

# レポート提出票

科目名: 情報工学実験1

実験課題名: 課題3 論理回路

実施日: 2020年 6月 15日

学籍番号: 4619055

氏名: 辰川力駆

共同実験者:

_____	_____
_____	_____
_____	_____
_____	_____

# 1 実験の要旨

背景

## 2 実験の目的

デジタル回路の設計・解析に必要な基本となるゲート素子 (AND、OR、NOT、NAND、NOR、EX-OR) の基礎的動作原理を理解し、その応用について考察する。

## 3 実験概要

本実験は論理回路実習装置 ITF-02 を用いて行う。この装置の使用に当たっては次の注意を守る。

- パネル面での結線は必ず電源スイッチをオフにしておく。
- リードチップの抜き差しはプラグの部分を持って行う。リード線を持って抜き差しすると断線の原因になる。
- パネル面での結線を行う場合には、リードチップがからまないようにその結線に合ったリードチップを使用する。

実験は以下に示す順序で行うこと。

- (1) 論理積 (AND) 回路
- (2) 論理和 (OR) 回路
- (3) 否定 (NOT) 回路
- (4) 論理積の否定 (NAND) 回路
- (5) 論理和の否定 (NOR) 回路
- (6) ド・モルガンの定理の証明
- (7) 排他的論理和 (EX-OR) 回路
- (8) 加算器 (ADDER) の実習
- (9) デコーダの実習
- (10) R-S フリップ・フロップ
- (11) J-K フリップ・フロップ

## 4 実験

### 4.1 操作手順

本器 ITF-02 の基本的な操作手順は、次の通りとする。

1. 電源スイッチを OFF にする。
2. 各実習項目における結線を行う。
3. 電源スイッチを ON にする。
4. 各実習項目における実習を行う。
5. 実習が終了したら電源スイッチを OFF にして、結線を解く。

#### <注意>

結線を行ったり、結線を解いたりするときは、原則として電源スイッチを OFF にしておくこと。ただし、実習を行っている途中で結線を変えたり、結線を増やしたりするときは、その都度電源スイッチを OFF にする必要はない。その場合には、出力信号をアースに短絡したり、出力端子と出力端子を接続したりしないよう注意すること。結線を途中で変えるときは、信号出力端子に差し込んであるリードチップを抜き、次に信号入力端子に差し込んであるリードチップを抜く。また、結線を追加するときは、リードチップを信号入力端子に差し込み、次に信号出力端子に差し込む。

### 4.2 組み合わせ回路の実習

組み合わせ回路は、出力が入力だけに関係する論理回路で、基本になる素子として、論理積 (AND)、論理和 (OR)、否定 (NOT)、論理積の否定 (NAND)、論理和の否定 (NOR) などがあり、その応用として排他的論理和 (Exclusive-OR)、半加算器 (Half-ADDER)、全加算器 (Full-ADDER)、エンコーダ、デコーダなどがある。

#### (1) 論理積 (AND) 回路

- 目的

$Y = A \cdot B$  を理解する。

- 理論

論理積は、 $Y = A \cdot B$  で表現され、入力  $A$  と  $B$  がいずれも “1” のときのみ、出力  $Y$  が “1”、他の条件ではすべて “0” となるもので、この式を満足する論理回路を AND 回路という。

表 1: AND の真理値表

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

- 実習

パネル上の AND 回路素子を使用し、 $Y = A \cdot B$  の真理値表 1 を表示器で表示して確認することにより行う。

A、B の入力レベルは、設定スイッチにより設定する。

## (2) 論理和 (OR) 回路

- 目的

$Y = A + B$  を理解する。

- 理論

論理和は、 $Y = A + B$  で表現され、入力 A と B がいずれも “0” のときのみ、出力 Y が “0”、他の条件ではすべて “1” となるもので、この式を満足する論理回路を OR 回路という。

