

# コンピュータ・マテリアル・サイエンス 第1回

## - コンピュータのしくみ -

名古屋工業大学セラミックス基盤工学研究センター 井田 隆

2001年9月11日

### 1. はじめに

コンピュータ・マテリアル・サイエンスでは、コンピュータを利用した材料科学について講義をしますが、私が分担する講義(4回)では、はじめに「コンピュータはどのような仕組みでものを考えたり、おぼえたりしているのか?」という素朴な問題について説明し、その後に、コンピュータを利用して実験データを統計的に処理するための具体的な方法について紹介する予定です。

20世紀の後半から、コンピュータは急速に世の中に普及してきました。現在、日本でふつうの社会生活を送っている限り、否応無しになんらかの形でコンピュータを利用しているような状況になっています。

たとえば、私の自宅にマイコン(=マイクロコンピュータ)を搭載した電化製品がどれだけあるかという... パソコン2台、プリンタ、携帯電話はいうまでもなく、エアコン、ガスファンヒーター、テレビ、ビデオデッキ、ビデオカメラ、衛星チューナー、オーディオ、洗濯機、炊飯器、扇風機、ミシン、デジタルピアノ、電話/FAX、こどものおもちゃ7~8個などを思い付き、探せばまだありそうなくらいです。いわゆる8ビットのマイコンは今では安いものなら100円くらいで売っているようです。

多くの人も、このようにたくさんのコンピュータに囲まれた生活を送っているだろうと思うのですが、コンピュータのしくみをちゃんと理解している人はあまり多くないのではないのでしょうか。もちろん、コンピュータのしくみを知らなくても使うことはできますし、市販のコンピュータは複雑なシステムですから、そのすべてを完璧に理解しようとするのはむしろ馬鹿げたことだろうと思います。しかし、本質的にはどのような原理で動作しているのかというエッセンスだけでも理解しておけば、嫌々ながら使うにしても若干は気分的に楽になるような気がします。

ところが、コンピュータについて解説すると称して書かれた文章などを読んでも、多くの場合に、あまり本質的なことが書かれていないように思います。また、やたらに専門用語が多く登場するにも関わらず用語の定義が不明瞭で、どう考えても理解のしようがないというような例も見受けられます。

私はコンピュータの専門家ではないので、私自身がどれほどコンピュータのことを理解しているかも怪しいものかもしれませんが、今までに実験に使う装置を制御するための電子回路を設計・製作したり、実験データを解析するためのコンピュータ・ソフトウェアを開発してきた経験を通じて学んできたことをふまえて、コンピュータについてなるべく本質的なことがらだけを解説したいと思います。できるだけ専門的な用語を避けて、専門的な用語を使う時にはきちんと説明し、初心者にとってもわかりやすい内容になるように心がけるつもりです。

### 2. 電子回路によるコンピュータ

現在市販されているコンピュータは、自分の知る限りではすべて電子回路を使って作られています。ここで「電子回路」という言葉を使っていますが、より一般的な「電気回路」よりも少し限定した意味合いで使っているということに注意して下さい。

## 2.1 電子回路 electronic circuit

「電子回路」という言葉は、もともと「真空管を使った電気回路」という意味だったのではないかと思います。真空管の一例は二極管と呼ばれるもので、図 2.1 のような構造をしています。これは内部を真空にして密封したガラス管に、2つの電極を挿入して一方の電極を加熱できるようにしてあるものです。この

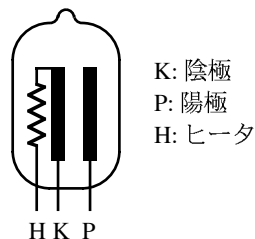


図 2.1 二極管，真空管ダイオード

とき、加熱された電極からは熱によって電子が放出される（この効果をエジソン効果というそうです）ので、この電極にマイナスの電圧をかけてもう一方の電極にプラスの電圧をかければ、マイナスと反発してプラスに引き寄せられる電子が真空中を飛行することによって電流が流れます。もしこれと逆向きの電圧をかけたとしても冷たい電極から放出される電子は少ないので、電流はずっと流れにくくなります。このような真空管は、電極が二つあることから二極管、ダイオードと呼ばれ、「電流の流れやすさが向きによって異なる」という面白い性質を持っています（この性質は面白いと言うだけでなく、非常に多くの目的で有用です）。この二極管で、マイナスの電圧をかける電極の方を「陰極」（カソード cathode）、プラスの方を「陽極」（アノード anode）または「プレート」（plate）と呼びます（化学では陽イオンのことをカチオン cation、陰イオンのことをアニオン anion と呼ぶので混同しないように）。

