論理関数の簡単化

$$T(f) = \cup_{p_j \in Q} T(p_j) \qquad Q \subseteq P_j$$

(a) $|Q|$ が最小

(b) 式全体で使われるリテラルの個数が最小
(|Q|が同じ組み合わせが複数あったら)

- カルノー法
  - ３～６変数に適用可能
  - 分かりやすいけど、見落としなどミスもある
- クワイン・マクラスキ法
  - 変数が多くても大丈夫
  - プログラムしやすい
- コンセンサス法など他にも多数の方法がある

クワイン・マクラスキ法のための準備

- **部分積項の系列表現**
  - 部分積項を「0, 1, -」で書く書き方
  - 肯定リテラルを１、否定リテラルを０、無い変数を−で書く
    - 例：５変数　$x_1\overline{x}_2x_4$ → 10−1−
- **併合**：１文字違う系列表現ペアを併せて違うところを−に置き換え
  - $\overline{x}_1\overline{x}_2x_3\overline{x}_4\overline{x}_5$ $\rightarrow$ 00100
  - $\overline{x}_1\overline{x}_2x_3\overline{x}_4x_5$ $\rightarrow$ 00101
  - 併合後：　0010−　$\rightarrow$　$\overline{x}_1\overline{x}_2x_3\overline{x}_4$
- 「0010−」と「0110−」を併合すると「0−10−」となる
- 併合の結果は部分積項よりリテラルが減っている

クワイン・マクラスキ法とは「併合可能なペア」を系統的に探して、順に併合し、リテラルを減らして簡単にしていく方法である

QM法前半

1. 真理値表（or 真入力ベクトル）から対応する積項系列表現を作る
2. 積項系列表現の１の個数毎に並べ替える（k表）
3. 併合可能なペアを捜す
    - 併合可能なものは１文字違いなので、１の個数が１違う所を見る
4. 併合したペアの積項系列表現を記し、併合したペアをチェック√
5. 併合可能なペアがなくなるか全てチェックされるまで繰り返す
    - k++して、ステップ２に戻るか、終了する

x1	x2	x3	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

0 表

001 √
100 √

011 √
110 √

111 √

1 表

0−1
1−0

−11
11-

さらに併合可能なペア無し

この段階でチェックされずに残っている積項が主項に対応する

全てチェックしたのでk+1表に移行

QM法後半：主項表

6. 最終的に残った積項（主項）から主項表を作る
7. $m_i \leq p_j$ となる組み合わせを主項表から探す（×）
8. 縦に見て、×が１つの列を探す→×がある所が必須主項に対応
9. 必須主項で覆われている最小項を消す
10. 残った最小項があれば、残りの主項からリテラルが少ないものを採用して、全ての最小項が覆われる組み合わせを探す
    - 必須主項と採用した主項の和を取り最簡形を得る

主項表

主項(p_j) ↓	$\overline{x}_1\overline{x}_2x_3$	$\overline{x}_1x_2x_3$	$x_1x_2\overline{x}_3$	$x_1x_2x_3$	$x_1\overline{x}_2\overline{x}_3$	←最小項m_i
$\overline{x}_1x_3$	×	×				
$x_1\overline{x}_3$			×		×	
x_2x_3		×		×		
x_1x_2			×	×		

$$f = \overline{x}_1x_3 + x_1\overline{x}_3 + x_1x_2 \quad \text{or} \quad f = \overline{x}_1x_3 + x_1\overline{x}_3 + x_2x_3$$

もう少し複雑な例

$$T(f_2) = \{00110, 00100, 01100, 11100, 10010, 10100, 00011, 00111, 00101, 01001, \\ 01011, 01101, 11111, 11101, 10011, 10111, 10101\}$$