表 2: OR の真理値表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

- 実習

パネル上の OR 回路素子を使用し、 $Y = A + B$  の真理値表 2 を表示器で表示して確認することにより行う。

A、B の入力レベルは、設定スイッチにより設定する。

## (3) 否定 (NOT) 回路

- 目的

$Y = \overline{A}$  を理解する。

- 理論

否定回路は、インバータとも言われ、 $Y = \overline{A}$ で表現される。入力と出力の関係は常に正反対になり、この式を満足する論理回路を否定回路という。

表 3: NOT の真理値表

A	Y
0	1
1	0

- 実習

パネル上の NOT 回路素子を使用し、入力  $A$  に対し出力  $Y = \overline{A}$  の真理値表 3 を表示器で表示して確認することにより行う。

$A$ 、 $B$  の入力レベルは、設定スイッチにより設定する。

#### (4) 論理積の否定 (NAND) 回路

- 目的

$Y = \overline{A \cdot B}$  を理解する。

- 理論

論理積の否定は、 $Y = \overline{A \cdot B}$ で表現され、入力  $A$  と  $B$  がいずれも “1” のときのみ、出力  $Y$  が “0”、他の条件ではすべて “1” となるもので、この式を満足する論理回路を NAND 回路という。

表 4: NAND の真理値表

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

- 実習

パネル上の NAND 回路素子を使用し、 $Y = \overline{A \cdot B}$  の真理値表 4 を表示器で表示して確認することにより行う。

$A$ 、 $B$  の入力レベルは、設定スイッチにより設定する。

## (5) 論理和の否定 (NOR) 回路

- 目的

$Y = \overline{A + B}$  を理解する。

- 理論

論理和の否定は、 $Y = \overline{A + B}$  で表現され、入力  $A$  と  $B$  がいずれも “0” のときのみ、出力  $Y$  が “1”、他の条件ではすべて “0” となるもので、この式を満足する論理回路を NOR 回路という。

表 5: NOR の真理値表

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

- 実習

パネル上の NOR 回路素子を使用し、 $Y = \overline{A + B}$  の真理値表 5 を表示器で表示して確認することにより行う。

$A$ 、 $B$  の入力レベルは、設定スイッチにより設定する。

## (6) ド・モルガンの定理の証明

- 目的

ド・モルガンの定理証明として、

$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

の式に、実際の値を入れて行う。

- 理論

ド・モルガンの定理は、式 (1)、および式 (2) で表示される。

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

$$\overline{A + B} = \overline{A} \cdot \overline{B} \quad (2)$$

式 (1)、および式 (2) の証明の際し、上記を書き直すと、式 (3)、および式 (4) になる。

$$Y_1 = \overline{A \cdot B} \quad Y_2 = \overline{A} + \overline{B} \quad Y_1 = Y_2 \quad (3)$$

$$Y_3 = \overline{A + B} \quad Y_4 = \overline{A} \cdot \overline{B} \quad Y_3 = Y_4 \quad (4)$$

表 6:  $Y_1 = \overline{A \cdot B}$ ,  $Y_2 = \overline{A} + \overline{B}$  の真理値表

$A$	$B$	$Y_1$	$Y_2$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

表 7:  $Y_3 = \overline{A + B}$ ,  $Y_4 = \overline{A} \cdot \overline{B}$  の真理値表

$A$	$B$	$Y_3$	$Y_4$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

- 実習

論理回路をパネル上で構成し、それぞれの真理値表6,7を表示器で表示して確認し、 $Y_1 = Y_2$ 、 $Y_3 = Y_4$ であれば証明が成立したという方法で行う。

## (7) 排他的論理和 (Exclusive-OR) 回路

- 目的

$Y = \overline{A} \cdot B + A \cdot \overline{B} = A \oplus B$  を理解する。

- 理論

排他的論理和は、 $Y = \overline{A} \cdot B + A \cdot \overline{B} = A \oplus B$  で表現され、入力  $A$  と  $B$  が同じレベルのとき、出力  $Y$  が “0”、異なるレベルのときは “1” となるもので、この式を満足する論理回路を Exclusive-OR 回路という。