図 2.2 のように、陰極と陽極の間に接触しないように金属製の網をはさんで3つ目の電極とすると、この電極にかかる電圧によって陰極から陽極へ飛行する電子の流れをコントロールすることができます。この電極をグリッド（格子）と呼びますが、グリッドにたとえばプラスの電圧をかけると、陰極から放出された電子がプラスの方に向かっていこうとするので流れる電流が増え、グリッドにかかる電圧をマイナス側に変化させれば流れる電流が減るというわけです。このような真空管のことを三極管と呼び、電力を増幅する目的で使われます。三極管が電力を増幅する作用を以下のように説明することができます。

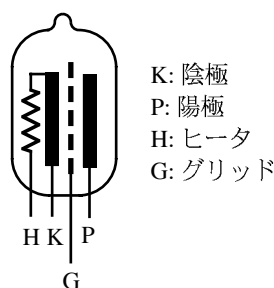


図 2.2 三極管

陰極から放出された電子は、はじめはグリッドの方向に吸い寄せられて行きます。一部の電子はグリッドの網にぶつかってしまうでしょうが、大部分の電子は勢い余ってグリッドの網の目をすり抜けて陽極まで飛んでいきます。そこで、陰極からグリッドへの電子の流れに比べて陰極から陽極への電子の流れの方がずっと大きくなるということになります。つまり、わずかな電流で大きな電流を制御できることになるわけです。

さて、ふつうの金属線ならば、電子が移動しているとしても目には見えないし良くわかりませんが、この「真空管」の中では、金属線がつながっていないのに電流が流れているわけだから、確かにいかにも「電子が真空中を飛行して移動している」ように見えますよね。そんな意味で、このような真空管を利用した電気回路 electric circuit のことを特別に「電子回路」"electronic circuit" と呼ぶことになったのだと思います。

## 2.2 半導体による電子回路 electronic circuit with semiconductors

半導体とよばれる物質を使って、前の節で説明した真空管とほとんど同じはたらきをする素子を作ることができます。たとえば P 型半導体と N 型半導体という 2 種類の半導体をくっつけると、P 型から N 型には電流が流れやすいけれど N 型から P 型へは電流が流れにくいという性質を示します。これは先の話に出てきた真空管のダイオードと同じはたらきをするだけでなく、安価で小型化も容易だという利点を持っています。そこでこのような素子を半導体ダイオードと呼ぶことになりました。でも、このような素子ができてしまうと、真空管のダイオードを使うメリットはほとんどありません。ダイオードというと本来は真空管のダイオードという意味だったのに、今ではダイオードと言えば、この半導体ダイオードのことを意味するようになってしまいました。

また、N 型の二つの半導体の間に薄い P 型の半導体の層をはさんだり、あるいは P 型の二つの半導体の間に薄い N 型の半導体の層をはさんだものをこしらえると、三極管と同じはたらきをするものができます。これをトランジスタと呼び、やはり多くの用途で三極管よりも便利です。

結局、ほとんどの用途で、真空管をダイオードやトランジスタなどの半導体素子に置き換えることができるようになって、真空管を使うメリットはほとんどなくなってしまいました。そこで、半導体素子を使った回路という意味でも電子回路という言葉が使われるようになったのだと思います。けれど、そういう歴史的な背景があることを知らなければ「電子回路」という言葉自体に違和感を覚える人も多いのではないのでしょうか？真空管だと見た目で見ても「電子が飛んでる」ように見えるのですが、半導体だとちょっとイメージしにくいですね。

## 2.3 電子回路を使わないコンピュータ？ computer without electronic circuits ?

真空管と半導体のどちらを使うとしても、いずれにしても「電子回路」を使わないとコンピュータを作ることにはできないのでしょうか？現実的にはそうとも言えるかもしれませんが、コンピュータが動作する原理から考えれば、かならずしも電子回路を使うことは本質的なことではありません。このことは後でももう少し詳しく説明しますが、電磁石で動くスイッチを使って、真空管や半導体をいっさい使わない（つまり「電子回路」ではない）電気回路によるコンピュータを作ることにも原理的には可能です。

もっと発想を飛躍させて、たとえばバルブ（弁）のように水や空気の流れを制限する仕組みを使って「水コンピュータ」や「風コンピュータ」をつくることを想像したら（役には立たないでしょうが）楽しいのではないかと思います。

