$\overline{X_i}$ を入れ替え，演算記号の・（AND）と＋（OR）を入れ替えて，

$$\overline{L} = f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, +, \cdot\right)$$

と形式的に表せる．このとき，次が成り立つ．

$$\begin{aligned} & f\left(X_1, \overline{X_1}, X_2, \overline{X_2}, \ldots, X_n, \overline{X_n}, \cdot, +\right) \\ & \quad = \overline{f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, +, \cdot\right)} \end{aligned} \tag{1.38}$$

証明　2 変数から始めて，数学的帰納法によって n 変数に拡張する．（省略）

定義 1.6（双対関数）　ド・モルガンの定理（定理 1.16）における n 変数論理関数 f に対して，すべての変数の否定（NOT）をとり，かつ，論理関数全体に NOT を施して得る n 変数論理関数を**双対関数**という．本書では，f の双対関数を f^d と表記する．すなわち，形式的には，

$$f^d = \overline{f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, \cdot, +\right)}$$

と表せる．

注意　双対な論理式や双対関数を得る操作では，演算記号の AND(“·”) と OR(“+”) は入れ替えない．この点がド・モルガンの定理と異なる．すなわち，L^d は $\overline{L}$ ではない．

ここで，ド・モルガンの定理（定理 1.16）の式 1.38によって，

$$\begin{aligned} f^d &= \overline{f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, \cdot, +\right)} \\ &= f\left(X_1, \overline{X_1}, X_2, \overline{X_2}, \ldots, X_n, \overline{X_n}, +, \cdot\right) \end{aligned}$$

である．これを f と比較すると,「論理関数 f の**双対関数** f^d は，f の AND($\cdot$) と OR($+$) を入れ替えて得る」ことが分かる．

定義 1.7（自己双対関数）　n 変数論理関数 f とその双対関数 f^d が同値である（等しい）とき，すなわち，

$$f = f^d$$

ならば，f を**自己双対関数**という．

奇数個（1 個か 3 個）のものの論理和.

［問題 2.2］ すべての最小項の包含条件（定理 2.7）U は次の通り.

$$\begin{aligned} U &= p\,t\,(p+q)\,(r+t)\,(q+s)\,(r+s) = p\,t\,(q+s)\,(r+s) \\ &= p\,t\,(s+q\,r) = p\,s\,t + p\,q\,r\,t \end{aligned}$$

したがって，最小積和形 f_m は次の通り.

$$f_m = p + s + t$$

［問題 2.3］ 最小積和形は次の通り.

$$\overline{A}\,\overline{B}\,C + \overline{B}\,\overline{D} + C\,\overline{D}$$

ファクタリングは，$\overline{B}$，C あるいは $\overline{D}$ をくくり出す形の 3 通りある．いずれもリテラル総数は “6” である．たとえば，$\overline{D}$ をくくり出して多段最小化した論理式は次の通り.

$$\overline{A}\,\overline{B}\,C + \overline{D}\,(\overline{B} + C)$$

［問題 2.4］ AND/OR 回路を合成してから，NAND 回路にする．回路図は図 4.9.

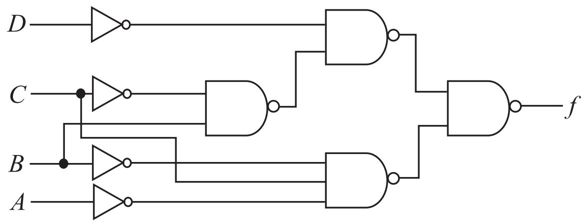

図 4.9　NAND 回路（問題 2.4 の解）

［問題 2.5］ AND/OR 回路を合成してから，NAND 回路に変換すると図 4.10.

