

# 電子回路の組み立てと動作実験【設計編】

NAND ゲートを用いた簡単なロジック回路の設計

農業システム工学分野

第3版

## 目次

1	目的	2
2	参考書について	2
3	ゲート回路の基礎知識（おさらい）	3
3.1	正論理と負論理	3
3.2	NOT ゲート（インバータ）	3
3.3	AND ゲート（論理積）	4
3.4	OR ゲート（論理和）	4
3.5	ブール代数の基本公式	5
3.6	NAND ゲート（否定論理積）	8
3.7	NOR ゲート（否定論理和）	8
3.8	Exclusive OR ゲート（排他的論理和）	9
4	NAND ゲートによる表現	9
4.1	NOT ゲート（インバータ）の NAND による表現	10
4.2	AND ゲート（論理積）の NAND による表現	10
4.3	OR ゲート（論理和）の NAND による表現	10
4.4	NOR ゲートの NAND による表現	10
4.5	Exclusive OR ゲート（排他的論理和）の NAND による表現	12
5	応用回路とその設計例	12
5.1	2 ビット 2 進数をデコードして出力（2 進 2 ビットデコーダ）	13
5.1.1	真理値表の作成	13
5.1.2	式による表現	13
5.1.3	2 進 2 ビットデコーダの回路図	13
5.2	3 入力多数決回路	14
5.2.1	真理値表の作成	14
5.2.2	積の和形式で表現（加法標準形）	15
5.2.3	式の変形（NAND 化）	16
5.2.4	図の置き換えによる NAND 化	17

5.3	パリティチェック	18
5.3.1	2入力パリティチェック	18
5.3.2	4ビット入力への拡張	19
5.4	半加算器	19
5.4.1	2進数の加算	19
5.4.2	半加算器	20
5.5	全加算器	20
5.5.1	全加算器の真理値表	21
5.5.2	全加算器の式	21
5.5.3	半加算器の組み合わせによる全加算器	21
5.5.4	NAND化した全加算器	22
5.5.5	半加算器と全加算器による多桁の2進加算	23
5.5.6	多数決回路再考	23
<b>6</b>	<b>使用する NAND ゲート IC</b>	<b>24</b>
6.1	74AC00 について	25
6.2	NAND の出力による LED の点灯	25
6.3	スイッチによる入力	25
6.4	バイパスコンデンサ（パスコン）について	26
6.5	CMOS ロジックの未使用入力の処理	27
<b>7</b>	<b>この実験で用いる回路について</b>	<b>27</b>
7.1	回路図の作成	28
<b>8</b>	<b>組み立て実習について</b>	<b>28</b>
8.1	注意事項	28
8.1.1	安全上の注意	28
8.1.2	静電気による破壊について	28
8.1.3	製作した回路の安全な取り扱いについて	28

## 1 目的

2入力の AND, OR, NAND, NOR, Exclusive OR について解説を行ない各種ロジック回路の基本的な動作について理解する。次に NAND ゲートを最大 12 個まで使用して、各自のオリジナルな回路もしくは本資料で解説される各種回路、または NAND ゲートにより表現された AND, OR, NOR, Exclusive OR を設計する。これら NAND ゲートのみで表現した回路について、今回および次回の実験で組み立てを行ない、実際の動作が正しいか確認を行なう。回路にはスイッチによる入力を用意し、これらをゲートの入力として LED を光らせて、各自の回路が設計どおりに動作しているかどうかを確認する。

## 2 参考書について

3 回生配当の講義「電気・電子工学」の教科書「メカトロニクスのための電子回路基礎」を適宜参照すること。

### 3 ゲート回路の基礎知識（おさらい）

本実験実習のために必要なゲート回路の基礎についておさらいしておく。

#### 3.1 正論理と負論理

本実験では正論理のみで考えていくこととする。すなわち回路の高いほうの電圧を  $H$ ，低いほうの電圧を  $L$  として表 1 のように考える。

表 1: 正論理

電圧	真偽	ブール代数での値	能動／非能動
$H$	真	1	active
$L$	偽	0	passive

#### 3.2 NOT ゲート（インバータ）

1 つの入力に対して表 2 で表される値を出力する。

表 2: NOT ゲート（インバータ）の真理値表

入力	出力
$A$	$Z$
0	1
1	0

1 が入力されれば 0 を，0 が入力されれば 1 を出力する。入力を反転するという言い方もする。NOT ゲートの記号は図 1 のように表す。ブール代数による表現では，

$$Z = \neg A \quad (1)$$

$$Z = \overline{A} \quad (2)$$

となる。本稿では式 2 の表記を用いる。



図 1: NOT ゲートの記号

### 3.3 AND ゲート（論理積）

2つ以上の入力すべてが1のとき出力が1となる。2入力の場合の真理値表を表3に示す。3入力以上に関しても同様に入力のうちひとつでも0があれば、出力は0となる。

表 3: AND ゲート（2 入力）の真理値表

入力		出力
A	B	Z
0	0	0
0	1	0
1	0	0
1	1	1

AND ゲートの記号は図2のように表記する。ブール代数による表現では、

$$Z = A \wedge B \quad (3)$$

$$Z = A \times B \quad (4)$$

$$Z = A \cdot B \quad (5)$$

$$Z = A \& B \quad (6)$$

$$Z = AB \quad (7)$$

となり、分野により様々な表記がある。

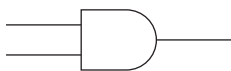


図 2: AND ゲートの記号

**練習 1.** 3入力の AND ゲートを 2入力 AND ゲート 2 個を使って表わせ。

### 3.4 OR ゲート（論理和）

2つ以上の入力のうちひとつでも1があれば出力が1となる。2入力の場合の真理値表を表4に示す。3入力以上に関しても同様に入力のすべてが0でなければ、出力は0とならない。

OR ゲートの記号は図3のように表記する。ブール代数による表現では、

$$Z = A \vee B \quad (8)$$

表 4: OR ゲート (2 入力) の真理値表

入力		出力
$A$	$B$	$Z$
0	0	0
0	1	1
1	0	1
1	1	1

$$Z = A + B \quad (9)$$

$$Z = A \mid B \quad (10)$$

となり, これも分野により複数の表記方法がある.

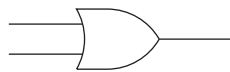


図 3: OR ゲートの記号

**練習 2.** 3 入力の OR ゲートを 2 入力 OR ゲート 2 個を使って表わせ.

### 3.5 ブール代数の基本公式

残りのゲートについて説明する前に, 以下に論理回路の設計によく用いられるブール代数の基本公式を示す. なお演算子の優先順位は一般的には否定 (NOT,  $\overline{\phantom{x}}$ ), 論理積 (AND,  $\cdot$ ), 論理和 (OR,  $+$ ) の順である (かっこで優先順位が示されていない場合).

#### 単位元

$$A + 0 = A \quad (11)$$

$$A \cdot 1 = A \quad (12)$$

#### 冪等

$$A + A = A \quad (13)$$

$$A \cdot A = A \quad (14)$$

**補元則**

$$A + \bar{A} = 1 \quad (15)$$

$$A \cdot \bar{A} = 0 \quad (16)$$

**二重否定**

$$\overline{\bar{A}} = A \quad (17)$$

**交換則**

$$A + B = B + A \quad (18)$$

$$A \cdot B = B \cdot A \quad (19)$$

**結合則**

$$(A + B) + C = A + (B + C) \quad (20)$$

$$(A \cdot B) \cdot C = A \cdot (B \cdot C) \quad (21)$$

**分配則**

$$A \cdot (B + C) = A \cdot B + A \cdot C \quad (22)$$

$$A + (B \cdot C) = (A + B) \cdot (A + C) \quad (23)$$

**吸収則**

$$A + A \cdot B = A \quad (24)$$

$$A \cdot (A + B) = A \quad (25)$$

**ド・モルガンの定理**

$$\overline{A + B} = \bar{A} \cdot \bar{B} \quad (26)$$

$$\overline{A \cdot B} = \bar{A} + \bar{B} \quad (27)$$

分配則の式 23 の両辺を論理回路で表すと、図 4 のようになる。これらは等価なので、一般にはゲートがひとつ少ない左のほうが良いと言える。

**練習 3.** ブール代数の公式の残りの式の各辺を論理回路記号を使って表わせ。

**練習 4.** 上記の各公式の両辺が等しいことを真理値表を書いて確かめてみよ。

**練習 5. (双対)** 上記の基本公式において、積の記号 (・) と和の記号 (+) を、かつ 0 と 1 をそれぞれ入れ替えても成立する。これを双対性という。各自で確かめてみよ。

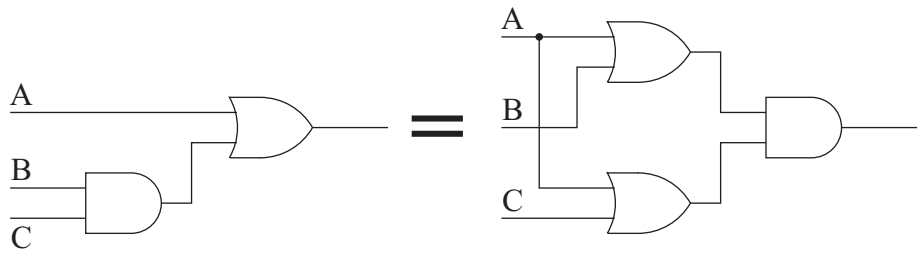


図 4: 分配則の両辺を論理回路で表す



図 5: NAND ゲートの記号

表 5: NAND ゲート（2 入力）の真理値表

入力		AND 出力	NAND 出力
A	B	Z	
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

### 3.6 NAND ゲート（否定論理積）

AND ゲートの出力を NOT ゲートに通したものを出力とする。すなわち図 5 で示されるものである。2 入力の場合の真理値表を表 5 に示す。3 入力以上に関しても同様に考える。