## 2.4 アナログ・コンピュータ analogue computer

現在市販されているコンピュータは（自分の知る限り）すべてデジタル・コンピュータと呼ばれるものですが、かつて特殊な用途で実際にアナログ・コンピュータと呼ばれるものが作られて使われたこともあるそうです。

デジタル・コンピュータは数値を 2 進数で表現して、1 という数字をたとえば +5 V, 0 という数字を 0 V という電圧に対応させて、原則的にこの 2 種類の電圧しか使いません。2 という数値を表したければ、電線を 2 本使って、"10" という電圧のパターンにします。十進数の 10 という数値を表したければ、最低でも 4 本の電線を使って、"1010" という電圧のパターンにするわけです。

それに対して、アナログ・コンピュータでは、1 つの数値を表すのに 1 本の電線を使って、 $x$  という数値を  $x$  V という電圧で表現します。つまり、1 という数値を 1 V, 2 という数値を 2 V という電圧で表現することになります。

二つの電圧を加えた電圧を出力する電子回路や、二つの電圧の差を出力する電子回路、対数や指数、時間について電圧を積分した値や微分した値を出力する回路の作り方は良く知られています。特定の計算を高速に実行できるアナログ・コンピュータを作ること、あながち非現実的なことではないだろうと思

ます。ただ、この話を詳しくすると少し専門的になりすぎるので、この講義ではデジタル・コンピュータの話だけをしたいと思います。また、今後特に断らない限りは、デジタル・コンピュータという意味でコンピュータという言葉を使うことにします。

## 2.5 論理回路 logical circuit

「論理回路」というと少しおおげさな気がしますが、「あたかも論理を扱っているように見える回路」くらいのことで、実際にはデジタル回路と同じことだと思っても良いだろうと思います。デジタル回路では、たとえば原則的に 0 V と 5 V との 2 種類の電圧（正確には電位と呼ぶべきかもしれませんが）しか使わないのですが、それぞれの電圧の値を「偽」= "false" = 0, 「真」= "true" = 1 という値に対応づければ、あたかも論理を扱っているように見えるわけです。

（デジタル）コンピュータは論理回路の組み合わせで動いています。これが本質的なことであって、トランジスタとかの半導体を使うことは必ずしも本質的なことではありません。ここでは「論理回路」とはどのようなものかという話をします。

### 2.5.1 記号論理とブール代数 symbolic logic and Boolean algebra

後で必要になるので（あまり厳密ではないのですが）記号論理とか論理代数と呼ばれることがらについての話をします。現在使われている方法は G. Boole という人が始めたものが元になっており、ブール代数とも良く呼ばれます。

はじめに、形式的な話をします。まず「0 と 1 の二つの数しかない世界」を考えます。そして次の 3 種類の演算（+, -, ×, ÷ のようなもの）の規則をつくります。

#### 1. NOT 演算 ( $\bar{X}$ ) の規則

$$\begin{aligned}\bar{0} &= 1 \\ \bar{1} &= 0\end{aligned}\tag{1}$$

#### 2. AND 演算 ( $A \wedge B$ ) の規則

$$\begin{aligned}0 \wedge 0 &= 0 \\ 0 \wedge 1 &= 0 \\ 1 \wedge 0 &= 0 \\ 1 \wedge 1 &= 1\end{aligned}$$

#### 3. OR 演算 ( $A \vee B$ ) の規則

$$\begin{aligned}0 \vee 0 &= 0 \\ 0 \vee 1 &= 1 \\ 1 \vee 0 &= 1 \\ 1 \vee 1 &= 1\end{aligned}\tag{2}$$

この、 $\wedge$ ,  $\vee$  という演算は、上の規則以外は +, -, ×, ÷ などの計算とまったく同じように扱えるとします。これを「論理演算」といいます。

どうして論理という名前がついているかということ、このような一連の規則は「論理」というものを単純化して扱うことに対応するからです。

ところで、「AND 演算」 $A \wedge B$  は、0 と 1 の値にしか使わないことを除けば、「かけ算」 $A \times B$  と同じことだということに気がつきませんか？「 $0 \times 0 = 0$ 」「 $0 \times 1 = 0$ 」「 $1 \times 0 = 0$ 」「 $1 \times 1 = 1$ 」ですね。そこで、AND 演算のこと、あるいは AND 演算の結果のことを、論理的なかけ算という意味で「論理積」という言葉で表します。