表 8:  $Y = \overline{A} \cdot B + A \cdot \overline{B} = A \oplus B$  の真理値表

$A$	$B$	$Y$
0	0	0
0	1	1
1	0	1
1	1	0

- 実習

論理回路をパネル上で構成し、真理値表 8 を表示器で表示して確認することにより行う。

A、B の入力レベルは、設定スイッチにより設定する。

## (8) 加算器 (ADDER)

- 目的

1. 加算器の  $S$  の部分が排他的論理和 (Exclusive-OR) であることを理解する。
2. 半加算器の動作を理解する。

- 理論

加算器には、下位からの桁上げを考慮しない半加算器 (Half-ADDER) と、下位からの桁上げを考慮する全加算器 (Full-ADDER) とがある。全加算器は、半加算器を 2 個、OR 回路を 1 個直列に接続した形になる。

半加算器の論理式は、次の式で与えられる。

$$S = \overline{A} \cdot B + A \cdot \overline{B} \quad (5)$$

$$C = A \cdot B \quad (6)$$

または、

$$S = A \oplus B \quad (7)$$

式 (5)、式 (7) から分かるように、回路の和 (Sum) を構成している部分は、Exclusive-OR になる。

全加算器の論理式は、次の式で与えられる。

$$\begin{aligned} S &= \overline{A} \cdot \overline{B} \cdot C_i + \overline{A} \cdot B \cdot \overline{C_i} + A \cdot \overline{B} \cdot \overline{C_i} + A \cdot B \cdot C_i \\ C_O &= \overline{A} \cdot B \cdot C_i + A \cdot \overline{B} \cdot C_i + A \cdot B \cdot \overline{C_i} + A \cdot B \cdot C_i \\ &= A \cdot B + B \cdot C_i + A \cdot C_i \end{aligned}$$

いま、 $S_1 = \overline{A} \cdot B + A \cdot \overline{B}$  とすれば、

$$S = \overline{S_1} \cdot C_i + S_1 \cdot \overline{C_i}$$

また、 $C_1 = A \cdot B$ 、 $C_2 = S_1 \cdot C_i$  とすれば、

$$\begin{aligned} C_O &= C_1 + C_2 \\ &= A \cdot B + S_1 \cdot C_i \end{aligned}$$

となり、さらに、Exclusive-OR を用いて表せば、

$$\begin{aligned} S &= A \oplus B \oplus C_i \\ C_O &= A \cdot B + (A \oplus B) \cdot C_i \end{aligned}$$

となる。



表 9: Half-ADDER の真理値表

$A$	$B$	$C$	$S$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

表 10: Full-ADDER の真理値表

$A$	$B$	$C_i$	$C_o$	$S$
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

- 実習

論理回路をパネル上で構成し、真理値表 9、10 を表示器で表示して確認することにより行う。

## (9) デコーダ (DECODER)

- 目的

2 進数を 10 進数に変換する動作を理解する。

- 理論

4 ビットの 2 進数コードを、もとの 10 進数に戻すようなコード翻訳動作をする論理回路を、デコーダといい、2 進数と 10 進数の関係式は、次のようになる。

2進数コード  $A = 2^0$  ビット、 $B = 2^1$  ビット  
 $C = 2^2$  ビット、 $D = 2^3$  ビット

10進数 “0” ~ “9” として

$$\begin{aligned} \text{“0”} &= \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} \\ \text{“1”} &= A \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} \\ \text{“2”} &= \overline{A} \cdot B \cdot \overline{C} \cdot \overline{D} \\ \text{“3”} &= A \cdot B \cdot \overline{C} \cdot \overline{D} \\ \text{“4”} &= \overline{A} \cdot \overline{B} \cdot C \cdot \overline{D} \\ \text{“5”} &= A \cdot \overline{B} \cdot C \cdot \overline{D} \\ \text{“6”} &= \overline{A} \cdot B \cdot C \cdot \overline{D} \\ \text{“7”} &= A \cdot B \cdot C \cdot \overline{D} \\ \text{“8”} &= \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot D \\ \text{“9”} &= A \cdot \overline{B} \cdot \overline{C} \cdot D \end{aligned}$$

