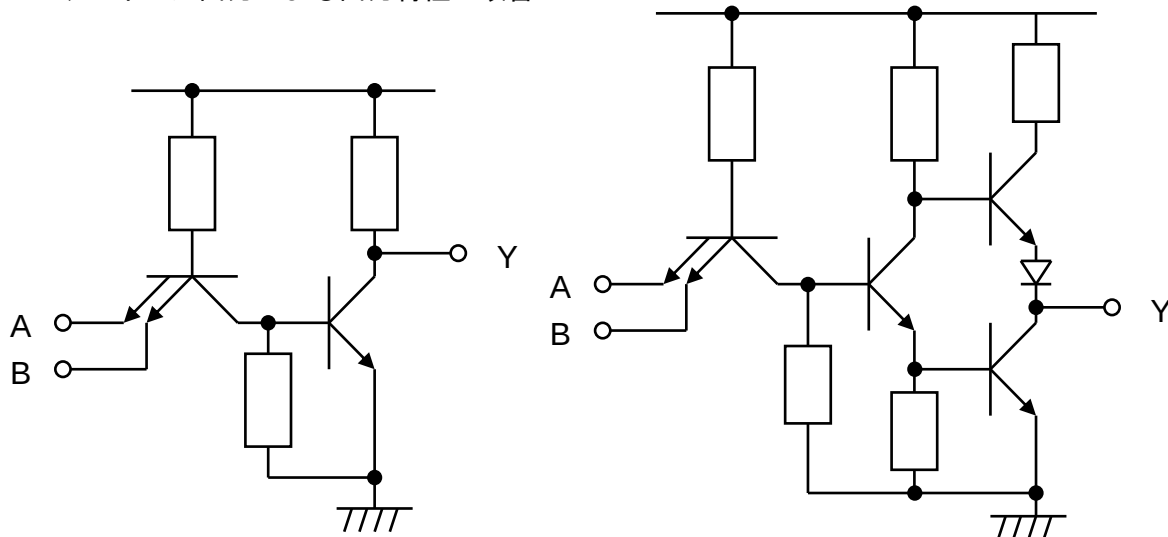
§ 5. TTL 素子（標準的論理 IC）

DTL には、いくつかの欠点があり、それらを改良した TTL（Transistor-Transistor Logic）が 1962 年に開発され、1970 年代以降広く使われることになった。現在は、CMOS に論理回路の主役を譲っているが、現在でも、使いやすさなどの点で広く使われている。