ブール代数による表現では、以下のようになる。

$$Z = \overline{A \wedge B} \quad (28)$$

$$Z = \overline{A \times B} \quad (29)$$

$$Z = \overline{A \cdot B} \quad (30)$$

$$Z = \overline{A \& B} \quad (31)$$

$$Z = \overline{AB} \quad (32)$$

**練習 6. （ド・モルガンの定理その 1）** ド・モルガンの定理の式 27 の左辺は NAND である。この式の右辺を NOT と OR で表し、真理値表を書いてこの等式が成り立つことを示せ。

### 3.7 NOR ゲート（否定論理和）

OR ゲートの出力を NOT ゲートに通したものを出力とする。すなわち図 6 で示されるものである。2 入力の場合の真理値表を表 6 に示す。3 入力以上に関しても同様に考える。



図 6: NOR ゲートの記号

表 6: NOR ゲート（2 入力）の真理値表

入力		OR 出力	NOR 出力
A	B		Z
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

ブール代数による表現では、以下のようになる。

$$Z = \overline{A \vee B} \quad (33)$$

$$Z = \overline{A + B} \quad (34)$$

$$Z = \overline{A | B} \quad (35)$$



**練習 7.** (ド・モルガンの定理その 2) ド・モルガンの定理の式 26 の左辺は NOR である。この式の右辺を NOT と AND で表し、真理値表を書いてこの等式が成り立つことを示せ。

### 3.8 Exclusive OR ゲート (排他的論理和)

2つの入力の値が同じときには 0 を、違う時には 1 を出力する。真理値表を表 7 に示す。記号は図 7 で示されるものである。Exclusive OR は XOR や EX-OR と書かれることもある。

表 7: XOR ゲート (2 入力) の真理値表

入力		出力
A	B	Z
0	0	0
0	1	1
1	0	1
1	1	0

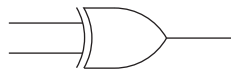


図 7: XOR ゲートの記号

ブール代数による表現では、以下のようになる (他の表記もあるが省略)。

$$Z = A \oplus B \quad (36)$$

$$Z = A \underline{\vee} B \quad (37)$$

XOR の演算を今まで出てきた論理積、論理和、否定を使うと以下のように表わせる。

$$Z = A \oplus B = A \cdot \bar{B} + \bar{A} \cdot B \quad (38)$$

**発展 1.** 3 入力以上の排他的論理和がどのようなものか調べてみよ。「入力がすべて同じ値なら 0, 違う値があれば 1 を出力」**ではない**。

## 4 NAND ゲートによる表現

ここで NOT および 2 入力の AND, OR, NOR, XOR の各ゲートを NAND ゲートを使って表記することを考える。

#### 4.1 NOT ゲート（インバータ）の NAND による表現

図 8 のようにすれば良い。ブール代数の公式のうち、式 14 を使っている。



図 8: NOT を NAND で実現

#### 4.2 AND ゲート（論理積）の NAND による表現

これは簡単に図 9 のように表わせる。二重否定の式 17 を使っている。

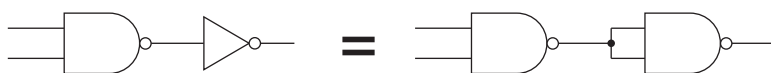


図 9: AND を NAND で実現

#### 4.3 OR ゲート（論理和）の NAND による表現

ド・モルガンの定理の式 27 より、NAND ゲートは図 10 のように表わせるので、入力 of 否定を打ち消す NOT を付け加えて、図 11 のように表現できる。

#### 4.4 NOR ゲートの NAND による表現

上記の NAND による OR の出力を否定すればよいから、図 12 のように表わすことができる。

**練習 8.** NAND ゲートの働きを NOR ゲートのみ使って表わしてみよ。

**練習 9.** 3 入力の NAND ゲートを 2 入力 NAND 3 個を使用して表わせ（2 入力 NAND のうちの 1 つはインバータとして使用する）。

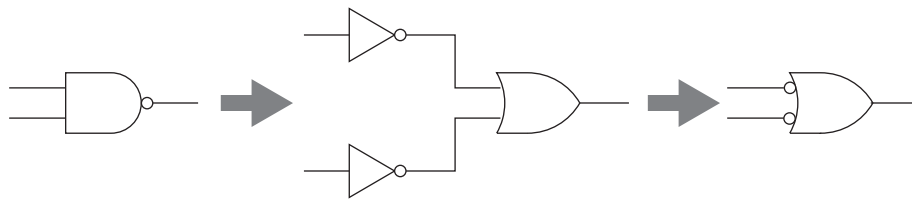


図 10: NAND の別の表現

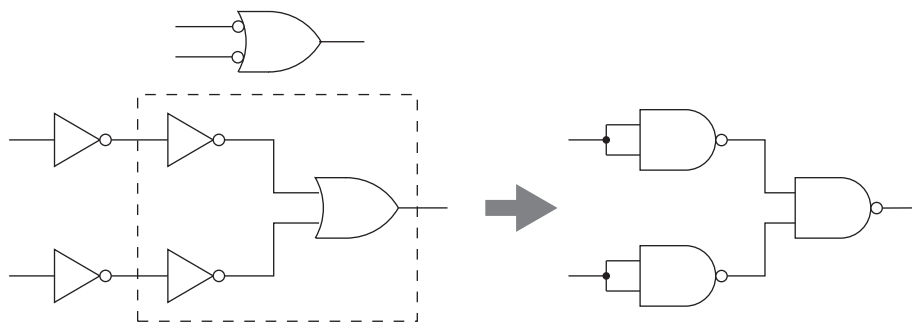


図 11: OR を NAND で実現

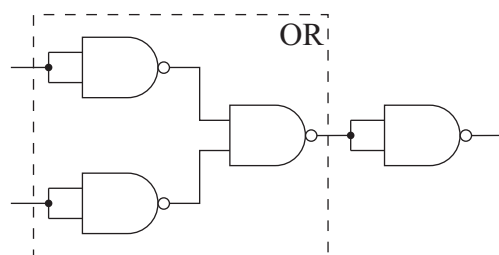


図 12: NOR を NAND で実現

**練習 10.** 3 入力の NOR ゲートを 2 入力 NAND のみを使用して表わせ、2 入力 NAND（インバータとして使用するものも含む）は何個必要か？

#### 4.5 Exclusive OR ゲート（排他的論理和）の NAND による表現

NAND ゲートのみを使った XOR の実現は図 13 となる。

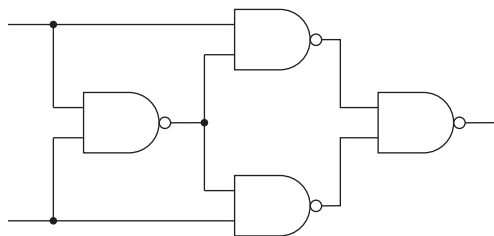


図 13: XOR を NAND で実現

**練習 11.** **（XOR ゲート）** 表 7 および式 38 に示される XOR が上記の図 13 となることをブール代数の式変形を用いて示せ。

【ヒント】図 13 をブール代数の式で表現すると、

$$\overline{\overline{(A \cdot (\overline{A \cdot B}))} \cdot (B \cdot (\overline{A \cdot B}))} \quad (39)$$

となる（各自確かめてみよ）。そこでこの式にたどり着くように、出発点の式 38 の  $A \cdot \overline{B} + \overline{A} \cdot B$  を変形すれば良い。ただしこれは厄介なので、逆に図 13 を表わす式 39 から始めて、 $A \cdot \overline{B} + \overline{A} \cdot B$  になるように変形していく。