この式を満足する真理値表を表 11 に示す。

表 11: デコーダ (2進-10進) の真理値表

	2進数				10進数									
	$D$	$C$	$B$	$A$	“0”	“1”	“2”	“3”	“4”	“5”	“6”	“7”	“8”	“9”
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	1	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	1	0	0	0	0	0	0	0
3	0	0	1	1	0	0	0	1	0	0	0	0	0	0
4	0	1	0	0	0	0	0	0	1	0	0	0	0	0
5	0	1	0	1	0	0	0	0	0	1	0	0	0	0
6	0	1	1	0	0	0	0	0	0	0	1	0	0	0
7	0	1	1	1	0	0	0	0	0	0	0	1	0	0
8	1	0	0	0	0	0	0	0	0	0	0	0	1	0
9	1	0	0	1	0	0	0	0	0	0	0	0	0	1

- 実習

パネル上のデコーダ部分を使用し、真理値表 11 を表示器で表示して確認することにより行う。

### 4.3 順序回路の実習

順序回路 (Sequential Circuit) は、出力が入力だけでなく回路そのものの状態によって左右される論理回路で、R-S フリップ・フロップ回路 (R-S Flip・Flop Circuit)、J-K フリップ・フロップ回路 (J-K Flip・Flop Circuit)、シフトレジスタ (Shift Register) などがあり、その応用として  $n$  進カウンタや、 $n$  ビットのシフトレジスタがある。

#### (10) R-S フリップ・フロップ回路

- 目的

1. R-S フリップ・フロップが1ビットの記憶素子であることを理解する。
2. R-S フリップ・フロップにおいて禁止とされる入力が存在することを理解する。

- 理論

R-S フリップ・フロップ回路は、“0”、または“1”の論理レベルを記録する機能を持った回路である。論理式、シンボル、真理値表は各々式 (8)、式 (9)、表 12 で表される。

$$Q^{(n+1)} = \overline{S^{(n)} \cdot \overline{Q^{(n)}}} \quad (8)$$

$$\overline{Q}^{(n+1)} = \overline{\overline{R^{(n)}} \cdot Q^{(n)}} \quad (9)$$

ただし時刻  $n$  に関して、 $S^{(n)}$ 、 $R^{(n)}$ 、および  $Q^{(n)}$ 、 $\overline{Q}^{(n)}$  は入力  $S$ 、 $R$  と出力  $Q$ 、 $\overline{Q}$  を表すものとする。

動作は、 $Q$  と  $\overline{Q}$  が相補の関係 ( $Q$  と  $\overline{Q}$  が互いに異なる値を持つ) にあるとき、 $R$  と  $S$  が“0”レベルの時、出力はもとの状態を保持し、 $S$  が“1”、 $R$  が“0”なら出力  $Q$  は“1”、 $S$  が“0”、 $R$  が“1”なら出力  $Q$  は“0”にそれぞれ落ち着く。しかしながら  $S$  と  $R$  が共に“1”の場合は  $Q$  と  $\overline{Q}$  が共に“1”となり、相補の関係が満たされなくなる。この状態において、 $R$  と  $S$  を同時に“0”レベルにすると、 $Q$  と  $\overline{Q}$  が“0”と“1”のレベルを交互に繰り返すこととなる。(ただし、実際には各 NAND 素子の応答速度の差異や配線の長さによって、 $Q$  と  $\overline{Q}$  が相補になるように落ち着く。) このため、R-S フリップ・フロップ回路では、 $S$  と  $R$  を共に“1”として入力することを“禁止”としている場合が多い。