＜DTL からの改良点と回路構成＞：今年の授業ではやっていませんが、参考のため掲載します。

マルチエミッタ入力による集積の容易さ・速度の改善

トータムポール出力による出力特性の改善



TTL による NAND 回路

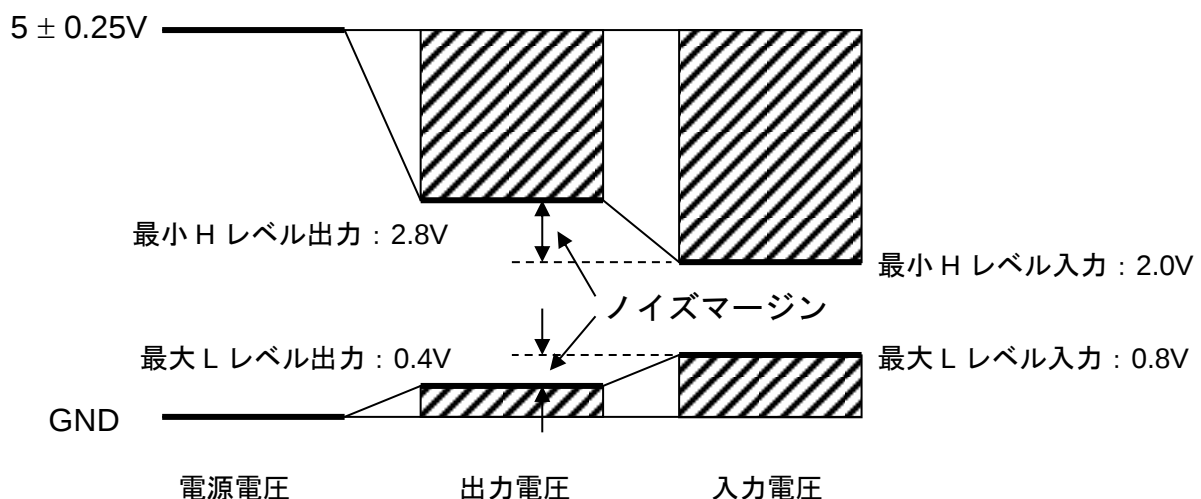
TTL の実用的回路

＜スレショルド電圧（Threshold voltage）＞

H と L はどこで区別されるか？どのようにして決められるか？

論理回路で最も優先すべきことは信頼性。何百万にも及ぶ論理ゲートをどのようにしてエラーなく動かすか？論理ゲートは、1カ所でも誤動作するとシステムとして正しく動作しない。

一つ一つのゲートができるだけ正しく動くように規格を決める必要がある。このため、H と L の規格は、出力と入力で別々に決定されている。



最小 H レベル出力 (2.8V) : これ以上の電圧が出力されるように規定されている

最大 L レベル出力 (0.4V) : これ以下の電圧が出力されるように規定されている

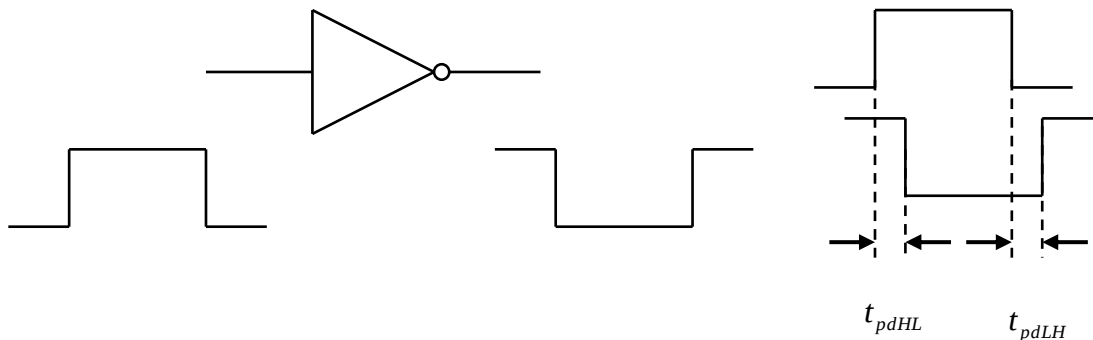
最小 H レベル入力 (2.0V) : これ以上の電圧を H レベルと認識するように規定されている

最大 L レベル入力 (0.8V) : これ以下の電圧を L レベルと認識するように規定されている

最小 H レベル出力 (2.8V) と最小 H レベル入力 (2.0V) の差 0.8V と、最大 L レベル出力 (0.4V) と最大 L レベル入力 (0.8V) の差 0.4V は、許容されるノイズという意味で、ノイズマージン（雑音余裕度）と呼ばれる。ノイズマージン以下のノイズであれば、論理回路は誤動作しないが、それ以上のノイズがあれば、誤動作する可能性が出てくる。

<スイッチング特性> : 今年の授業ではやっていませんが、参考のため掲載します。

TTL などの論理素子を使用する上では、動的な特性を知る必要である。論理素子の動的な特性を特徴づける量が、入力の変化に対する出力の変化の遅れ時間である、伝搬遅延時間（propagation delay）である。



伝搬遅延時間は、TTL では、2~10ns である。

<消費電力>

TTL は、L の出力の時に電流が流れるため、消費電力は比較的大きく、1 ゲートあたり、1~2mW である。すなわち、10,000 ゲートで 10~20W になるため、高集積度には適していない。

ちなみに、1995 年頃に発表されたペンティアムプロセッサは、約 80 万ゲートであったため、TTL で製作すると 800~1600W となる（実際は CMOS であり、消費電力は 10W 程度であった）。