[0]			[1]			[2]			[3]		
a.	00100	√	a-c.	0010-	√	ac-di.	001--	②	1-4.	--10-	①
b.	00011	√	a-d.	001-0	√	ac-fk.	0-10-	1√	2-5.	--10-	:1-4
c.	00101	√	a-f.	0-100	√	ac-hm.	-010-	2√	3-6.	--10-	:1-4
d.	00110	√	a-h.	-0100	√	ad-ci.	001--	:acdi			
e.	01001	√	b-i.	00-11	√	af-ck.	0-10-	:acfk			
f.	01100	√	b-j.	0-011	⑥	af-hn.	--100	3√			
g.	10010	√	b-l.	-0011	√	ah-cm.	-010-	:achm			
h.	10100	√	c-i.	001-1	√	ah-fn.	--100	:afhn			
i.	00111	√	c-k.	0-101	√	bi-lo.	-0-11	③			
j.	01011	√	c-m.	-0101	√	bl-io.	-0-11	:bilo			
k.	01101	√	d-i.	0011-	√	ci-mo.	-01-1	④			
l.	10011	√	e-j.	010-1	⑦	ck-mp.	--101	6√			
m.	10101	√	e-k.	01-01	⑧	cm-io.	-01-1	:cimo			
n.	11100	√	f-k.	0110-	√	cm-kp.	--101	:ckmp			
o.	10111	√	f-n.	-1100	√	fk-np.	-110-	5√			
p.	11101	√	g-l.	1001-	⑨	fn-kp.	-110-	:fknp			
q.	11111	√	h-m.	1010-	√	hm-np.	1-10-	4√			
			h-n.	1-100	√	hn-mp.	1-10-	:hmnp			
			i-o.	-0111	√	mo-pq.	1-1-1	⑤			
			k-p.	-1101	√	mp-oq.	1-1-1	:mopq			
			l-o.	10-11	√						
			m-o.	101-1	√						
			m-p.	1-101	√						
			n-p.	1110-	√						
			o-q.	1-111	√						
			p-q.	111-1	√						

表 3.7　クワイン - マクラスキ法の適用例

例の続き

		a.	b.	c.	d.	e.	f.	g.	h.	i.	j.	k.	l.	m.	n.	o.	p.	q.	
① --10-	:acfk-hmnp	x		x			x		x			x		x	x		x		必須 1-fhn
② 001--	:acdi	x		x	x					x									必須 2-d
③ -0-11	:bilo		x							x			x			x			
④ -01-1	:cimo			x						x				x		x			
⑤ 1-1-1	:mopq													x		x	x	x	必須 5-q
⑥ 0-011	:bj		x								x								
⑦ 010-1	:ej					x					x								
⑧ 01-01	:ek					x						x							
⑨ 1001-	:gl							x					x						必須 9-g
被覆数		2	2	3	1	2	1	1	1	3	2	2	2	3	1	3	2	1	
被覆 by 必須		1,2	↑	1,2	2	↑	1	9	1	2	↑	1	9	1,5	1	5	1,5	5	

表 3.8　主項表

10.残った最小項があれば、残りの主項からリテラルが少ないものを採用して、全ての最小項が覆われる組み合わせを探す

$$(U_3+U_6)(U_7+U_8)(U_6+U_7)=U_3U_7+U_6U_7+U_6U_8$$

主項関数：Ui i番目の主項を使う

今の場合はどの組み合わせも積項の数が２つなので、リテラルの少ないものを選ぶ

$|p_3|=3, |p_6|=4, |p_7|=4, |p_8|=4$ だから、U_3U_7 の組み合わせが最も短い

$$f_2=x_3\overline{x}_4+\overline{x}_1\overline{x}_2x_3+x_1x_3x_5+x_1\overline{x}_2\overline{x}_3x_4+\overline{x}_2x_4x_5+\overline{x}_1x_2\overline{x}_3x_5$$

ここまでのまとめ：論理関数の簡単化

$$T(f) = \cup_{p_j \in Q} T(p_j) \qquad Q \subseteq P_j$$

(a) $|Q|$ が最小

(b) 式全体で使われるリテラルの個数が最小
(|Q|が同じ組み合わせが複数あったら)

- カルノー法
  - ３～６変数に適用可能
  - 分かりやすいけど、<u>見落としなどミスもある</u>
- クワイン・マクラスキ法
  - 変数が多くても大丈夫。ただし、n2^n程度のメモリが必要なので大きなｎでは無理
  - プログラムしやすい