**練習 12.** **（NOR ゲートによる表現）** 上と同様に NOT と 2 入力の AND, OR, NAND, XOR の各ゲートを NOR で書き表せ。

**練習 13.** **（完全系）** すべての論理回路は NAND のみ、もしくは NOR のみを使って書き換えることができる。NOT, AND, OR（および NAND, NOR）を使った適当な論理回路を書き、それを NAND のみ、または NOR のみで書き換えてみよ。真理値表を用いて書き換え前と書き換え後の回路が等価であることを確かめてみよ。

## 5 応用回路とその設計例

今回の実習で用いる規模のゲートを使用した、ごく簡単な応用回路を示す。ここでは真理値表と簡単な式変形のみで設計して行く。Karnaugh 図法や Quine 法といった簡略化の方法は学習していないので、用いないこととする。各種の簡略化技法を使用しないため、ごく初歩的で単純な設計過程を示すだけである。

## 5.1 2ビット2進数をデコードして出力（2進2ビットデコーダ）

2つの入力を2ビット（2桁）の2進数と解釈し、4つの出力線の中からその値に対応した出力のみ  $H$  を出力する回路である。4つの出力それぞれに LED を付けておき、10進数で0から3に対応させておくと、入力の2桁2進数00, 01, 10, 11の値に応じて、0, 1, 2, 3を示すランプ（LED）が光る。

### 5.1.1 真理値表の作成

2ビット（2桁）の2進数を示す2つの入力をそれぞれ  $A_1, A_0$  として、出力を  $Z_0, Z_1, Z_2, Z_3$  とする。 $A_1$  が上位桁、 $A_0$  が下位桁とする。また  $Z_0, Z_1, Z_2, Z_3$  はそれぞれ10進の0, 1, 2, 3を示す出力である。出力は入力の2進数の値に応じていずれか1つに  $H$  が出力される。表8にその真理値表を示す。

表 8: 2進2桁デコーダの真理値表

入力		出力			
$A_1$	$A_0$	$Z_0$	$Z_1$	$Z_2$	$Z_3$
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

### 5.1.2 式による表現

4つの出力  $Z_0, Z_1, Z_2, Z_3$  のそれぞれについて別々に考えればよい。10進で0を表わす出力  $Z_0$  については、 $A_1 = 0, A_0 = 0$  のときのみ出力が1になるので、

$$Z_0 = \overline{A_1} \cdot \overline{A_0} \quad (40)$$

他の出力についても同様に、

$$Z_1 = \overline{A_1} \cdot A_0 \quad (41)$$

$$Z_2 = A_1 \cdot \overline{A_0} \quad (42)$$

$$Z_3 = A_1 \cdot A_0 \quad (43)$$

### 5.1.3 2進2ビットデコーダの回路図

式40～式43を回路図に表わすと図14のようになる。これをNANDで表現すると図15となる。この図15はNANDゲートを12個使用している。

**練習 14.** LEDを光らせる場合、 $H$ で点灯ではなく、 $L$ で点灯する回路にすれば、図15の出力のインバータ（NOT）は不要となる。後述の図37のLED回路を出力に追加し、出力の4つのインバータ（NOT）を省いた回路図を書け。

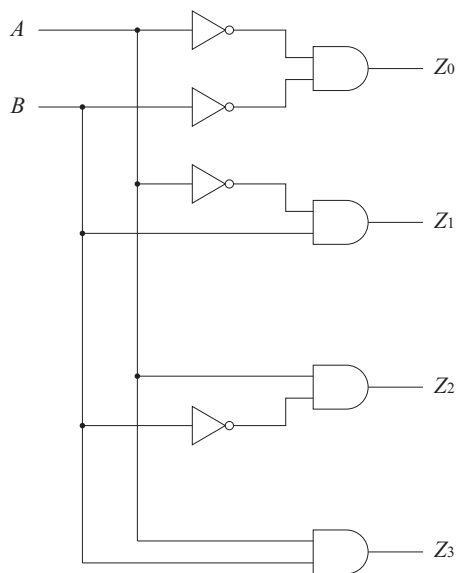


図 14: AND と NOT による 2 進 2 ビットデコーダ

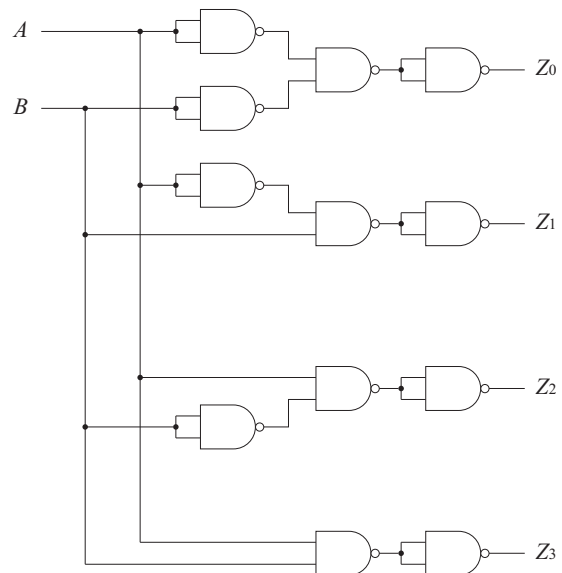


図 15: NAND による 2 進 2 ビットデコーダ

## 5.2 3 入力多数決回路

3 つの入力のうち 2 つ以上が  $H$  の時、出力が  $H$  となる回路を考える。

### 5.2.1 真理値表の作成

3 つの入力をそれぞれ  $A$ ,  $B$ ,  $C$  として、出力を  $Z$  とする。  $A$ ,  $B$ ,  $C$  のすべての組み合わせに対して出力  $Z$  がどのような値をとるかを表にする。 多数決という条件から、入力の 2 つ以上が 1 になった場合のみ、出力  $Z$  が 1 となる。 表 9 にその真理値表を示す。

表 9: 3 入力の多数決の真理値表

	入力			出力
	$A$	$B$	$C$	$Z$
(1)	0	0	0	0
(2)	0	0	1	0
(3)	0	1	0	0
(4)	0	1	1	1
(5)	1	0	0	0
(6)	1	0	1	1
(7)	1	1	0	1
(8)	1	1	1	1

### 5.2.2 積の和形式で表現（加法標準形）

表 9 の各行のうち，出力  $Z$  が 1 になっている行に着目する．それらは (4), (6), (7), (8) の各行である．(4) の行は積 (AND) を使って，

$$Z_4 = \bar{A} \cdot B \cdot C \quad (44)$$

と表わせる．(6), (7), (8) についても同様に積で表わすと，

$$Z_6 = A \cdot \bar{B} \cdot C \quad (45)$$

$$Z_7 = A \cdot B \cdot \bar{C} \quad (46)$$

$$Z_8 = A \cdot B \cdot C \quad (47)$$

と表わせる．これら 4 つのケースの和が求める多数決のケース全体を表わすと考えて，

$$Z = \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot C + A \cdot B \cdot \bar{C} + A \cdot B \cdot C \quad (48)$$

と積の和形式で書くことができる．

この式 48 を式変形で簡略化して行く．5 ページの冪等（式 13）を用いると，

$$A \cdot B \cdot C = A \cdot B \cdot C + A \cdot B \cdot C + A \cdot B \cdot C \quad (49)$$

となる．式 48 の  $A \cdot B \cdot C$  を上式で置き換えて，

$$Z = \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot C + A \cdot B \cdot \bar{C} + A \cdot B \cdot C + A \cdot B \cdot C + A \cdot B \cdot C \quad (50)$$

とする．項の順番を並び替えてまとめると，

$$Z = (\bar{A} \cdot B \cdot C + A \cdot B \cdot C) + (A \cdot \bar{B} \cdot C + A \cdot B \cdot C) + (A \cdot B \cdot \bar{C} + A \cdot B \cdot C) \quad (51)$$

となる．6 ページの分配則（式 22）を各カッコ内で用いると，

$$Z = (\bar{A} + A) \cdot B \cdot C + (\bar{B} + B) \cdot A \cdot C + (\bar{C} + C) \cdot A \cdot B \quad (52)$$

となる．ここで補元則の式 15（6 ページ）により，

$$\bar{A} + A = \bar{B} + B = \bar{C} + C = 1 \quad (53)$$

なので結局，式 48 は，

$$Z = A \cdot B + B \cdot C + C \cdot A \quad (54)$$

となる．

この式 54 をそのまま回路図に書き下すと，図 16 のようになる．

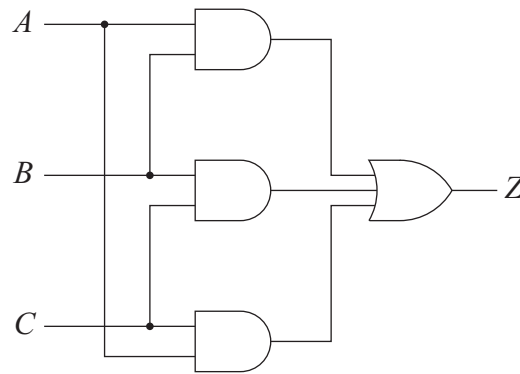


図 16: AND と OR による 3 入力多数決回路

### 5.2.3 式の変形 (NAND 化)

この実験では NAND ゲートのみを用いることになっている。そこで図 16 を、NAND を用いて表現することにする。まずは式変形により NAND 化を行なってみる。

