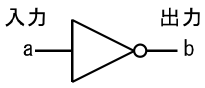
コンピュータ工学 講義プリント(5 月 1 日)

・ディジタル回路のゲート回路と論理式（教科書 55 ページ参照）

ディジタル回路で用いられる基本的な演算素子はゲート回路とも呼ばれる。代表的なゲート回路とその真理値表および論理式を次の図に示す。



回路図

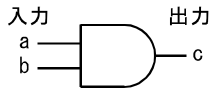
a	b
0	1
1	0

真理値表

$b = \overline{a}$

論理式

(a) NOT 回路



回路図

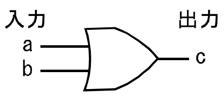
a	b	c
0	0	0
0	1	0
1	0	0
1	1	1

真理値表

$c = a \cdot b$

論理式

(b) AND 回路



回路図

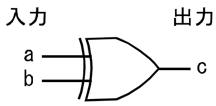
a	b	c
0	0	0
0	1	1
1	0	1
1	1	1

真理値表

$c = a + b$

論理式

(c) OR 回路



回路図

a	b	c
0	0	0
0	1	1
1	0	1
1	1	0

真理値表

$c = \overline{a} \cdot b + a \cdot \overline{b}$

論理式

(d) XOR 回路

図 1、代表的なゲート回路

論理式は、ゲート回路の挙動を数式で表したものである。この様に、0 か 1 しか値を取らない変数を扱う代数の事をブール代数という。

ブール代数の+の記号(論理和)は普段使う足し算の記号(算術和)と異なる意味を持ち、1+1=1 になる事に注意。また、・の記号の方が+の記号より演算の優先順位が高い事にも注意。例えば $a+b\cdot c$ は $a+(b\cdot c)$ と解釈する。

・論理圧縮（回路の最適化）

次のページの図 2(a)のディジタル回路は、図 2(b)の OR 回路と同じ働きをする。この様に、回路の動作を変えないようにゲート回路の数を減らし、消費電力やコストなどの削減を行う事を論理圧縮とか回路の最適化と呼ぶ。

・ブール代数の代表的な定理

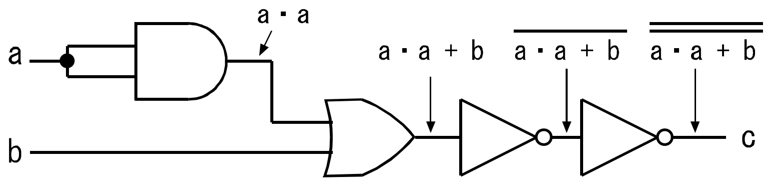
次のページの表 1 に、ブール代数で用いられる代表的な定理を示した。たくさん定理があるが、全て憶える必要はない。ただし、定理の証明はできるようにしておきたい。

例として $a\cdot(\overline{a}+b)=a\cdot b$ を 2 種類の方法で証明してみる。どちらか理解しやすい方法で証明できれば十分である。

例 1、他の定理を使う方法

分配の法則より次の式が成り立つ。

$$a\cdot(\overline{a}+b)=a\cdot\overline{a}+a\cdot b$$



回路図

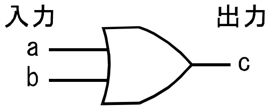
$c = \overline{\overline{a\cdot a+b}}$

論理式

a	b	a・a	a・a + b	$\overline{a\cdot a+b}$	c
0	0	0	0	1	0
0	1	0	1	0	1
1	0	1	1	0	1
1	1	1	1	0	1

真理値表

(a) 論理圧縮前の回路



回路図

a	b	c
0	0	0
0	1	1
1	0	1
1	1	1

真理値表

c = a + b

論理式

(b) 論理圧縮後の回路

図 2、ディジタル回路の最適化（論理圧縮）

表 1、ブール代数の代表的な定理

名称	公式	名称	公式
公理	1+a=1 $0\cdot a=0$	交換の法則	$a+b=b+a$ $a\cdot b=b\cdot a$
恒等の法則	$0+a=a$ $1\cdot a=a$	結合の法則	$a+(b+c)=(a+b)+c$ $a\cdot(b\cdot c)=(a\cdot b)\cdot c$
同一の法則	$a+a=a$ $a\cdot a=a$	分配の法則	$a\cdot(b+c)=a\cdot b+a\cdot c$ $a+(b\cdot c)=(a+b)\cdot(a+c)$
補元の法則	$a+\overline{a}=1$ $a\cdot\overline{a}=0$	吸収の法則	$a\cdot(a+b)=a$ $a+a\cdot b=a$ $a+\overline{a}\cdot b=a+b$ $a\cdot(\overline{a}+b)=a\cdot b$
復元の法則	$\overline{\overline{a}}=a$	ド・モルガンの定理	$\overline{a+b}=\overline{a}\cdot\overline{b}$ $\overline{a\cdot b}=\overline{a}+\overline{b}$