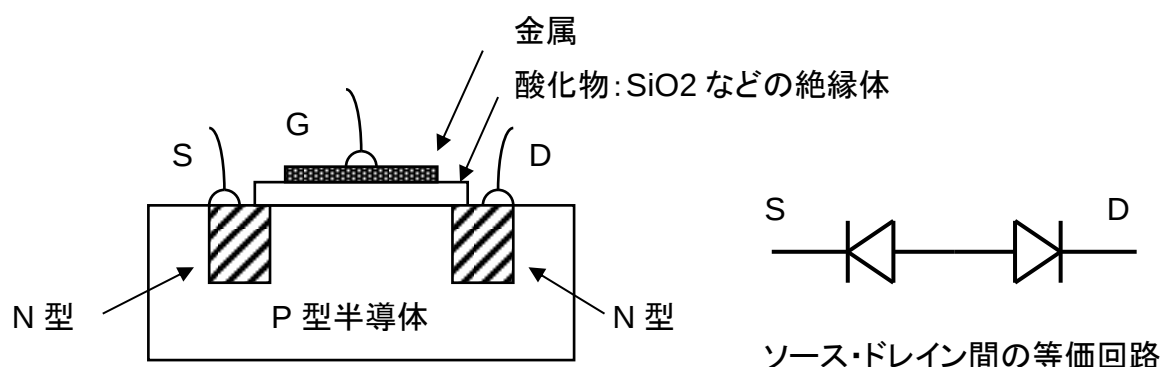
§ 6. C-MOSによる論理回路

MOS とは、Metal Oxide Semiconductor の略であり、金属酸化物半導体と訳される。

MOS とは、これらが層構造をもつものであり、流れるキャリアの違いにより、N 型 MOS FET と P 型 MOS FET がある (FET とは Field Effect Transistor の略で電界効果トランジスタと訳される)。

そして、NMOS FET と PMOS FET を組み合わせたゲート構造を、CMOS (Complementary Metal Oxide Semiconductor : 相補的金属酸化物半導体) と呼び、CMOS 構造を持つ論理 IC は、TTL に変わって、現在、論理 IC の標準となっており、LSI もほとんどが CMOS で作られている。

<NMOS FET の構造>

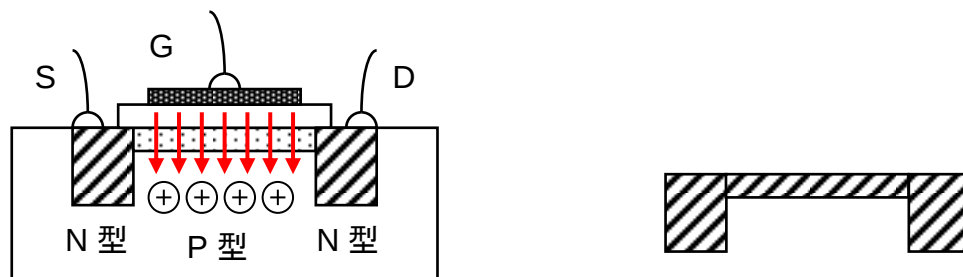


上に、NMOS FET の構造 (断面図) を示す。

まず、P 型半導体の中に、図のように N 型半導体を作り、その上に SiO₂ などの酸化物 (絶縁体) を被せる。その酸化物の上に、金属電極をつける。それぞれにリード線をつけ、左から、ソース (source : S)、ゲート (gate : G)、ドレイン (drain : D) と名付ける。

ゲートに正の電圧を印加する (正にバイアスする) と、ゲートから p 型半導体に向かって電界が発生し、p 型半導体の中のホールは、ゲートから離れた方向に押し出され、ゲートと p 型半導体の間に、n 型半導体のような状態 (MOS 反転層 : 下の図のドットで示した領域) ができる。