はじめに NAND は式で書くと  $\overline{X \cdot Y}$  (もしくは  $\overline{X} + \overline{Y}$ ) の形になるということを認識しておく。次に、上で積の和形式 (加法標準形) により表現した式 54 を、二重否定の式 17 を用いて次の式 55 のように表わす。

$$Z = \overline{\overline{A \cdot B} + \overline{B \cdot C}} + C \cdot A \quad (55)$$

式 55 の左辺第 1 項と第 2 項にド・モルガンの定理の式 27 (6 ページ) を適用し、次の式 56 のように変形する。

$$Z = \overline{(\overline{A \cdot B}) \cdot (\overline{B \cdot C})} + C \cdot A \quad (56)$$

$\overline{A \cdot B}$  および  $\overline{B \cdot C}$  がそれぞれ NAND になっており、これら 2 つを入力とする  $\overline{(\overline{A \cdot B}) \cdot (\overline{B \cdot C})}$  も NAND になっていることに留意する。

さらに同じく二重否定の式 17 (6 ページ) を用いて、

$$Z = \overline{\overline{(\overline{A \cdot B}) \cdot (\overline{B \cdot C})} + C \cdot A} \quad (57)$$

とする。左辺の上から 2 本目の否定の棒 (  $\overline{\phantom{x}}$  ) と + を用いて、ド・モルガンの定理の式 26 (6 ページ) を適用すると、

$$Z = \overline{\overline{(\overline{A \cdot B}) \cdot (\overline{B \cdot C})} \cdot \overline{C \cdot A}} \quad (58)$$

となった。この式 58 は NAND と NOT で表わされている。すなわち、

- $\overline{A \cdot B}$  および  $\overline{B \cdot C}$  はそれぞれ A, B および B, C を入力とする NAND
- $\overline{(\overline{A \cdot B}) \cdot (\overline{B \cdot C})}$  は  $\overline{A \cdot B}$  と  $\overline{B \cdot C}$  を入力とする NAND



- $\overline{\overline{A \cdot B} \cdot \overline{B \cdot C}}$  は上の否定 (NOT)
- $\overline{C \cdot A}$  は C, A を入力とする NAND
- 式 58 は  $\overline{\overline{A \cdot B} \cdot \overline{B \cdot C}}$  と  $\overline{C \cdot A}$  を入力とする NAND

これを回路図にそのまま表わすと図 17 のようになる。

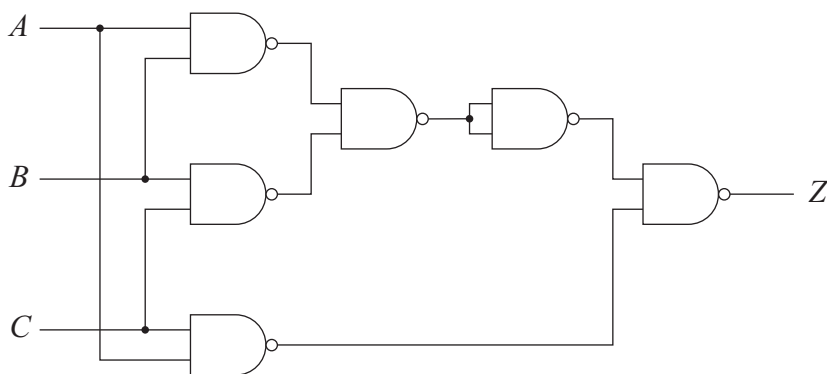


図 17: NAND による 3 入力多数決回路

**練習 15.** 式 54 から式 58 への変形を，別の式変形の方法で行なってみよ。

#### 5.2.4 図の置き換えによる NAND 化

これまでの練習で AND や OR を NAND で表現したので，それを用いて単純に置き換えても良い。図 16 の 3 入力 OR を 2 入力 OR に変えたものが図 18 である。

各 AND と OR を NAND による表現に置き換えたものが図 19 である。NOT と NOT が連続する部分は冗長なので簡略化できる。結局，図 17 と同じものができあがる。

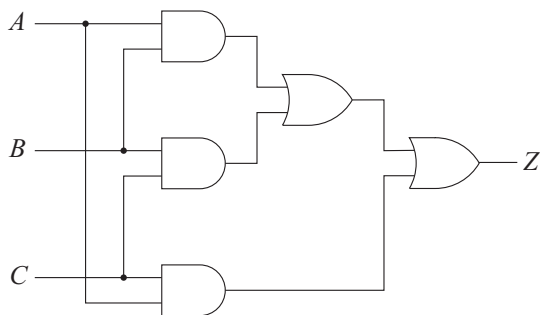


図 18: 2 入力 AND と 2 入力 OR による 3 入力多数決回路

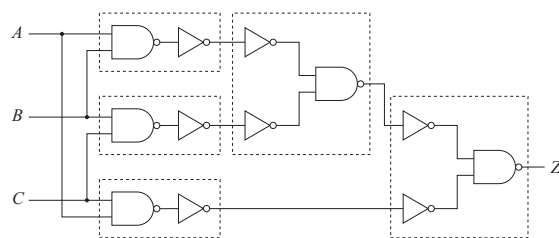


図 19: NAND による 3 入力多数決回路

**発展 2. (2進2ビットエンコーダ)** 5.1 節 (13 ページ) のデコーダ回路とは逆に, それぞれ 10 進数の 0, 1, 2, 3 に対応した 4 つの入力  $A_0, A_1, A_2, A_3$  のうち, いずれか 1 つを  $H$  にする, すなわちいずれか 1 つのスイッチを入れると, その 10 進数に対応した 2 ビット (2 桁) の 2 進数 ( $Z_1, Z_0$ ) を出力する回路を設計せよ. 2 本の出力に LED を付けておくと, 2 桁 2 進数 00, 01, 10, 11 の値を示すランプ (LED) が光る (1 すなわち  $H$  のビットの LED が光る). 4 つの入力は 1 つしか  $H$  にしないことになっているが, 同時に 2 つ以上  $H$  にした場合は出力が 00 になるものとする.

### 5.3 パリティチェック

複数の入力のうち,  $H$  (すなわち 1) になっている入力の個数が偶数個のとき 1 を出力する回路 (偶数パリティチェック), 同じく 1 になっている入力の個数が奇数個のとき 1 を出力する回路 (奇数パリティチェック) を考える. パリティの用途については各自調べてみることに.

#### 5.3.1 2 入力パリティチェック

まず入力が 2 本の場合について考える. 偶数パリティチェックの出力を  $P_e$ , 奇数パリティチェックの出力を  $P_o$  とし, 入力  $A_1, A_0$  に対する真理値表を作ると, 表 10 のようになる.

表 10: 2 ビットのパリティチェック

入力		出力	
$A_1$	$A_0$	$P_e$	$P_o$
0	0	1	0
0	1	0	1
1	0	0	1
1	1	1	0

この表を見て気づくことは, 奇数パリティ  $P_o$  が XOR の真理値表 (9 ページの表 7) と同じことである. したがって, 2 ビットの奇数パリティチェックはそのまま XOR となり, 偶数パリティチェックはその否定 (NOT) となる (図 20, 図 21).