［問題 2.6］ 最小積和形は f と同じ．ドントケアがある場合は次の通り.

$$\overline{B}\,\overline{C} + A\,D + A\,C$$

公理 2（NOT）　“0” の NOT は “1”，“1” の NOT は “0” である．

$$\overline{0} = 1 \tag{1.1}$$

$$\overline{1} = 0 \tag{1.2}$$

公理 3（AND）　“1” どうしの AND だけが “1” であり，外（ほか）の 3 通りの組み合わせの AND は “0” である．

$$0 \cdot 0 = 0 \tag{1.3}$$

$$0 \cdot 1 = 0 \tag{1.4}$$

$$1 \cdot 0 = 0 \tag{1.5}$$

$$1 \cdot 1 = 1 \tag{1.6}$$

公理 4（OR）　“0” どうしの OR だけが “0” であり，外の 3 通りの組み合わせの OR は “1” である．

$$0 + 0 = 0 \tag{1.7}$$

$$0 + 1 = 1 \tag{1.8}$$

$$1 + 0 = 1 \tag{1.9}$$

$$1 + 1 = 1 \tag{1.10}$$

1.2.2　論理関係を表す数式

論理代数では，論理や論理どうしの関係を数式によって表現する．

[1]　論理関係と論理式

3 種類の論理演算を使うと様々な論理の関係が数式によって表現できる．論理の関係を**論理関係**，論理関係を表す数式を**論理式**，とそれぞれいう．

例　題

1.1　「『A である』かつ『B でない』，または,『C でない』か『D である』」という論理関係を論理式で表しなさい．

［解］

$$A \cdot \overline{B} + (\overline{C} + D)$$

論理式は論理変数あるいは論理定数（“0” か “1” の論理値）と 3 種類の論理演算記号を用いて記述できる．論理式中のすべての論理変数に論理値を代入すれば論理式そのものの値（論理値）も一意に決まる．

論理値が等しい論理式は**同値**であり，その関係は算術演算と同様に “=” によって表せる．左辺と右辺が “=” によって結ばれた論理式がある場合,「その論理式は成立している」という．

例　題

1.2　例題 1.1 の論理式が論理値として「“1” をとる」場合，および，論理関係「『A である』かつ『D でない』」と同値でない場合，との 2 つの論理関係を論理式で表しなさい．

［解］

$$A \cdot \overline{B} + (\overline{C} + D) = 1$$

$$A \cdot \overline{B} + (\overline{C} + D) \neq A \cdot \overline{D}$$

---

［2］　基本論理演算の演算順位

3 種類の基本論理演算である NOT, AND, OR 間に演算順位を与える．

> **定義 1.4（基本論理演算順位）**　3 種類の基本論理演算の優先順位は，高い方から低い方へ，次の順とする．
>
> NOT(“$\overline{\quad}$”) → AND(“·”) → OR(“+”)

この暗黙の演算順位を変更したいときは，算術演算と同様にカッコ（ ）によって明示する．最内側のカッコの演算が最優先され，外側へ行くにしたがって優先順位が低くなる．同一カッコ内では定義 1.4 の演算順位にしたがう．

$\overline{X_i}$ を入れ替え，演算記号の・（AND）と＋（OR）を入れ替えて，

$$\overline{L} = f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, +, \cdot\right)$$

と形式的に表せる．このとき，次が成り立つ．

$$\begin{aligned} & f\left(X_1, \overline{X_1}, X_2, \overline{X_2}, \ldots, X_n, \overline{X_n}, \cdot, +\right) \\ & = \overline{f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, +, \cdot\right)} \end{aligned} \tag{1.38}$$

証明　2 変数から始めて，数学的帰納法によって n 変数に拡張する．（省略）

定義 1.6（双対関数）　ド・モルガンの定理（定理 1.16）における n 変数論理関数 f に対して，すべての変数の否定（NOT）をとり，かつ，論理関数全体に NOT を施して得る n 変数論理関数を**双対関数**という．本書では，f の双対関数を f^d と表記する．すなわち，形式的には，

$$f^d = \overline{f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, \cdot, +\right)}$$

と表せる．

ここで，ド・モルガンの定理（定理 1.16）の式 1.38によって，

$$\begin{aligned} f^d &= \overline{f\left(\overline{X_1}, X_1, \overline{X_2}, X_2, \ldots, \overline{X_n}, X_n, \cdot, +\right)} \\ &= f\left(X_1, \overline{X_1}, X_2, \overline{X_2}, \ldots, X_n, \overline{X_n}, +, \cdot\right) \end{aligned}$$

である．これを f と比較すると，「論理関数 f の**双対関数** f^d は，f の AND$(\cdot)$ と OR$(+)$ を入れ替えて得る」ことが分かる．