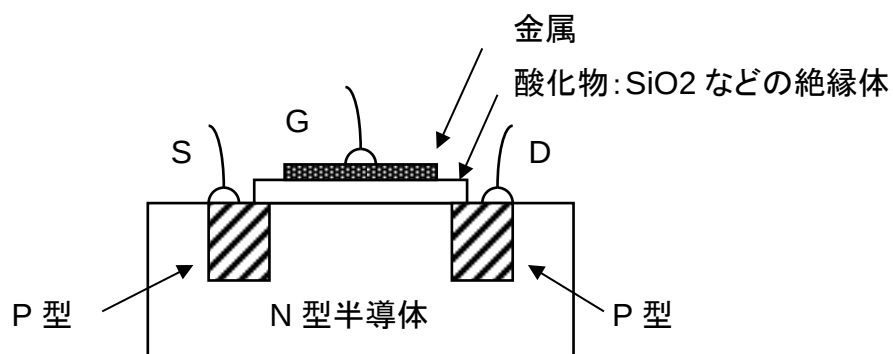
ソースとドレイン間は、上の図の右に示すように、ダイオードが逆に接続された構造をしているため、その間に電圧を加えても電流は流れないが、下のように MOS 反転層 (これを N 型のチャンネルと呼ぶ) ができると、N 型半導体の通路が完成され、電圧を加えると電流が流れる。チャンネルの幅は、ゲート電圧で制御できるため、増幅作用を持たせることもできる。



ゲートバイアスの印加による MOS 反転層の形成

反転層の形成による N 型半導体の生成

<PMOS FET の構造>



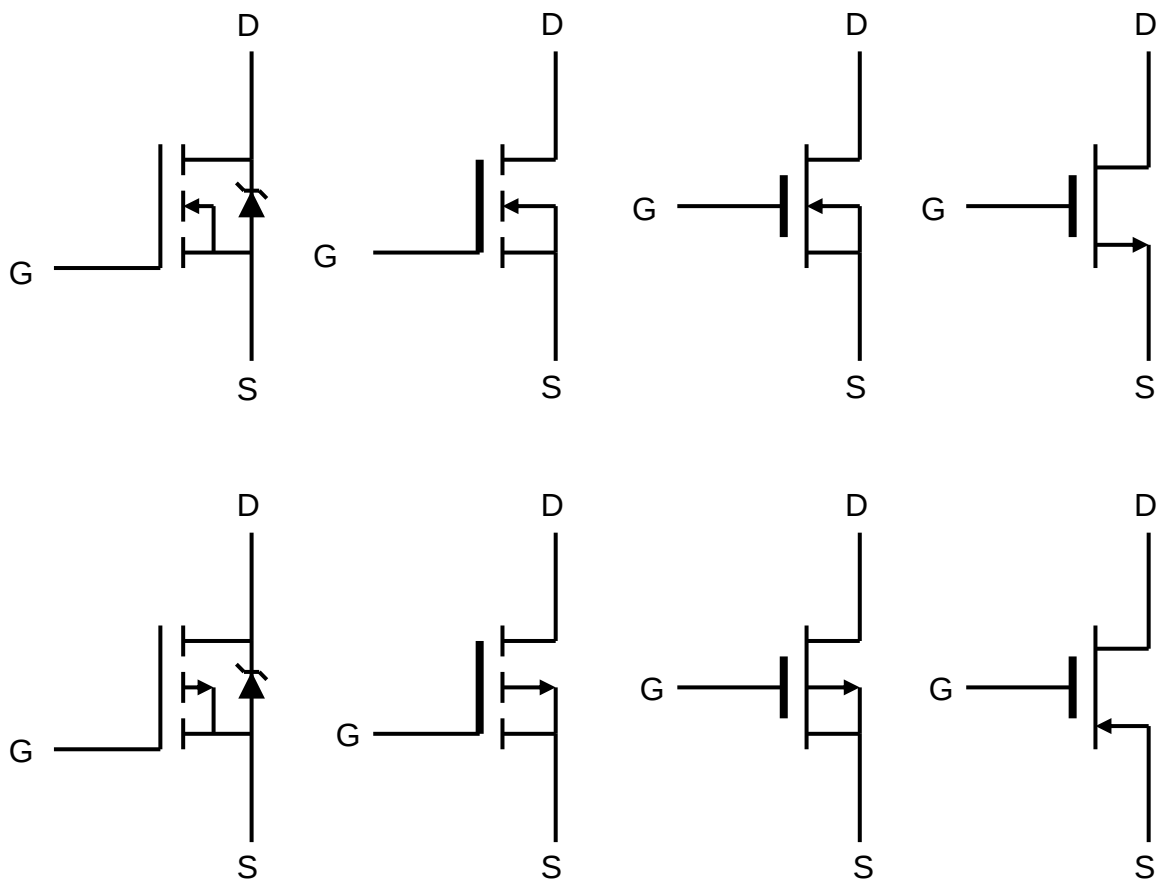
NMOS FET に対して、N 型半導体と P 型半導体を入れ替えただけの構造となっている。