補元の法則より $a \cdot \overline{a} = 0$ なので次の式が成り立つ。

$$a \cdot (\overline{a} + b) = 0 + a \cdot b$$

恒等の法則より $0 + a \cdot b = a \cdot b$ なので、次の式が成立する。

$$a \cdot (\overline{a} + b) = a \cdot b$$

例 2、真理値表を書いて証明する方法(こちらの方が簡単)

表 2 に示す真理値表より、明らかに $a \cdot (\overline{a} + b) = a \cdot b$ が成立する。

表 2、$a \cdot (\overline{a} + b)$ と $a \cdot b$ の真理値表

a	b	$\overline{a}$	$\overline{a} + b$	$a \cdot (\overline{a} + b)$	$a \cdot b$
0	0	1	1	0	0
0	1	1	1	0	0
1	0	0	0	0	0
1	1	0	1	1	1

ブール代数の定理や真理値表を使うと、論理圧縮ができる。$\overline{\overline{a \cdot a + b}} = a + b$ を表 1 の定理を用いて証明し、図 2(a)の回路が図 2(b)の回路に最適化できることを示してみよ。

・真理値表から論理式やディジタル回路を生成する方法

真理値表が与えられれば、機械的に論理式を求める事ができる。例として次のページの図 3(a)のような真理値表の回路を考える。この回路は、3 つの入力 a 、b 、c の中で、多かった入力値を出力 d に出力する回路である。(つまり、多数決回路)

図 3(b)を見れば分かるように、$d = \overline{a} \cdot b \cdot c + a \cdot \overline{b} \cdot c + a \cdot b \cdot \overline{c} + a \cdot b \cdot c$ という論理式になる事が分かる。

さらにこの論理式からディジタル回路を生成すると、図 3(c)のような回路図となる。ただし、この回路は最適な回路ではないので、論理圧縮をする必要がある。

・カルノー図

論理圧縮を、図を使って簡単に行うために、カルノー図と呼ばれる図が使われる事がある。しかし、現在では論理圧縮はコンピュータで自動的に行えるため、手書きでカルノー図を書いて論理圧縮をする機会は減った。よって、ここでは簡単に紹介するのみにし、深くは触れない事にする。

次のページの図 4(a)は AND 回路の真理値表であるが、同じ内容を図 4(b)の様な形式で表した図をカルノー図と呼ぶ。

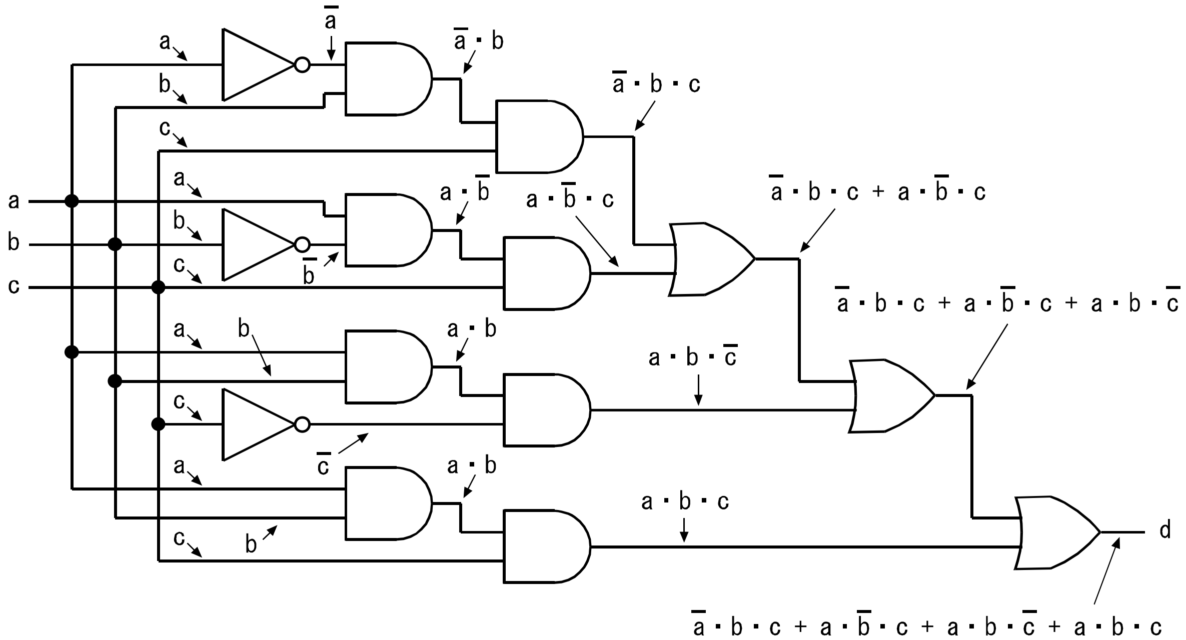
図 3(a)の、多数決回路の真理値表をカルノー図で表現したものを、5 ページの図 5(a)に示す。図 5(a)～(d)より多数決回路の論理式は $d = a \cdot b + b \cdot c + c \cdot a$ となる事が分かる。この論理式より多数決回路を生成すると図 5(e)の様になり、図 3(c)よりはかなり簡略化された回路が得られる事が分かる。(ただし、これでも最適な回路ではない。図 5(f)の様な回路もある)