「OR 演算」 $A \vee B$  は「たし算」 $A + B$  と似ています。ただし、 $1 + 1 = 2$  に対して  $1 \vee 1 = 1$  というところが違ってきます。それでも、論理的な足し算という意味で「論理和」という言葉が使われます。

任意の論理値(つまり 0 か 1 の値)を取る変数を論理変数と呼びます。たとえば「今日が休講だということが正しいかどうか」ということを  $A$  という記号で表して「正しい」ということを「 $A$  は 1 という値を取る」と言い換えて「正しくない」ということを「 $A$  は 0 という値を取る」と言い換えることにします。つまり、今日が休講なら  $A = 1$ 、休講でなければ  $A = 0$  と書くことができます。

また「今日が休講だ」という記述を否定すれば「今日が休講でない」ということになりますから「今日が休講でないということが正しいかどうか」ということが  $\bar{A}$  と表されて、もし今日が休講なら  $A = 1$  だから  $\bar{A} = 0$ 、休講でなければ  $A = 0$  だから  $\bar{A} = 1$  になるということは 1 番目の「NOT の規則」から確かめることができます。

今度は「来週は試験だということが正しいかどうか」ということを  $B$  という記号で表して「今日が休講でなおかつ来週は試験だということが正しいかどうか」ということが  $A \wedge B$  と表されるということになり、また「今日が休講であるか来週が試験であるかのどちらかだということが正しいかどうか」ということ(何が言いたいのかよくわかりませんが)が  $A \vee B$  と表されていることになります。

なんだか、簡単なことをわざと難しくしているように思えなくもないのですが、実際に使う論理演算は、(論理と言う名前がついていますが)必ずしも普通の意味での「論理」と対応づけられなければいけないという必然性はありません。必要なことは、普通の( +, -, ×, ÷ などの)計算の規則ではなく、上にあげた「論理演算の規則」に従うということです。もし頭が混乱してきたら、論理演算の規則に戻って考えるようにしてください。

## 2.5.2 ド・モルガンの法則 De Morgan's law

法則などというとおおげさな気もするのですが、

$$\overline{A \wedge B} = \bar{A} \vee \bar{B} \quad (3)$$

$$\overline{A \vee B} = \bar{A} \wedge \bar{B} \quad (4)$$

という二つの関係のことをド・モルガンの法則と言います。前節の「論理演算の規則」を使えば簡単に導くことができますが、実際に論理演算の中で良く出てくるのでド・モルガンの法則と呼んで使うことにします。ついでに、次の二つの式もド・モルガンの法則と呼ぶことにしてもよいでしょう。

$$A \wedge B = \overline{\overline{A \wedge B}} \quad (5)$$

$$A \vee B = \overline{\overline{A \vee B}} \quad (6)$$

もちろん、これも前節の「論理演算の規則」を使えば簡単に導くことができます。規則から  $\overline{\bar{0}} = 1, \overline{\bar{1}} = 0$  だから、一般的に  $\overline{\bar{A}} = A$  と書けることを確認しておいてください。

この法則に何か意味を持たせたければ、たとえば次のような記述を想像してみたいかがでしょうか？「Q 君がタワケでもないしアホでもないということはない」このとき、私がいいたいの「Q 君はタワケであるかアホであるかのどちらかである」ということだから、式 (6) に対応していますね？

余談ですが「A ならば B である」( $A \rightarrow B$ ) ということが「A でないかまたは B である」( $\bar{A} \vee B$ ) ということと同じことであるということも「Q 君はタワケでなければアホである」という言葉が「Q 君はタワケかアホかのどちらかである」ということと同じ意味であると考えれば大いに納得できますね。

## 2.6 電子回路による論理演算

電子回路によって論理演算を実現するための具体的な方法について説明します。ド・モルガンの法則を使えば、すべての論理演算を NOT 演算と AND 演算の組み合わせ、あるいは NOT 演算と OR 演算の組み合わせだけで表現できることがわかります。ここでは、はじめに NOT 演算を実現する回路、つぎに AND 演算を実現する回路について説明します。

論理値 1 を +5 V、0 を 0 V という電圧で表現することにします。この +5 V という電圧を選んだのは、トランジスタを使う時にはこのくらいの電圧を選んでおくのが都合が良いからで、実際に最近まではほとんどすべての論理回路に +5 V という電圧が使われていました。ただし、最近のコンピュータでは発熱を抑えるために +3.3 V とか +3 V という値が使われるようになってきたようです。