ただし、ゲートをソースに対して負にバイアスしたときに、MOS 反転層が形成され、ソース・ドレイン間が電氣的にオンとなる。

<MOS FET の記号> : 今年の授業ではやっていませんが、参考のため掲載します。

MOS FET の記号としては、以下のいくつかのものが使われている。

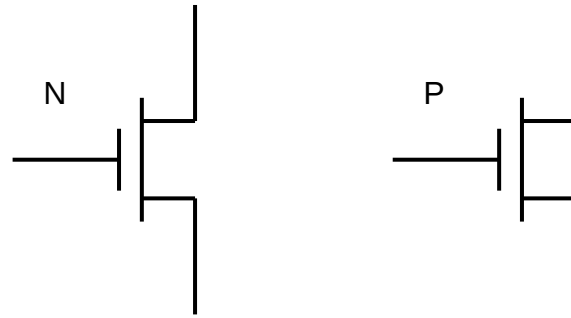
上段が NMOS FET, 下段が PMOS FET.



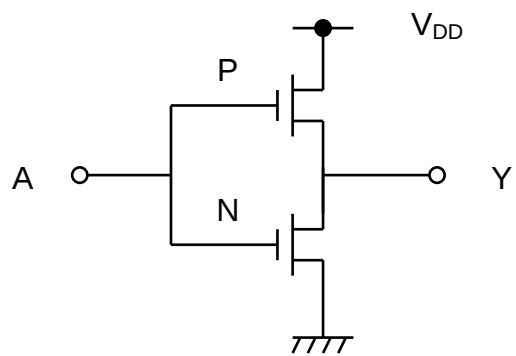
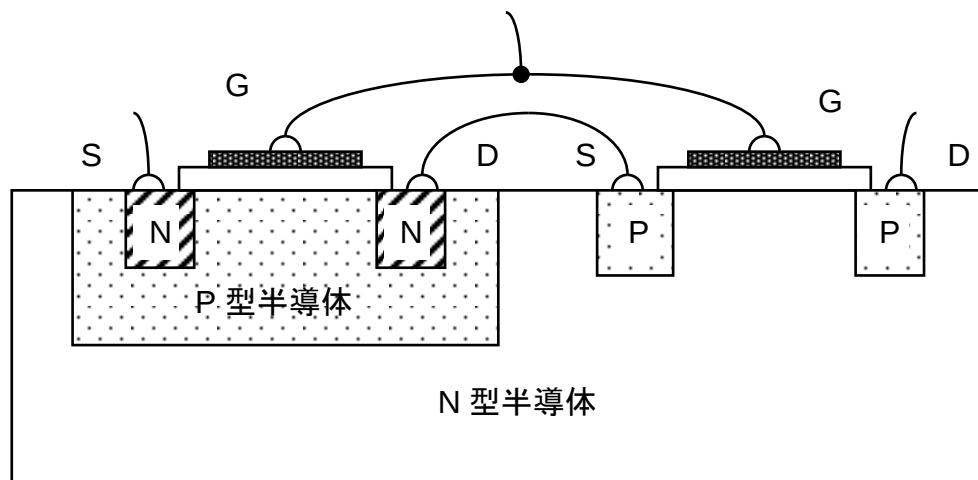
一番詳細な記号. DS 間のツェナーダイオードが記載されている.

DS 間のツェナーダイオードが省略されている.