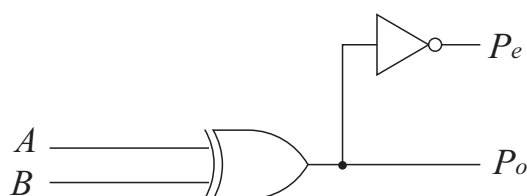


図 20: 2 ビットのパリティチェック回路

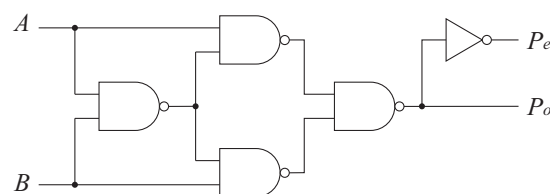


図 21: NAND による 2 ビットのパリティチェック回路

### 5.3.2 4ビット入力への拡張

入力が4ビットの場合，上記の2ビットを利用して拡張してみる．4本の入力を2本ずつに分けてそれぞれに対して上記のXORによる2ビットパリティチェック出力を得る．この2つの出力が奇数・奇数（1 1）および偶数・偶数（0 0）のときに，最終的な4ビットパリティチェックは偶数となるので，結局同じようにXORをとれば良いことがわかる．4ビットパリティチェックの回路を図22と図23に示す．

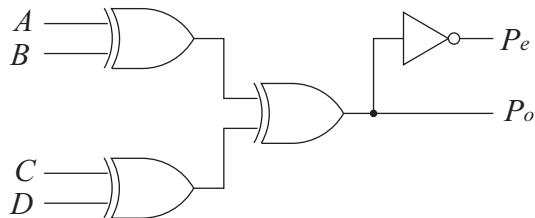


図 22: 4ビットのパリティチェック回路

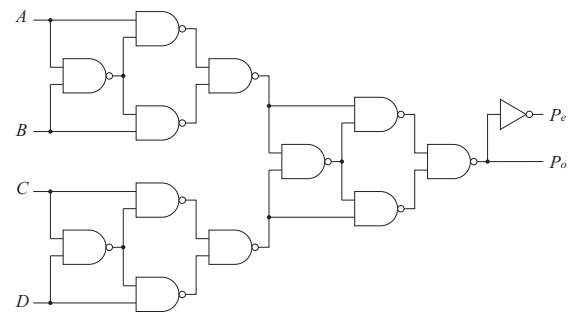


図 23: NAND による 4ビットのパリティチェック回路

**練習 16.** 入力が3ビット，5～8ビットのそれぞれの場合についてのパリティチェック回路を考えよ．

## 5.4 半加算器

足し算を行なう回路について見ていく．加算はもちろん2進数で行なわれ，桁数も多桁は考えないこととする．まずは半加算器について考える．

### 5.4.1 2進数の加算

1桁2進数どうしの加算は次のようになる．2進数では0と1しか使用しないので，1+1で結果が桁上がりして2桁となる．

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 0 &= 1 \\ 1 + 1 &= 10 \end{aligned}$$

**練習 17.** 2進2桁どうしの加算を行なえ．

### 5.4.2 半加算器

上でみた1桁(1ビット)2進数どうしの加算を行なう回路を考える。これを半加算器と呼ぶ。結果は2ビットとなるので、出力が2つとなる。最下位桁を  $s$ 、桁上がりが見される桁を  $c_{out}$  として、真理値表を書くと、表11のようになる。上の式そのままである。

表 11: 半加算器による1ビット2進数どうしの加算

入力		出力	
$A$	$B$	$c_{out}$	$s$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

表11で  $s$  は XOR の出力と同じであることがわかる。また、 $c_{out}$  は AND そのものである。

$$s = A \oplus B = A \cdot \bar{B} + \bar{A} \cdot B \quad (59)$$

$$c_{out} = A \cdot B \quad (60)$$

これを回路図で表わすと図24のようになる。この回路を NAND のみで構成すると図25のようになる。この場合、図24の AND の部分は XOR の中にある NAND を利用することができる。

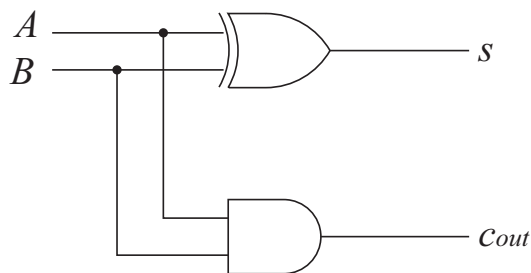


図 24: 半加算器 (XOR と AND)

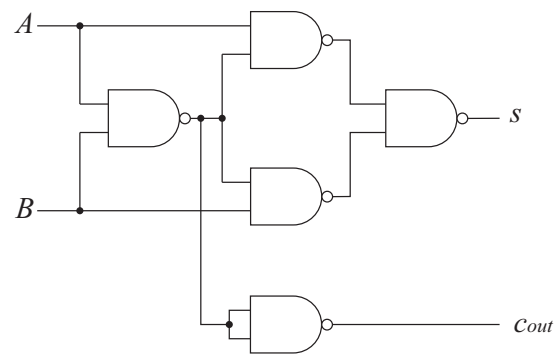


図 25: NAND による半加算器

## 5.5 全加算器

上で説明した半加算器は上位桁への桁上がりはあるが、下位桁から桁上りを処理する機能はない。そのため半加算器だけを並べて多桁の加算を行なうことはできない。そこで下位桁から来る桁上がり ( $C_{in}$ ) を処理できる全加算器について考える。

### 5.5.1 全加算器の真理値表

半加算器の真理値表に加えて、下位桁からやってくる桁上りを考える必要がある。下位桁から来る桁上りを  $C_{in}$  として真理値表を書くと表 12 のようになる。

表 12: 全加算器

	入力			出力	
	$A$	$B$	$C_{in}$	$C_{out}$	$S$
(1)	0	0	0	0	0
(2)	0	0	1	0	1
(3)	0	1	0	0	1
(4)	0	1	1	1	0
(5)	1	0	0	0	1
(6)	1	0	1	1	0
(7)	1	1	0	1	0
(8)	1	1	1	1	1

### 5.5.2 全加算器の式

表 12 から全加算器の出力  $C_{out}$  と  $S$  を表わす式を考える。これまでと同じ方法で、表 12 の  $C_{out}$  や  $S$  が 1 になっている行について積の和形式で表わす。まず  $C_{out}$  に関しては、表中の (4), (6), (7), (8) が 1 になっている行なので、

$$C_{out} = \bar{A} \cdot B \cdot C_{in} + A \cdot \bar{B} \cdot C_{in} + A \cdot B \cdot \bar{C}_{in} + A \cdot B \cdot C_{in} \quad (61)$$

となる。これはよく見ると 3 入力多数決回路の式と同じである（15 ページの式 48 を参照）。この式の簡略化は多数決回路で見たものと同じで結局、式 54（15 ページ）と同じである。

$$C_{out} = A \cdot B + B \cdot C_{in} + C_{in} \cdot A \quad (62)$$

$S$  に関しては、(2), (3), (5), (8) の各行が 1 になっているので、同じく積の和形式で表わして、

$$S = \bar{A} \cdot \bar{B} \cdot C_{in} + \bar{A} \cdot B \cdot \bar{C}_{in} + A \cdot \bar{B} \cdot \bar{C}_{in} + A \cdot B \cdot C_{in} \quad (63)$$

となる。

この式 62 および式 63 をそのまま回路図に書き下したものが図 26 である。次にこれを簡略化していくことにする。

### 5.5.3 半加算器の組み合わせによる全加算器

上記の全加算器は前述の半加算器を 2 つ組み合わせても実現できる。式 63 を次のように変形する。

$$S = (A \cdot B + \bar{A} \cdot \bar{B}) \cdot C_{in} + (\bar{A} \cdot B + A \cdot \bar{B}) \cdot \bar{C}_{in} \quad (64)$$

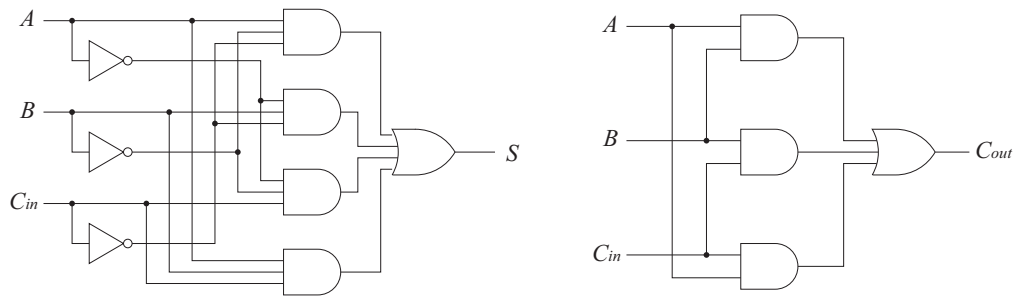


図 26: 簡略化前の全加算器

上の式 64 の右辺第 2 項の中の  $(\bar{A} \cdot B + A \cdot \bar{B})$  は  $A$  と  $B$  を入力とする XOR の出力そのもので、半加算器の出力  $s$  (式 59) と同じである。したがって、

$$(\bar{A} \cdot B + A \cdot \bar{B}) \cdot \overline{C_{in}} = s \cdot \overline{C_{in}} \quad (65)$$