注意　厳密には，双対関数を得る操作では，定義 1.5（11 ページ）による双対な論理式を得る操作と同じく，論理演算記号の AND(“·”) と OR(“+”) の入れ替えの外に，論理定数の “0” と “1” の入れ替えも適用する．双対を得るこれらの操作では，論理変数の否定はとらない点がド・モルガンの定理と異なる．

定義 1.7（自己双対関数）　n 変数論理関数 f とその双対関数 f^d が同値である（等しい）とき，すなわち，

$$f = f^d$$

ならば，f を**自己双対関数**という．

まえがき

本書では,「コンピュータサイエンス」を情報科学，情報工学，計算機科学，計算機工学などの総称として用いている．

そして，本書は，大学学部，高等専門学校，専修学校のコンピュータサイエンス系学科における「論理回路」と「論理設計」の教科書として書き下ろした．

その内容は,（社）情報処理学会が策定した「大学の理工系学部情報系学科のためのコンピュータサイエンス教育カリキュラム J97」の $\boxed{\text{U-1}}$ **論理回路**；$\boxed{\text{U-6}}$ **論理設計**；の 2 科目に準拠している．

本書では，コンピュータサイエンスを支える論理代数とそのハードウェアによる実現である論理回路との関係について，電気に関する専門知識がなくても理解できるように，解き明かしている．また，古典的な知識や理論だけではなく，最新の理論や実用的な手法についても平易に解説している．各所で，コンピュータハードウェアの基本原理である「論理回路」を実例として紹介し，理論と実際との関連に興味をつなげるようにしている．

J97 では，この分野の講義を，① **論理回路**：数学的な概念（ソフトウェア）による組み合わせ回路や順序回路といった論理回路（ハードウェア）の実現；② **論理設計**：論理回路の効率の良い設計手法の理論と実際；の 2 科目に分けて学習することを提案している．

本書の構成では，①には，1 章，2 章の 2.1 と 2.2 節，3 章の 3.1，3.2 節が，②には，2 章の 2.3〜2.5 節，3 章の 3.3〜3.5 節，4 章が，それぞれ対応している．また，J97 では，それぞれの講義科目に，1.5 時間× 15 回（週 1 回で半年）をあてることも推奨している．この分類を参考にして，各機関での実現可能性に合わせて，本書を教科書として活用してもらえば幸いである．

本書についてのコメントや意見は著者（shibayam@kit.jp）への E メールで

送られたい．適切なコメント等については，今後の増刷や改訂時に対応する．それらの対応によるあるいは自主的な修正のたびに，その情報を著者の公式ホームページ（http://www.ark.is.kit.ac.jp/~shibayam/）で公開する．

2章を中心とする「組み合わせ回路とその設計」および3章を中心とする「同期式順序回路とその設計」のそれぞれに関する演習問題の補遺については，同じ公式ホームページ（http://www.ark.is.kit.ac.jp/~shibayam/）において，＜学部の講義（シラバス）＞・＜論理設計＞ → ＜成績評価の方法＞・＜過去問＞・＜各年度＞ → とたどる各ページで公開している．

これらの演習問題は全問とも，本書を教科書とする講義の履修者に対して，学期末に，著者が課したオリジナルな定期試験問題である．各年度の問題はすべてが組み合わせ回路（[A]）と同期式順序回路（[B]）の各1問をセットとしており，全部で18セット（計36問）の演習問題を掲載している．各セットの標準的な解答時間はセット（2問）当たり90分である．

この公式ホームページに掲載している演習問題は，組み合わせ回路と同期式順序回路それぞれについて，理論と実際のそれぞれを問う基本問題から発展問題までの順序立てた小問で構成してある．したがって，本書の各章末に収録してある演習問題に比べると，問題それぞれが総合的かつ体系的で完備している．

学習成果の確認のための一助としてチャレンジすることを勧める．

いつもながら，丁寧に査読をしてくれた同僚・平田博章 氏に深謝する．

最後に，我が家の構成員：妻・真木子，4人の子供たち・風野（ふうや），すみれ，ののみ，蒼宙（あおき）に,「君らの健康が本書執筆の推進剤だよ！ お・お・き・に」.

1999年 初夏 および 2013年 初春

京都・松ヶ崎あるいは岩倉にて

柴山　潔

下線部のUrlをクリックすると，ブラウザで当該ページを開きます．