論理式

- 形式的定義（3回目でやった）

1. 0,1, 変数は論理式である

2. E, F が論理式の時、$\overline{E}, (E \cdot F), (E + F), (E \oplus F)$ は論理式である

3. 以上の（1）、（2）だけでできるものが論理式である

$$\boxed{\text{基本演算子}\rightarrow\text{論理式の性質}\rightarrow\text{式の変形操作}}$$

- ここでは逆の見方をする（**代数系**）

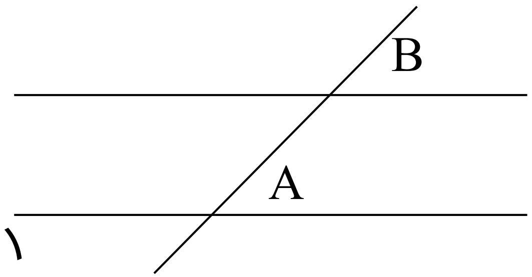
$$\boxed{\text{式の変形操作}\rightarrow\text{論理式の性質}\rightarrow\text{論理演算子}}$$

- 代数系での定義の流れ

1. 演算子と元の集合を指定する
2. 公理系を定める（この公理系が演算子の意味を定める）

公理とは？

- 他の命題を導くための前提
    - 真実である必要はない
- 公理主義数学
    - 公理から全てを導き、全体として矛盾のない体系を作る
    - 例：ユークリッド幾何学の第5公理
        - 平行線の錯角は等しい(A=B)
        - 平面では正しいが球面では成立しない
            - 非ユークリッド幾何学



モデル

- モデルとは、抽象代数において対象集合を特定し、演算子を公理系を満たす関数として特定したもの
  - 例１： $< \mathcal{B};\cdot,+;1,0 >$ →今までやってきた論理関数
  - 例２：命題論理
    - 知識を真偽の決まる文として表現：「１は奇数である」
  - 例３：集合論
- ブール代数の公理系を満たすモデルは沢山ある

公理系の例：ブール代数の公理系

- 演算子と元が具体的に何であるかを特定しない抽象代数

【ブール代数の公理系】† $(x, y, z \in \mathcal{B})$

(1)	単位元	$x \cdot 1 = x$,	$x + 1 = 1$.
(2)	零元	$x \cdot 0 = 0$,	$x + 0 = x$.
(3)	べき等律	$x \cdot x = x$,	$x + x = x$.
(4)	交換律	$x \cdot y = y \cdot x$,	$x + y = y + x$.
(5)	結合律	$x \cdot (y \cdot z) = (x \cdot y) \cdot z$,	$x + (y + z) = (x + y) + z$.
(6)	吸収律	$x \cdot (x + y) = x$,	$x + (x \cdot y) = x$.
(7)	分配律	$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$,	$x + (y \cdot z) = (x + y) \cdot (x + z)$.
(8)	相補律	$x \cdot \overline{x} = 0$,	$x + \overline{x} = 1$.
(9)	二重否定	$\overline{\overline{x}} = x$.	
(10)	ド・モルガン律	$\overline{x \cdot y} = \overline{x} + \overline{y}$,	$\overline{x + y} = \overline{x} \cdot \overline{y}$.