また式 64 の右辺第 1 項の中の  $(A \cdot B + \bar{A} \cdot \bar{B})$  は以下のように変形して、半加算器の出力  $s$  の否定すなわち  $\bar{s}$  となることがわかる。

$$A \cdot B + \bar{A} \cdot \bar{B} = (A + \bar{B})(\bar{A} + B) \quad (66)$$

$$= \overline{\bar{A} \cdot B + A \cdot \bar{B}} \quad (67)$$

$$= \bar{s} \quad (68)$$

ゆえに式 64 は、

$$S = s \cdot \overline{C_{in}} + \bar{s} \cdot C_{in} \quad (69)$$

となる。式 69 の中の  $s$  は  $A$  と  $B$  を入力とする半加算器の出力  $s$  そのものであるが、式 69 の右辺全体の形は  $x \cdot \bar{y} + \bar{x} \cdot y$  の形なので、これも XOR の形すなわち  $s$  と  $C_{in}$  を入力とする半加算器の出力となっている。

次に  $C_{out}$  について見ていく。式 61 から以下のように変形する。

$$C_{out} = (\bar{A} \cdot B + A \cdot \bar{B}) \cdot C_{in} + A \cdot B \cdot (\overline{C_{in}} + C_{in}) \quad (70)$$

$$= (\bar{A} \cdot B + A \cdot \bar{B}) \cdot C_{in} + A \cdot B \quad (71)$$

右辺第 1 項の  $(\bar{A} \cdot B + A \cdot \bar{B})$  は  $s$  であった。また右辺第 2 項の  $A \cdot B$  は  $A$  と  $B$  の AND であるが、半加算器の桁上げ出力  $c_{out}$  も  $A \cdot B$  であった (式 60)。したがって、

$$C_{out} = s \cdot C_{in} + c_{out} \quad (72)$$

となり、 $s$  と  $c_{out}$  で半加算器を利用できることがわかる。

式 69 と式 72 から半加算器 (half adder, HA と記す) を用いた全加算器をブロック図で表わすと図 27 のようになる。式 69 と式 72 を XOR と AND を使って回路図で表わすと図 28 のようになる。

#### 5.5.4 NAND 化した全加算器

図 28 で示した全加算器を NAND 化して簡略化したものを図 29 に示す。NAND ゲート 9 個を用いて作ることができる。

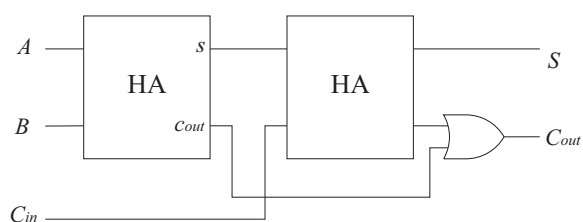


図 27: 半加算器を利用した全加算器

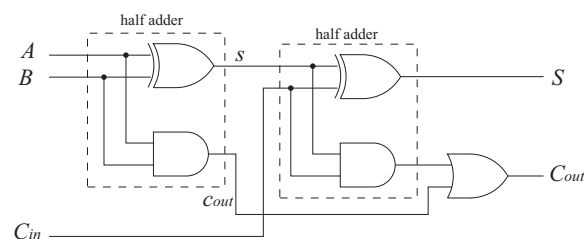


図 28: 半加算器を利用した全加算器 (XOR, AND, OR)

**練習 18.** 図 28 を NAND で置き換えて，簡略化すると図 29 になることを各自確かめよ．

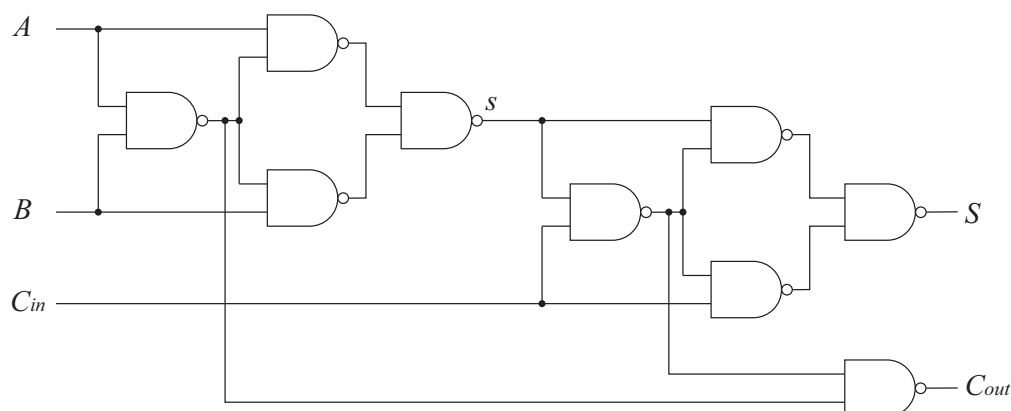


図 29: NAND 化した全加算器

### 5.5.5 半加算器と全加算器による多桁の2進加算

最下桁は半加算器を，それより上位の桁は全加算器を用いて多桁の加算器を作ることができる．これを図 30 に示す．図中の HA は半加算器 (half adder)，FA は全加算器 (full adder) を示している．

**発展 3.** (減算器) 引き算をする減算器 (並列減算器) について調べてみよ．

### 5.5.6 多数決回路再考

15 ページの式 54 および 21 ページの式 62 で見たとおり，全加算器の桁上げ出力  $C_{out}$  は 3 入力の多数決回路の出力と同じである．図 29 の NAND 化した全加算器で  $C_{out}$  に関係するところを抜き出せば，これも 3 入力多数決回路となる．これを図 31 に示す．図 19 (17 ページ) と同じく NAND6 個で構成できる．

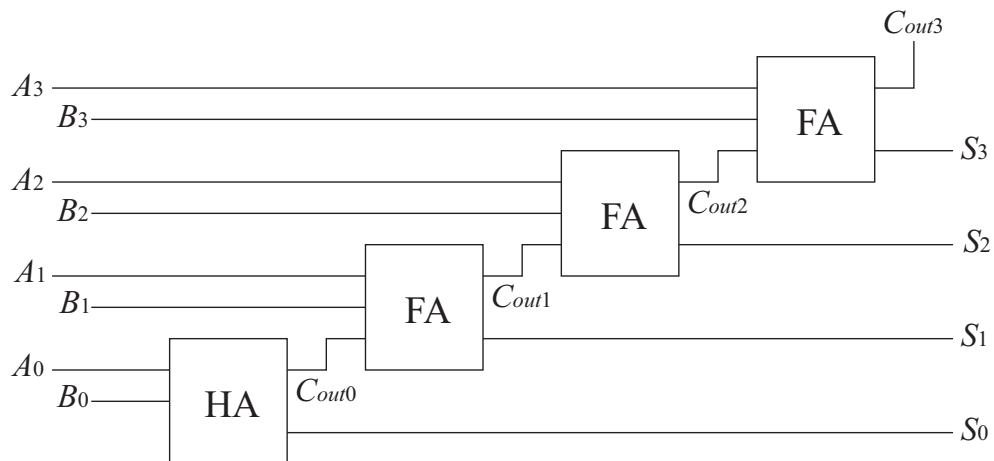


図 30: 並列加算器

**発展 4.** 3入力多数決回路の2つの例（17ページの図19と24ページの図31）を様々な角度から比較せよ。ヒント：ファンイン、遅延時間

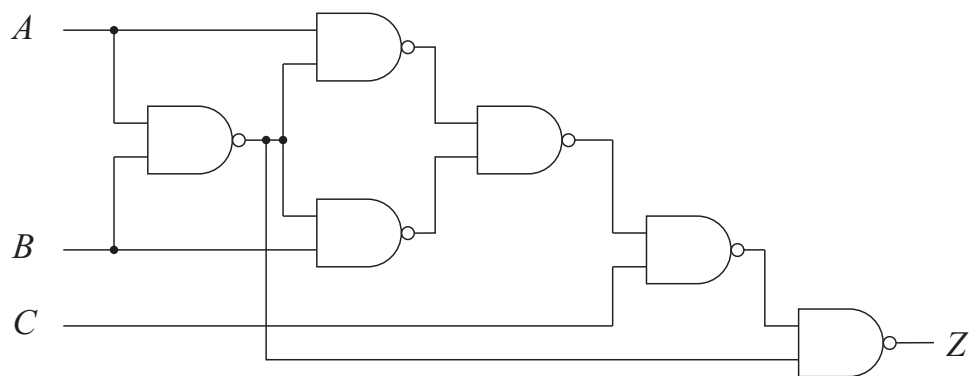


図 31: 別の3入力多数決回路 (NAND)

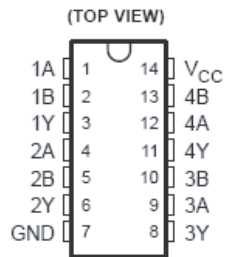
## 6 使用する NAND ゲート IC

今回の実習で使用する NAND ゲートは SN74AC00（又はその互換品）である。この IC は 2 入力 NAND ゲートが 4 つ入っている CMOS で、動作電圧は 2[V]~6[V] となっている。これを今回は乾電池 2 本（約 3[V]）で動作させる。



## 6.1 74AC00 について

Texas Instruments のデータシート [3] からピン配置, 真理値表, ロジックダイアグラムを図 32, 図 33, 図 34 に示す.