### 2.6.1 NOT 回路

NOT 演算を電子回路で実現するためには、「入力が 0 V なら +5 V を出力し、入力が +5 V なら 0 V を出力する」回路を作ればよいことになります。そのような回路のことを NOT 回路と呼びます。トランジスタを使った NOT 回路は図 2.3 のようなものです。

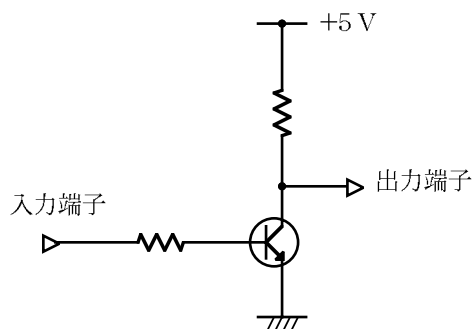


図 2.3 トランジスタを使った NOT 回路。

図 2.3 の中で「横棒に小さな黒丸、横に +5 V と書いてある」記号は、+5 V の電源(のプラス側の端子)に接続することを表している、また「横棒の下にななめの線」の記号はアース(地球)earth とかグラウンド(地面)ground と呼ばれるもので、実際には +5 V の電源のマイナス側の端子に接続することを表しています。「ギザギザの線」は抵抗器 resistor を表している、普通はだいたい  $1\text{ k}\Omega$  くらいの値のものを使います。どうして抵抗器をつなぐのかについては後で説明します。「丸に縦棒と 3 本の線、そのうち一つは外に向いた矢印」の記号は(NPN 型)トランジスタを表しています(図 2.4)。NPN 型のトランジスタは

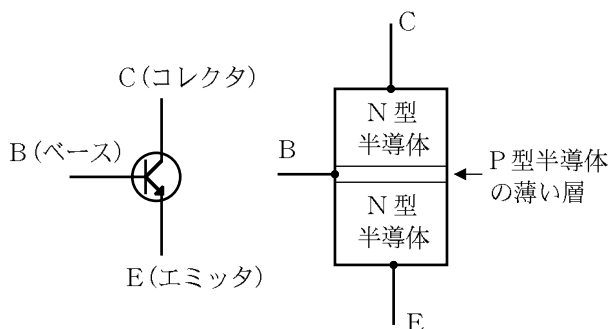


図 2.4 (NPN 型)トランジスタの回路記号と構造。

N 型半導体の間を薄い P 型半導体ではさんだ構造を持っています。トランジスタから出ている 3 本の線にはそれぞれ名前がついていて、ベース base、コレクタ collector、エミッタ emitter と呼びます。ベースは P 型半導体の薄い層に接続されていて、コレクタとエミッタは 両側の N 型半導体に接続されています。

トランジスタがどのように動作するかは、ややわかりにくいのではないかと思います。「ベース B からエミッタ E に少し電流を流すと、コレクタ C からエミッタ E へと電流が流れやすくなる」というのはたらしきをするものであると考えてください。別の言い方をすれば、「ベースからエミッタへ流れる小さい電流で、コレクタからエミッタへ流れる電流を制御することができる」素子ということになります。

図 2.5 に示す「リレー」という素子が、論理回路ではトランジスタと同じはたらきをすると考えれば、わかりやすいのではないかと思います。リレーとは、図で示したように、電磁石でスイッチをオンにしたりオフにしたりするような素子です。

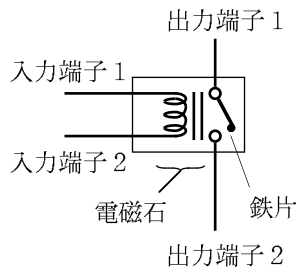


図 2.5 リレーの模式図。スイッチはバネの力で通常 OFF の状態になっている。入力端子 1 から入力端子 2 に電流を流すと、スイッチの鉄片が電磁石に吸い寄せられて ON になる。