表 12: R-S フリップ・フロップの真理値表

$S^{(n)}$	$R^{(n)}$	$Q^{(n+1)}$	$\overline{Q}^{(n+1)}$	
0	0	$Q^{(n)}$	$\overline{Q}^{(n)}$	
0	1	0	1	
1	0	1	0	
1	1	1	1	禁止

- 実習

パネル上の R-S フリップ・フロップの素子を使用し、 $S$ 、 $R$  の入力レベルに対する出力レベル  $Q$ 、 $\overline{Q}$  のレベルを表示器で確認することにより行う。

## (11) J-K フリップ・フロップ回路

### ● 目的

1. J-K フリップ・フロップがトリガ型フリップ・フロップ回路であることを理解する。
2. J、K をともに “1” としたときには、T フリップ・フロップにもなることを理解する。

### ● 理論

R-S フリップ・フロップ回路は、 $R$ 、 $S$  のレベルが直接出力を決定するのに対して、J-K フリップ・フロップ回路は、 $J$  と  $K$  のレベルの他に、トリガが加えられないと出力が決定されない、トリガ型のフリップ・フロップ回路の一種である。

論理式、真理値表は、各々式 (10)、(11)、表 13 で表される。

$$Q^{(n+1)} = \overline{K^{(n)}} \cdot Q^{(n)} + J^{(n)} \cdot \overline{Q}^{(n)} \quad (10)$$

$$\overline{Q}^{(n+1)} = \overline{Q^{(n+1)}} \quad (11)$$

動作は次のようになる。

1.  $J$  と  $K$  が “0” レベルのときの出力  $Q$  は、トリガパルス  $T$  が加えられても元の状態を保持する。
2.  $J$  が “1”、 $K$  が “0” のときの出力  $Q$  は、トリガパルス  $T$  が加えられると “1” になり、この状態でさらにトリガパルスが加えられても、元の状態 (“1”) を保持する。
3.  $J$  が “0”、 $K$  が “1” のときの出力  $Q$  は、トリガパルス  $T$  が加えられると “0” となり、この状態でさらにトリガパルスが加えられても、元の状態 (“0”) を保持する。
4.  $J$  と  $K$  が “0” レベルのときの出力  $Q$  はトリガパルス  $T$  が加えられるごとにレベルが反転する。(T フリップ・フロップ: トグルフリップ・フロップ)
5.  $PC$  を “0” にすると、 $J$ 、 $K$ 、 $T$  のレベルに関係なく、出力  $Q$  は “0” になる。(Pre-Clear)

なお、トリガパルス入力端子の先端は否定回路同様に “○” と書かれるが、これはトリガパルスの下降部でトリガされることを意味する。

表 13: J-K フリップ・フロップの真理値表

$J^{(n)}$	$K^{(n)}$	$T$	$Q^{(n+1)}$	$\overline{Q}^{(n+1)}$	動作
0	0	↓	$Q^{(n)}$	$\overline{Q}^{(n)}$	ホールド
0	1	↓	0	1	リセット
1	0	↓	1	0	セット
1	1	↓	$\overline{Q}^{(n)}$	$Q^{(n)}$	トグル

$PC$	$Q^{(n+1)}$
0	0
1	$Q^{(n)}$

- 実習

パネル上の J-K フリップ・フロップの素子を使用し、 $J$ 、 $K$  の入力に対してトリガパルスを加えたときの出力  $Q$ 、 $\overline{Q}$  のレベルを表示器で確認することにより行う。

トリガパルスは、パネル上のパルス発生器から、手動による単発パルスを用いる。また、 $PC$  端子には、同じパルス発生器のクロックパルスを使用する。

## 5 検討・考察

## 6 結論

## 参考文献

- [1] J. J. Collins et al., *PRE*, **52**(4):R3321, 1995.
- [2] E. M. Izhikevich, *IEEE Trans. NN*, **14**(6):1569, 2003.