FUNCTION TABLE  
(each gate)

INPUTS		OUTPUT
A	B	Y
H	H	L
L	X	H
X	L	H



図 32: ピン配置 (文献 [3] より引用)

図 33: 真理値表 (文献 [3] より引用)

図 34: ダイアグラム (文献 [3] より引用)

74AC00 のピン配置をわかりやすく書き直すと図 35 のようになる.  $V_{CC}$  には電源のプラス側 (2[V] ~ 6[V]) を接続する. GND はグラウンドに接続する.

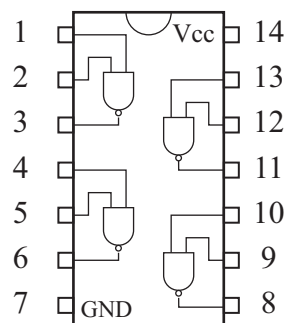


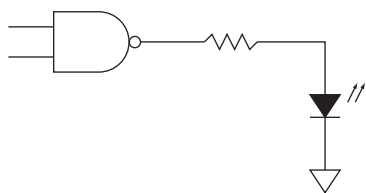
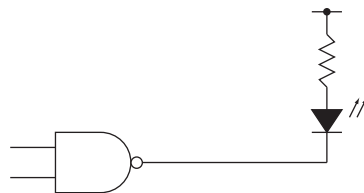
図 35: 74AC00 のピン配置と中のゲート

## 6.2 NAND の出力による LED の点灯

この 74AC00 は比較的大きな電流を出力できる. 図 36 や図 37 のような回路を用い, NAND の出力で LED を光らせることができる. 図 36 は NAND の出力が  $H$  つまり 1 のときに LED が点灯し, 図 37 では逆に出力が  $L$  すなわち 0 のときに LED が点灯する. 今回はわかりやすく, 図 36 のほうを用いる.

## 6.3 スイッチによる入力

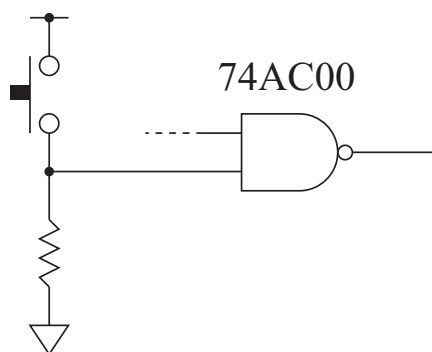
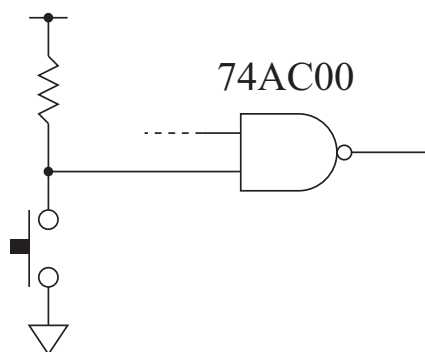
この実験で使用するタクトスイッチを使った入力回路の例を図 38 および図 39 に示す. 図 38 ではスイッチを押した時 (接点がつながり)  $H$  すなわち 1 が入力される. 図 39 では逆にスイッチを押し

図 36:  $H$  で LED が点灯図 37:  $L$  で LED が点灯

た時、 $L$ つまり0が入力される。ここではわかりやすい図38を使用することとする。図中のプルダウン抵抗、プルアップ抵抗は必ず使用すること。図からも明らかなように、抵抗を使用しないで電源またはグランドと直結すると、スイッチを押した時、電源をショートさせてしまうことになり、大変危険である。

今回はゲート回路へスイッチで入力しLEDで出力を見るという単純な回路なので、図38や図39の入力回路を使用できるが、フリップフロップのような回路への入力に使用すると問題が生じる。このタクトスイッチは機械接点なので、押したり離したりした時に、きわめて短い時間（ミリ秒のオーダー）に接点がON/OFFを繰り返すという現象が起こる。これを**チャタリング**という。チャタリングが問題となる場合には、その影響をなくす回路を使用したり、マイコンへの入力の場合にはソフト側で対応したりする。

**発展 5.** チャタリングおよびその防止方法について調べよ。

図 38: スイッチ ON で  $H$  を入力図 39: スイッチ ON で  $L$  を入力

## 6.4 バイパスコンデンサ（パスコン）について

IC1 つずつに必ず付けること。これを付けないとノイズによる誤動作が起きることがある。注意点はICの電源ピン（Vcc）とグランド（GND）の間に最短距離で付けることである。離れた場所に線をのばして取り付けてもパスコンの効果が発揮されない。

## 6.5 CMOS ロジックの未使用入力の処理

今回使用する NAND ゲート IC SN74AC00 は CMOS ロジックであり、**使用しない入力は GND または Vcc に接続しておく**。未使用の入力を開放のままにしておくと、誤動作や消費電流の増加、IC の破壊が起こることがある。余った NAND ゲートの入力は必ず上記のように処理しておくこと。なおこれは入力についてであり、余ったゲートの出力をこのように処理してはいけない（図 40、図 41 参照）。

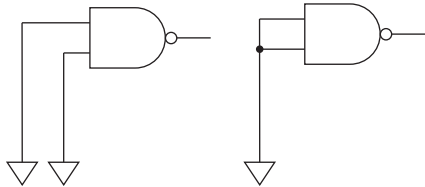


図 40: CMOS の未使用入力の処理 (GND)

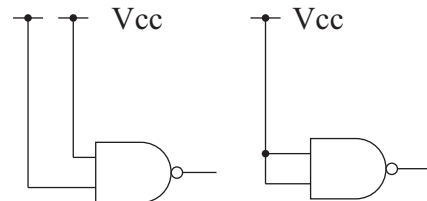


図 41: CMOS の未使用入力の処理 (Vcc)

## 7 この実験で用いる回路について

図??に今回の実験で制作する回路の全体を示す。図中の四角の空欄の中には、各自が下記のうちからひとつ選んだ回路をいれる。

- 自分で設計した（本資料に載っていない）オリジナルな回路
- 2進数デコーダ回路
- 多数決回路
- パリティチェック回路
- 加算器（半加算器，全加算器）を用いた加算回路
- AND を NAND ゲートのみを使って表わしたもの
- OR を NAND ゲートのみを使って表わしたもの
- NOR を NAND ゲートのみを使って表わしたもの
- XOR を NAND ゲートのみを使って表わしたもの

ただし使用できる 74AC00 は 1 個以上 3 個以下なので，NAND ゲート 12 個までで表わされる回路となる。

## 7.1 回路図の作成

実習に先立って、まずどの回路を作るか選択する（あるいは自分のオリジナル回路を設計しても良い）。選択（あるいは設計）した回路を図42に書き込む。図42には電池、電源スイッチ、入力スイッチ回路、出力LED回路がかき込まれている。電源は乾電池2本（3[V]）を使用する。電源と回路の間に電源スイッチがある。入力スイッチ回路、出力LED回路については各自の製作する回路に応じて必要数が変わるので、余った分は組み立てる必要はなく、足りない場合には適宜、図に追加すること。

次に、図43はIC（74AC00）をピン配置どおりに記載した同じ回路図であるので、図42にもとづき74AC00まわりの結線を図43に書き込む。バイパスコンデンサについてはすでに関き込んである。使用しないゲートの入力の空きピン処理（Vcc かグランドにつなぐ）も忘れずに関きすること。入力スイッチ回路、出力LED回路については各自の製作する回路の必要数に応じて適宜、消去または追加すること。

電池のプラス側から電源スイッチを通して上に伸びて短い横棒につながっている部分は+3[V]を表わしている。図中の同じこの記号どうしは組み立てのときはすべてつなぐ。同じくグランド（GND）を示す下向き三角（▽）どうしも、組み立てのときはすべて接続すること。

## 8 組み立て実習について

組み立て実習でこの回路を組み立てて、動作の確認を行なう。この回路から実際の部品は位置を考え、各自で1つ組み立てる。実習の資料の注意を良く読み、ケガ等の無いように気をつけて組み立てる。

### 8.1 注意事項

#### 8.1.1 安全上の注意

資料「電子回路組み立て」に示される安全上の注意を遵守すること。

#### 8.1.2 静電気による破壊について

CMOSの素子は静電気により破壊されることがある。直接手に持つ場合には気をつけること。通常は黒い導電性のスポンジに刺してある。

#### 8.1.3 製作した回路の安全な取り扱いについて

製作した回路は基板がむき出しになっている。そのため電池を入れたままで基板が金属に接触すると（スイッチがOFFでも）ショートして発熱、発火、破裂の恐れがある。運ぶ時には不用意にカバン等に放り込まず、必ず電池を電池ボックスから抜いて運搬する。また電池もショートさせると危険なので、持ち運び時には注意すること。