リレーを使った NOT 回路を図 2.6 に示します。入力が 0 V のとき、スイッチはバネの力で OFF になっ

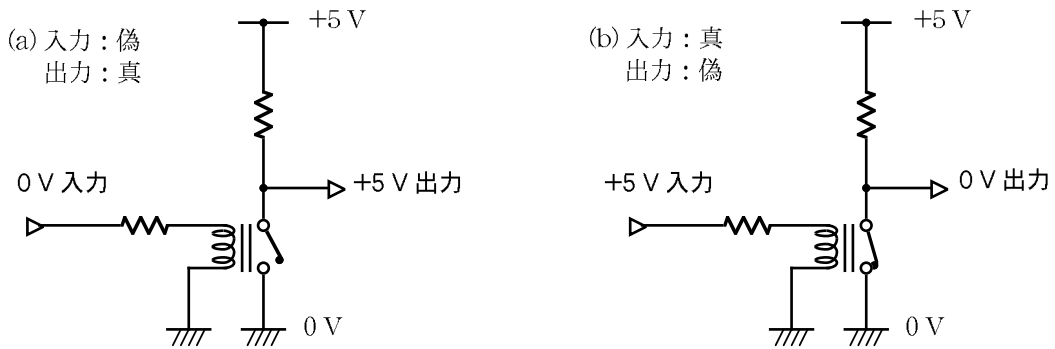


図 2.6 リレーによる NOT 回路。(a) 入力が 0 V のとき、出力は +5 V である。(b) 入力に +5 V をかけると、出力が 0 V になる。

ているので、抵抗を介して +5 V 電源と接続されている出力の電圧は +5 V になります。入力が +5 V のとき、電磁石の力でスイッチが ON になるので、出力はグランドと接続されて 0 V になります。

電磁石に使われているコイルの抵抗はふつう小さいので、直接 +5 V をかけると電流が流れ過ぎてしまい、無駄な電力消費や発熱の原因になり、最悪の場合にはコイルが焼き切れてしまいます。入力側の抵抗は、この電流を制限するはたらきを持ち、電流制限抵抗と呼ばれます。トランジスタを使った NOT 回路でも、入力側と出力側の 2 つの抵抗のはたらきは、リレーの場合とほとんど同じと考えてよいと思います。

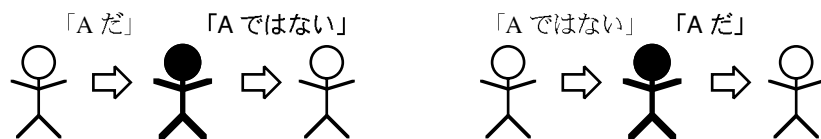


図 2.7 NOT 回路のはたらき。つねに聞いたことと反対のことを言う人と同じことである。

NOT 回路のはたらきは、図 2.7 のように、「つねに聞いたことと反対のことを言う人」とだいたい同じ

ことです。

NOT 回路は情報の劣化を防ぐはたらきも持ちます。これもしレーを使った NOT 回路 ( 図 2.6 ) で考えるとわかりやすいと思います。バネの力に応じて電流制限抵抗の大きさを調節してやれば、たとえばちょうど +2.5 V 以上のときにスイッチが ON になり、+2.5 V 未満の電圧ではスイッチが OFF になるようにすることができるでしょう。つまり、入力が +2.5 V 未満では +5 V を出力し、入力が +2.5 V 以上では 0 V を出力する回路を実現することができます。この場合、図 2.8 のように「つねに聞いたことと反対

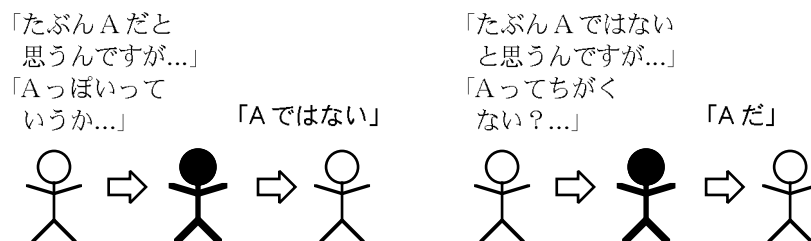


図 2.8 NOT 回路のはたらき (2)。つねに聞いたことと反対のことを強く言い切る。

のことを強く言い切る人」と同じことですね。

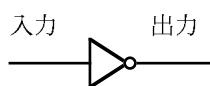


図 2.9 NOT 回路の回路記号

論理回路では、NOT 回路をひとまとまりとして、図 2.9 のような記号であらわします。また、これを「NOT 素子」と呼びます。記号では +5 V とグランドに接続する線を省略します。

