

[B-2. 組み合わせ回路]

目的 論理回路は、入力によってのみ出力が決定される組み合わせ回路と、入力と回路の状態によって出力が決定される順序回路に大別できる。本実験では、組み合わせ回路に焦点をあて、その設計、動作確認を通して、組み合わせ回路の理解を深める。

1 原理

1.1 基本論理演算

論理代数（ブール代数）は、電子計算機の回路設計などの論理設計の有効な手段である。この論理代数で取り扱う変数は、“0”と“1”の2種類しかとらない変数で、論理変数という。最も基本的な論理演算は論理和、論理積、否定の3つである。

1.1.1 論理和（OR）演算

2変数 A, B とその関数 Y との間に、A または B が “1” であれば Y が “1” となる関係があるとき、Y は A, B の論理和であるといい、

$$Y = A + B$$

で表現する。論理和演算の結果は、以下の4つであり、これは論理演算の公理といわれる。

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 1$$

すなわち、2つの変数の論理和は、どちらかが “1” であれば結果は “1” になる。

1.1.2 論理積（AND）演算

2変数 A, B とその関数 Y との間に、A および B が “1” であるときのみ Y が “1” となる関係があるとき、Y は A, B の論理積であるといい、

$$Y = A \cdot B$$

で表現する。この結果は以下の4つしかない。

$$0 \cdot 0 = 0$$

$$0 \cdot 1 = 0$$

$$1 \cdot 0 = 0$$

$$1 \cdot 1 = 1$$

すなわち、2つの変数の論理積は2つとも “1” でなければ、結果は “1” とならない。

1.1.3 否定 (NOT) 演算

ある変数 A が “1” のとき関数 Y が “0”, 逆に A が “0” のとき Y が “1” となる関係があるとき, Y は A の否定といい,

$$Y = \bar{A}$$

と表記する. 否定演算の結果は以下の 2 つしかない.

$$\bar{0} = 1$$

$$\bar{1} = 0$$

すなわち, 否定演算の結果は “0” と “1” が逆に対応する.

1.1.4 真理値表

論理演算においては, 論理変数を取り得る値は “0” または “1” しかなく, しかも必ずどちらか 1 つの値しか取らない. この値を真理値という. また論理変数 A, B, C, … を組み合わせて作られる論理関数 Y (A, B, C, …) との関係を具体的に表現する方法として, 真理値表というものが使われる. これは各変数のすべての状態と, その結果を表で示したものである. 例えば, 論理和演算 ($Y = A + B$) の真理値表は表 1 のようになる.

表 1: 論理和演算の真理値表

| B | A | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

1.1.5 基本定理

論理代数では, 3 つの基本論理演算の公理に基づいて, いくつかの重要な基本定理が導かれる. それらを表 2 に示す.

表 2: 基本的な定理

| 定理の名称 | 定理の公式 |
|-------|-----------------------|
| 恒等の法則 | $0 + A = A$ |
| | $1 \cdot A = A$ |
| 同一の法則 | $A + A = A$ |
| | $A \cdot A = A$ |
| 補元の法則 | $A + \bar{A} = 1$ |
| | $A \cdot \bar{A} = 0$ |

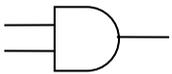
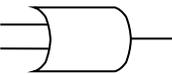
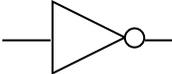
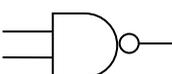
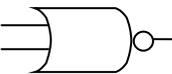
復元の法則 $A = \overline{\overline{A}}$

交換の法則 $A + B = B + A$
 $A \cdot B = B \cdot A$

1.2 基本論理回路

論理回路は、前述の論理代数で厳密に表現できるが、MIL (military standard specification) 記号法を用いると、直観的に分かりやすい。MIL 記号法による基本ゲートとその論理式、および動作内容をまとめたものを表 3 に示す。

表 3 : MIL 記号法による基本的なゲート

| 論理記号 | 論理式 | 動作 |
|---|----------------------------|--|
|  | $Y = A \cdot B$ | すべての入力が“1”の時、出力が“1”になる |
|  | $Y = A + B$ | 1つでも入力が“1”の時、出力が“1”になる |
|  | $Y = \overline{A}$ | 入力が“1”の時、出力が“0”になる 入力が“0”の時、出力が“1”になる |
|  | $Y = \overline{A \cdot B}$ | すべての入力が“1”の時、出力が“0”になる |
|  | $Y = \overline{A + B}$ | 1つでも入力が“1”の時、出力が“0”になる |

1.3 組み合わせ回路の設計 (加法標準形設計法)

加法標準形設計法は真理値表が与えられて、それに従った論理回路を設計する場合、最も簡単な方法である。手順は以下の通りである。

- ① 真理値表の“1”が出力されている行をチェックする。
- ② 入力信号とその反転信号を平行線で描く (反転記号は、記号の上に bar 「 $\overline{\quad}$ 」を書いて表す)。
- ③ “0”の場合反転記号から、“1”の場合もとの入力信号から線を引っ張り AND ゲートに入力する。

- ④ それらの AND 出力を OR で結ぶ.