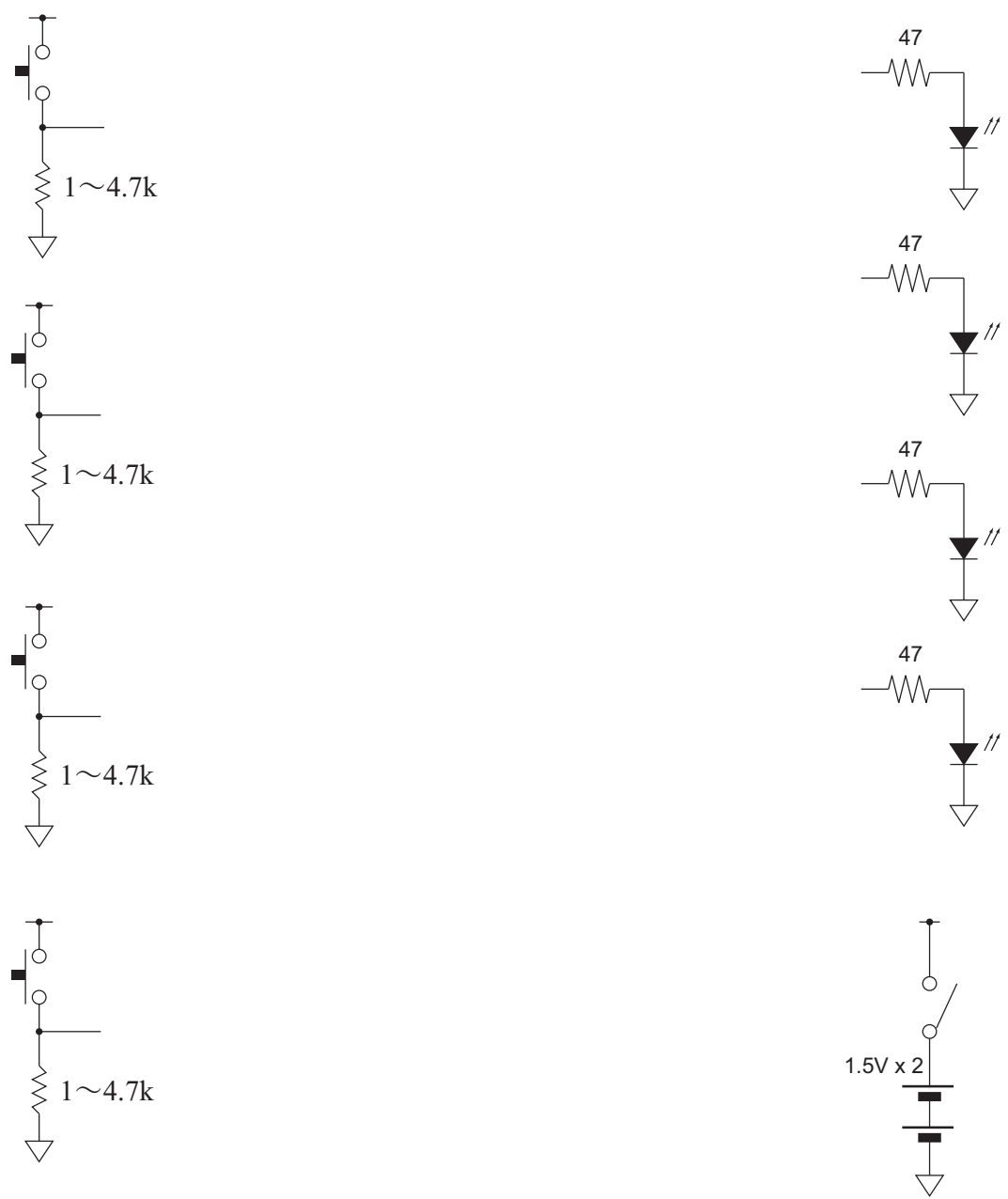


図 42: 実験の回路図 (その 1)

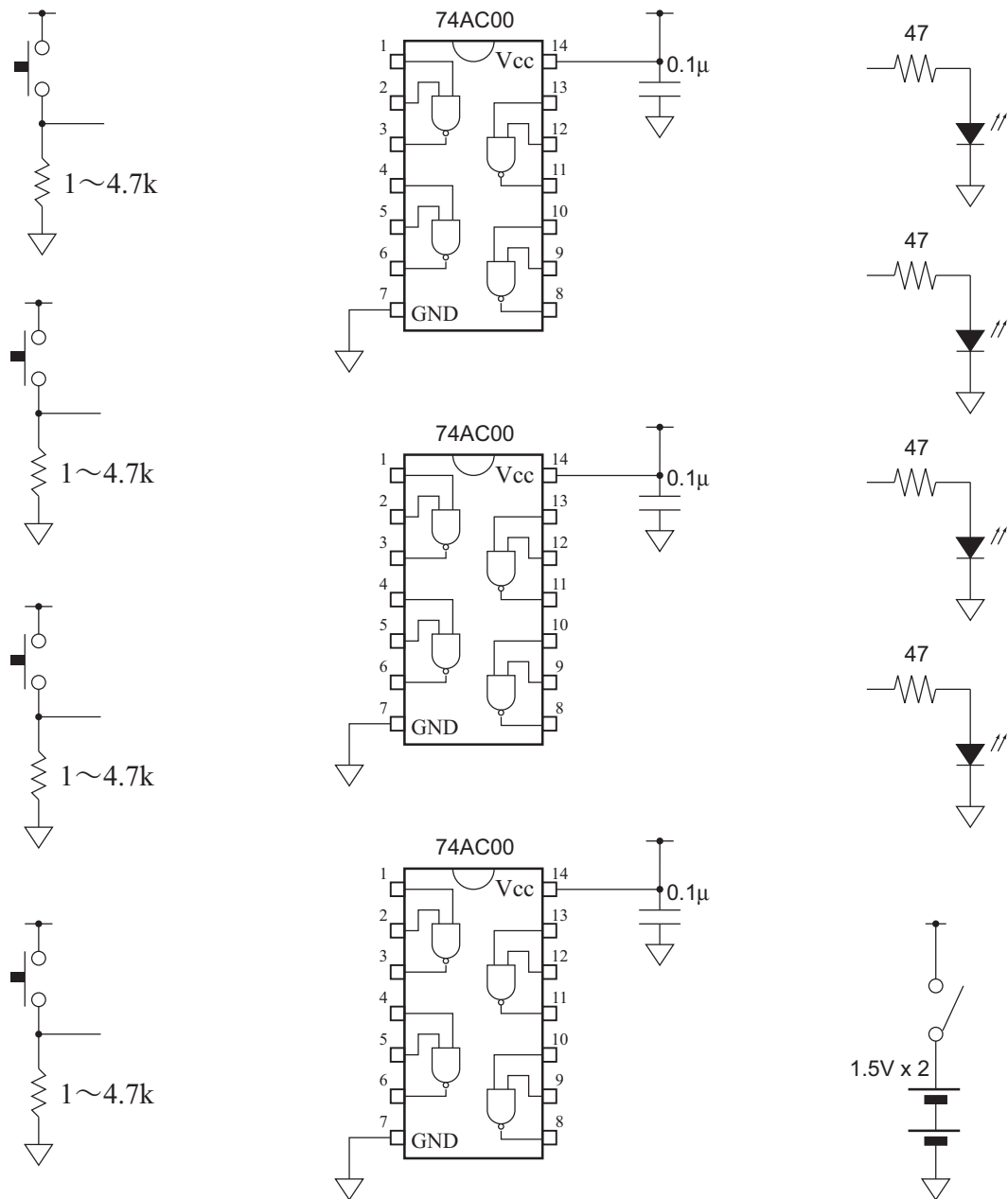


図 43: 実験の回路図 (その 2)

## 参考文献

- [1] 西堀賢司, 1993. メカトロニクスのための電子回路基礎, コロナ社.
- [2] 猪飼國夫, 本田中二, 1990. 定本ディジタル・システムの設計, CQ 出版社.
- [3] Texas Instruments, SN54AC00, SN74AC00 QUADRUPLE 2 INPUT POSITIVE NAND GATES, (データシート) .  
<http://focus.tij.co.jp/jp/lit/ds/symlink/sn74ac00.pdf>
- [4] 橋本順次, 1977. デジタル技術実戦教室 (電子展望別冊) , 誠文堂新光社.
- [5] 井澤裕司. 論理回路 1, 信州大学. [http://laputa.cs.shinshu-u.ac.jp/yizawa/logic\\_textbook.html](http://laputa.cs.shinshu-u.ac.jp/yizawa/logic_textbook.html)

(2011 年度版)