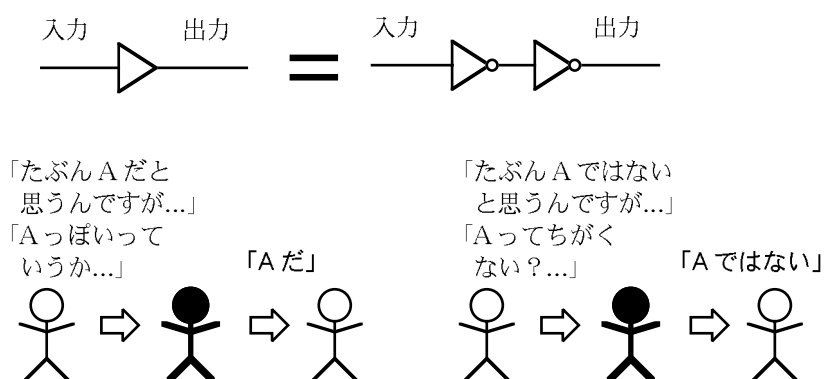


図 2.10 バッファ回路の回路記号とはたらき。「つねに同じ方向の意見を強く言い切る人」と同じことである。

NOT 回路を 2 つ直列に接続すれば「反対の反対は賛成」ですから、図 2.10 のように「つねに同じ方向の意見を強く言い切る人」と同じことになります。このような回路をバッファ回路と呼びます。バッファ回路を使えば、何かの原因で入力がちょうど 0 V か +5 V でなくても出力は必ず 0 V か +5 V になるわけです。論理演算としては何もしていないのと同じことですが、実際に正しく動作する論理回路を製作するためには必要になる場合があります。



## 2.6.2 AND 回路

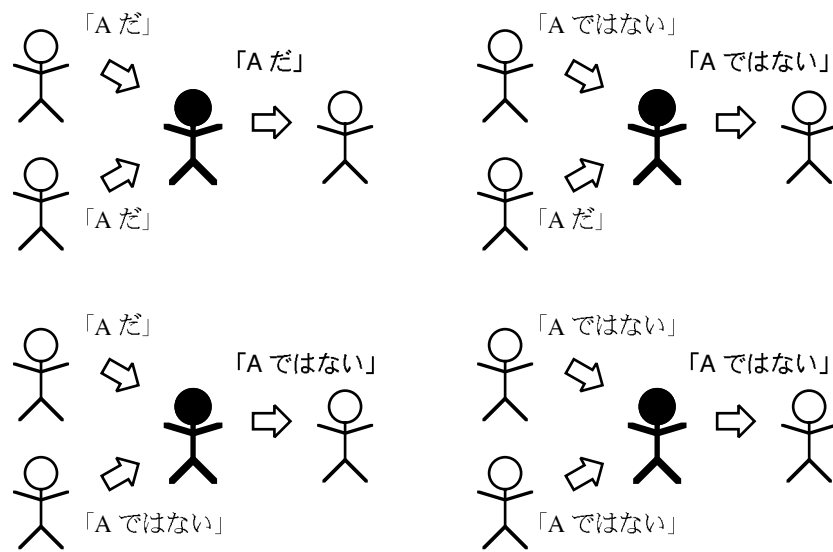


図 2.11 AND 回路のはたらき

AND 演算を実現するためには、図 2.11 のようなはたらきをする回路をつくれればよいことになります。つまり、「2つの入力を持ち、入力の両方が +5 V のときだけ +5 V を出力し、それ以外では 0 V を出力する回路」を作ればよいわけです。

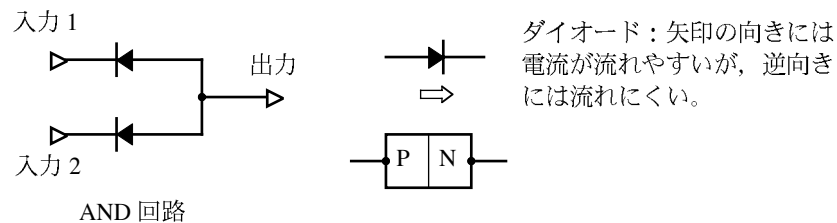


図 2.12 ダイオードによる AND 回路

良く使われる方法は、ダイオードを使って図 2.12 のように接続することです。半導体ダイオードは P 型半導体と N 型半導体を貼り合わせた構造を持ち、P 型半導体の側から N 型半導体の側（順方向）へは電流が流れやすいが、逆向き（逆方向）には電流が流れにくいという性質を持ったものです。