1.4 カルノー図による組み合わせ回路の簡単化

加法標準形設計法の問題点は、真理値表上で“1”の出力1つに対して1つのANDゲートを割り当てる点にある。このためどうしてもANDゲートの数、入力数が増えてしまう。では、どうすればいいかというところ“1”のグループに対して1つのANDゲートを割り付けることができれば、簡単化が可能になる。“1”のグループを見つけるために、カルノー図と呼ばれる特殊な2次元真理値表を利用する。この図を利用することで、ANDゲートを割り付ける“1”のグループを効率良く見つける事が出来る。

カルノー図を用いた簡単化は以下のような手順で行われる。

- ① 真理値表の代わりにカルノー図に“1”，“0”を書き込む。
- ② “1”のグループをループでくくる。
- ③ ループに対応するANDゲートの入力を読み取り、回路を構成する。

カルノー図は基本的に真理値表を2次元に変形したものであるが、入力の表記順序に工夫が施されている。

| | | | | |
|----------|----|----|----|----|
| BA DC | 00 | 01 | 11 | 10 |
| 00 | | | | |
| 01 | | | | |
| 11 | | | | |
| 10 | | | | |

(a) 4入力

| | | | | |
|---------|----|----|----|----|
| BA C | 00 | 01 | 11 | 10 |
| 0 | | | | |
| 1 | | | | |

(b) 3入力

図1：カルノー図

図1は3入力、4入力に対するカルノー図である。この図を見ると縦横の値の順番が普通の2進数のように

00 → 01 → 10 → 11

でなく、

00 → 01 → 11 → 10

となっていることが分かる。つまり、カルノー図上で縦でも横でも“1”が2つ並んだ場合、その2つの“1”の間には入力1ビット変化しただけであるという関係が生じる。このよ

うな性質を利用して，“1”のグループを見つける。

| BA \ DC | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00 | | | | |
| 01 | | 1 | 1 | |
| 11 | | | | |
| 10 | | | | |

| BA \ DC | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00 | | | | |
| 01 | | 1 | | |
| 11 | | 1 | | |
| 10 | | | | |

(a) 横方向に“1”が2つ並んだ場合

(b) 縦方向に“1”が2つ並んだ場合

図2：カルノー図における“1”のグループ

図2(a)は横に2つ並んだ例であるが，上を見ると

入力 $B=0, A=1$ と $B=1, A=1$

であることが分かる。このカルノー図上の2つの“1”は $A=1$ という点が共通であり， B が“0”か“1”かという点が異なっている。つまり，この“1”のグループは以下のような性質を持つ。

$A=1, C=1, D=0$ であり， B は“1”でも“0”でもどちらでもよい

「“1”でも“0”でもどちらでもよいのなら， B という入力があってもなくても同じなので省略してしまう」というのがカルノー図を用いた簡単化の基本的方針である。そして，この“1”のグループを，

$$A \cdot C \cdot \bar{D}$$

と表記する。同様に縦に2つ“1”が並んだ図2(b)の場合を考えてみると，

$A=1, B=0, C=1$ であり， D は“1”でも“0”でもどちらでもよい

ということが分かり， D が省略可能である。この“1”のグループは，

$$A \cdot \bar{B} \cdot C$$

と表現できる。

2 実験項目

1. NAND ゲートを用いて NOT, AND, OR を設計, 作成し, 動作を確認せよ.
2. 4 本入力線があり, これで 2 進数 (0~15) を表現するものとする. カルノー図による簡単化手法を用いて次の課題を行え.
 - (a) 素数を検出する回路を設計せよ.
(入力された数が素数であれば出力が 1 となる回路を設計せよ)
 - (b) (a)で作成した回路を NAND ゲートのみを用いて設計せよ.
 - (c) (b)で設計した回路を作成し, 動作を確認せよ.
3. 株主総会がある. A 氏は全体の 35%, B 氏は 30%, C 氏は 19%, D 氏は 16%の株をそれぞれ持っている. ある議題について, 4 人は賛成か反対か手元のスイッチを押すことにより投票する. 賛成者の持株が 50%以上になれば, その議題は可決, 50%未満ならば否決される.
 - (a) 議題の可否を判別する回路を設計せよ.
 - (b) (a)で設計した回路を作成し, 動作を確認せよ.

3 考察

実験結果について考察せよ.

4 調査

- (1) 組み合わせ回路は, 実験書にある加法標準形と, 乗法標準形のどちらを用いても設計することができる. 乗法標準形はどのような方法かを調査せよ. また, どのような場合に加法標準形を用い, どのような場合に乗法標準形を用いるのが便利であるかを検討せよ.
- (2) NAND ゲートのみを用いて回路を作成する利点と欠点について, 設計と実装それぞれの面から考察せよ.
- (3) 組み合わせ回路の実現には, 多くの場合でスタティック CMOS 回路が用いられている. 他の回路形式との比較とともに, その理由を調査せよ.

以上