a	b	c	d	備考
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{a}$・b・cは、この行のみ1になる
1	0	0	0	
1	0	1	1	a・$\overline{b}$・cは、この行のみ1になる
1	1	0	1	a・b・$\overline{c}$は、この行のみ1になる
1	1	1	1	a・b・cは、この行のみ1になる

(a) 多数決回路の真理値表

a	b	c	$\overline{a}$・b・c	a・$\overline{b}$・c	a・b・$\overline{c}$	a・b・c	$\overline{a}$・b・c＋a・$\overline{b}$・c＋a・b・$\overline{c}$＋a・b・c
0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	1	1	0	0	0	1
1	0	0	0	0	0	0	0
1	0	1	0	1	0	0	1
1	1	0	0	0	1	0	1
1	1	1	0	0	0	1	1

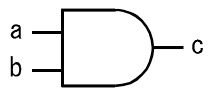
(b) $\overline{a}$・b・c＋a・$\overline{b}$・c＋a・b・$\overline{c}$＋a・b・cの真理値表



(c) 真理値表から生成した多数決回路の回路図

図3、真理値表からのディジタル回路の合成

a	b	c
0	0	0
0	1	0
1	0	0
1	1	1



(a) AND回路の真理値表

a \ b	0	1
0	0	0
1	0	1

cの値

(b) AND回路のカルノー図

図4、AND回路の真理値表とカルノー図

数字の並びに注意

a b \ c	0	1
0 0	0	0
0 1	0	1
1 1	1	1
1 0	0	1

(a) 多数決回路のカルノー図

a b \ c	0	1
0 0	0	0
0 1	0	0
1 1	1	1
1 0	0	0

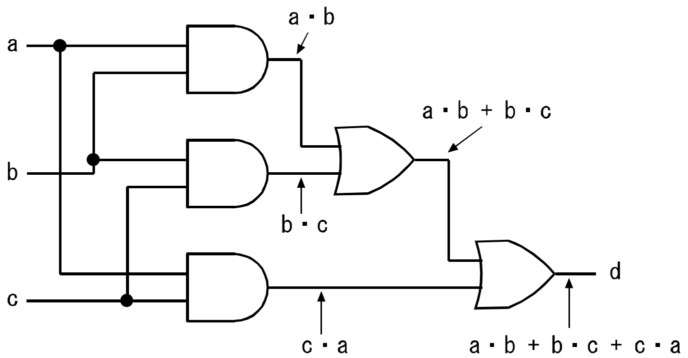
(b) a・b のカルノー図

a b \ c	0	1
0 0	0	0
0 1	0	1
1 1	0	1
1 0	0	0

(c) b・c のカルノー図

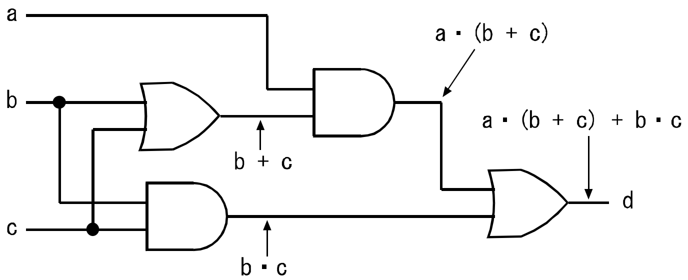
a b \ c	0	1
0 0	0	0
0 1	0	0
1 1	0	1
1 0	0	1

(d) c・a のカルノー図



(e) d = a・b + b・c + c・a の論理式から生成した多数決回路

d = a・b + b・c + c・a = a・b + a・c + b・c = a・(b + c) + b・c
と変形できる事を利用して多数決回路を生成すると、次の回路になる。



(f) d = a・(b + c) + b・c の論理式から生成した多数決回路

図 5、カルノー図を用いた多数決回路の論理圧縮