ちょっとわかりにくいと思うので、図 2.13 に具体的な例を示します。順方向に電流を流すときの抵抗が  $2 \Omega$ （抵抗が小さい＝流れやすい）であり、逆方向に電流を流すときの抵抗が  $998 \Omega$ （抵抗が大きい＝流れにくい）であるとして、このとき、入力の電圧の組み合わせによって、出力の電圧が図 2.13 のように変わります。入力の両方とも 0 V のときは出力も 0 V、入力の両方とも +5 V のときは出力も +5 V になるということはいまでもありません。ところが、入力の片方が 0 V でもう一方が +5 V のときには、出力が +0.01 V になって、中間の +2.5 V という値と比べてずっと 0 V に近い値になります。ダイオードの逆方向と順方向の抵抗の比がもっと大きければ、出力電圧はもっと 0 V に近い値になりますし、また、この AND 回路のうしろに、前の節で説明した NOT 回路を接続すれば、すでに説明したように AND 回路の部分は厳密に 0 V を出力しているのと同質的には同じことになります。

ただし、この簡単な AND 回路は、必ずしも図 2.11 のように理想的な動作をしているわけではなく、ど

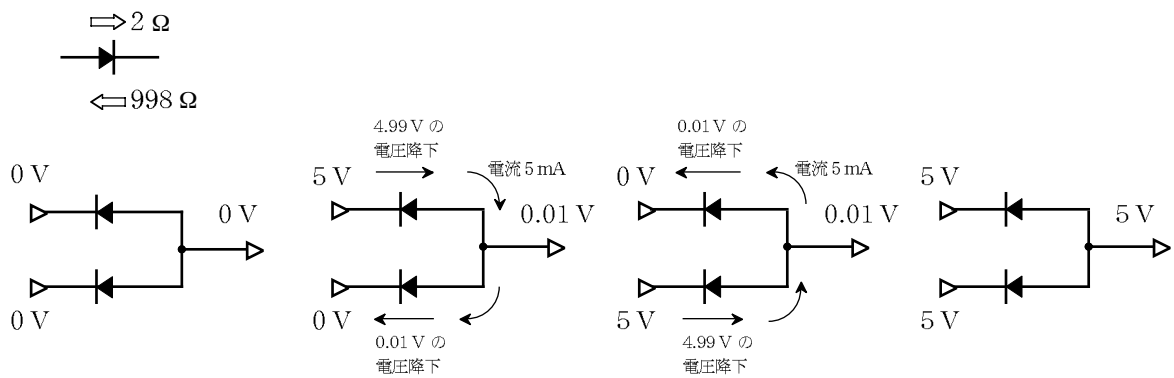


図 2.13 ダイオードによる AND 回路の具体的な例

ちらかというと図 2.14 のような動作をしているということになります。

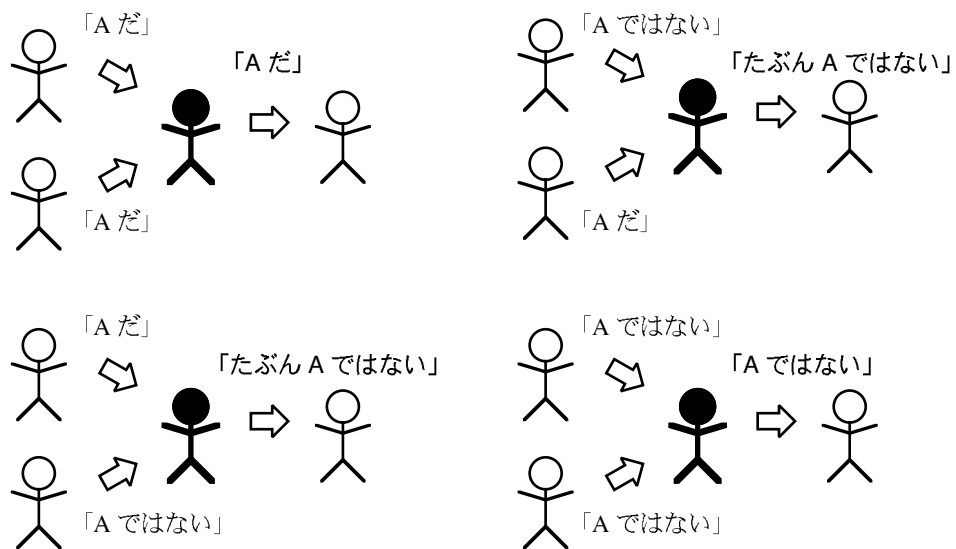


図 2.14 ダイオードによる AND 回路のはたらき