ここでは、簡単のため、NMOS FET と PMOS FET をそれぞれ、以下のように記す。



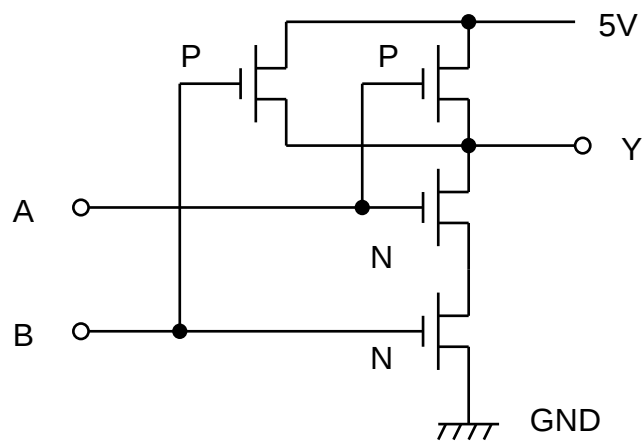
<CMOS による反転ゲート>



$V_{DD} = 3\sim 18V$

A	Y
L	H
H	L

<CMOS による NAND 回路> : 今年の授業ではやっていませんが、参考のため掲載します.



A	B	Y
L	L	H
L	H	H
H	L	H
H	H	L

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

<CMOS の特徴>

CMOS の特長

- (1) 回路が単純で集積化が容易.
- (2) 消費電力が少ない (電流は静止状態では流れず, スイッチングの時に流れる).

CMOS の欠点

- (1) 静電気で壊れやすい.
- (2) 電流ドライブに不向き (このような時には TTL などを使う).

§ 7. ゲート回路の組み合わせによる論理回路

<組み合わせ回路と順序回路>

論理回路は、次の2つの種類に分けられる。

○組み合わせ回路 (combinational circuit) :

入力の論理値だけで組み合わせだけで出力が決まる回路

$$Y = f(A, B, C, \dots)$$

論理回路に時間の概念はない。静的な状態。

○順序回路 (sequential circuit)

現在の入力の論理値だけでなく、前の状態の出力によって出力が決まる回路

$$Y_{n+1} = f(Y_n, A, B, C, \dots)$$

論理回路に、時間ないし状態の概念が入ってくる。

順序回路の基礎をなすものが、フリップ・フロップである。

§ 8. フリップフロップとその応用

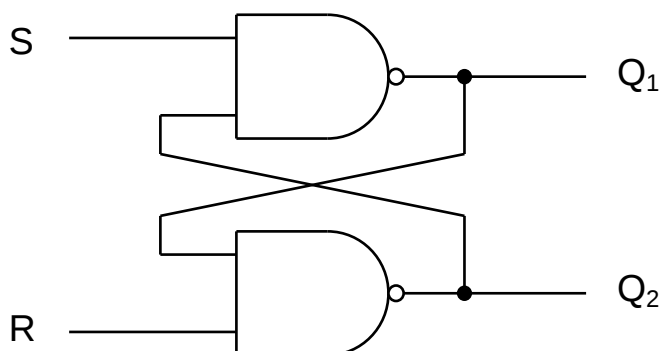
フリップ・フロップ (flip flop : FF) とは？

二つの安定な状態を持ち、その間が移り変わる電子回路。フリップとフロップは、蝶の羽の羽ばたきを表す擬態語である (wooden flip-flop とは下駄のことらしい)。

FF の中で、最も簡単で基礎的なものが RF-FF であり、最も複雑で多機能なものが JK-FF であるので、これらの2つの FF を解説する。

<RF-FF>

RS-FF は、以下のように、2つの NAND 回路を以下のように接続した回路である。



接続図

S	R	Q ₁	Q ₂
L	L	H	H
L	H	H	L
H	L	L	H
H	H	*	*

動作表

S と R が、両方 H のとき、 Q_1 と Q_2 の出力は、HL と LH のどちらにもなりうる。

すなわち、その前の状態で、S と R が、L と H で、出力が HL となっていて、S が L から H になったのであれば、出力は HL のままである。逆に、S と R が H と L で、出力が LH となっていて、R が L から H になったのであれば、出力は LH のままである。このように、どちらの出力になるかは、前の状態の影響を受けるため、この回路が順序回路であることを示している。

なお、S は Set（数を置く）を意味し、Preset とも呼ばれる。S を L にすることにより、 Q_1 に 1 という数を置くことになる。R は Reset（Set をやめる）を意味し、R を L にすることにより、 Q_1 に 0 を置くことになる。

このように、S と R の入力、L 入力の時に有効になり、このような入力を、負論理入力、Low active 入力などと言う。