演算子が公理系から定義されている点に注意

例：演算子 $\overline{}$ が相補律から定義されている

ブール代数の公理系の特徴

- 冗長：他の公理から導ける公理がある
  - 例：　(6) 吸収律　　$x\cdot(x+y)=x$

$$x\cdot(x+y)=xx+xy=x(1+y)=x$$

（7）分配律　　（3）べき等律　（1）単位元

- 双対性：・と +, 0と1を入れ替えると対になる式が公理に含まれる

【ブール代数の公理系】† $(x,y,z\in\mathcal{B})$

(1)	単位元	$x\cdot 1=x$,		$x+1=1$.
(2)	零元	$x\cdot 0=0$,		$x+0=x$.
(3)	べき等律	$x\cdot x=x$,	$\longleftrightarrow$	$x+x=x$.
(4)	交換律	$x\cdot y=y\cdot x$,	$\longleftrightarrow$	$x+y=y+x$.
(5)	結合律	$x\cdot(y\cdot z)=(x\cdot y)\cdot z$,	$\longleftrightarrow$	$x+(y+z)=(x+y)+z$.
(6)	吸収律	$x\cdot(x+y)=x$,	$\longleftrightarrow$	$x+(x\cdot y)=x$.
(7)	分配律	$x\cdot(y+z)=(x\cdot y)+(x\cdot z)$,	$\longleftrightarrow$	$x+(y\cdot z)=(x+y)\cdot(x+z)$.
(8)	相補律	$x\cdot\overline{x}=0$,	$\longleftrightarrow$	$x+\overline{x}=1$.
(9)	二重否定	$\overline{\overline{x}}=x$.		
(10)	ド・モルガン律	$\overline{x\cdot y}=\overline{x}+\overline{y}$,	$\longleftrightarrow$	$\overline{x+y}=\overline{x}\cdot\overline{y}$.

$\longleftrightarrow$ 双対（dual）な式

非冗長な公理系：ハンチントンの公理系

【ハンチントンの公理系】　($\mathcal{B}$ はこの公理系で定義される要素の集合である.)

(1) (非自明性)	$x \neq y$ となる $x, y \in \mathcal{B}$が少なくとも 1 組存在する.		
(2) (演算子)	$x, y \in \mathcal{B}$ならば，	$x \cdot y \in \mathcal{B}$，	$x + y \in \mathcal{B}$.
(3) 交換律	$x \cdot y = y \cdot x$,		$x + y = y + x$.
(4) 分配律	$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$,		$x + (y \cdot z) = (x + y) \cdot (x + z)$.
(5) 単位元・零元	$x \cdot 1 = x$,	$x + 0 = x$	となる要素 1, 0 が $\mathcal{B}$ に存在する.
(6) 相補律	$x \cdot \overline{x} = 0$,	$x + \overline{x} = 1$	となる$\overline{x} \in \mathcal{B}$が存在する.

- 冗長でない
- ブール代数の公理系はハンチントンの公理系から導くことが出来る
  - 結合律すら証明の対象：(5) 結合律　$x \cdot (y \cdot z) = (x \cdot y) \cdot z$,　$x + (y + z) = (x + y) + z$.
  - 証明は計算機学入門108ページ

参考：ANDとXORの公理系

- ORの代わりにXOR
  - $x + y = x \oplus y \oplus xy$ で置き換え可能

【可換環・体の公理系】† （x, y, zは対象集合の元である.）

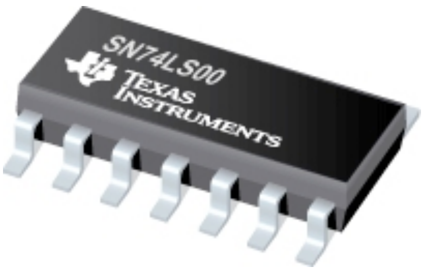
(R1)	交換律	$x \oplus y = y \oplus x$,	$x \cdot y = y \cdot x$.
(R2)	結合律	$x \oplus (y \oplus z) = (x \oplus y) \oplus z$,	$x \cdot (y \cdot z) = (x \cdot y) \cdot z$.
(R3)	分配律	$x \cdot (y \oplus z) = (x \cdot y) \oplus (x \cdot z)$,	$(y \oplus z) \cdot x = (y \cdot x) \oplus (z \cdot x)$.
(R4)	零元・単位元	$x \oplus 0 = x$,	$x \cdot 1 = x$.
(R5)	逆元（加法）	$\forall x$; $x \oplus y = 0$ を満たす解 $y \ (= (-x))$ が存在する.	
(F5)	逆元（乗法）	$\forall x \neq 0$; $x \cdot y = 1$ を満たす解 $y \ (= x^{-1})$ が存在する.	

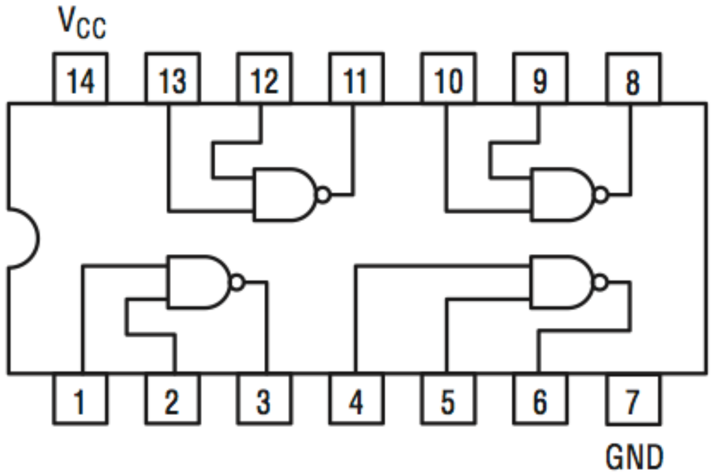
完全系（万能な系）

- 任意の論理関数は{NOT, AND, OR}で書く事ができた
- {NOT, AND, OR}は完全系（万能な系）であると呼ぶ
- **完全系**：すべての論理関数を表現できる代数系
  - 以下の系を始め多くの完全系がある
    - {NAND}
    - {NOR}
    - {EXOR, AND, 1}
    - {P, M, 1, 0} (P: パリティ関数、M: 多数決関数)
- **極小完全系**：それ以上演算子を減らせない系
  - 例：＋は $x+y=\bar{\bar{x}}+\bar{\bar{y}}=\overline{\bar{x}\cdot\bar{y}}$ によりNOTとANDに置き換え可能なので、{NOT, AND}も万能で、これ以上は減らせないので、極小完全系。
  - 同様に{OR, NOT}、{NOR}、{NAND}なども極小完全系

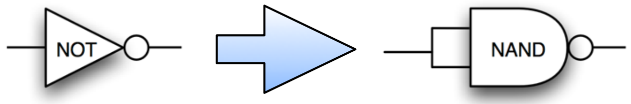
NANDの完全性

- SN74LS00（型番的に一番基本的な回路）
  - NANDが4個入ったIC

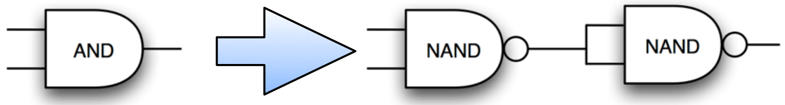




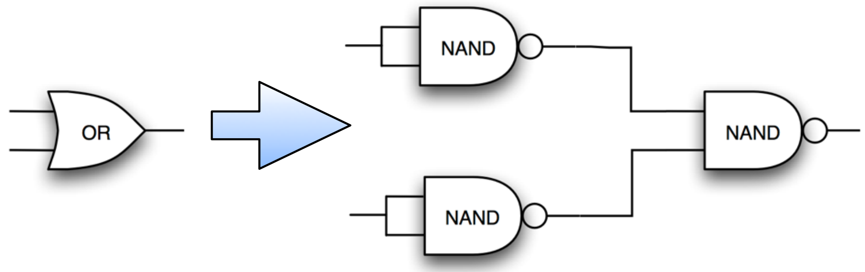
$$\bar{x} = \overline{x \cdot x}$$



$$x \cdot y = \overline{\overline{x \cdot y}}$$



$$x + y = \bar{\bar{x}} + \bar{\bar{y}} = \overline{\bar{x} \cdot \bar{y}}$$



同族関数

- 変数の交換、変数の否定、関数値の否定を施して出来る新しい論理関数は、リテラルが異なっているだけで形は同一であると考え、**同族関数**と呼ぶ

例： $x \cdot y \rightarrow \overline{x} \cdot y \rightarrow x \cdot \overline{y} \rightarrow \overline{x} \cdot \overline{y}$

$\rightarrow$ （関数の否定） $\rightarrow x + y \rightarrow \overline{x} + y \rightarrow x + \overline{y} \rightarrow \overline{x} + \overline{y}$

  - 10個の2変数論理関数のうち上記の８つは全て同族である
    - 残り2つは互いに同族

$$XOR = x\overline{y} + \overline{x}y \rightarrow EQUI = \overline{xy} + xy$$

- 同様に論理関数の諸性質が様々なテキストにまとめられています（例えば、計算気学入門p91-p97）
  - 期末試験には出しませんが、自習しておくこと