情報工学実験1
基本ゲート回路

学籍番号 045718D : 翁長絵美、グループ C

実地日 : 平成 17 年 5 月 31 日
実地日 : 平成 17 年 6 月 7 日

共同実験者
沖津　望 : 045717F
小野裕作 : 045719B
狩野　昴 : 045720F

1　実験目的

現代社会に欠かすことのできないコンピュータは、大規模なディジタル回路によって構成されている。本実験では、ディジタル回路の構成要素である基本ゲートと理路演算の基礎を習得することを目的とする。また本実験では、NAND ゲートを用いて他のゲート回路 (NOT,AND,OR,NOR,XOR) を構成することによって、汎用ロジック IC の基本的な使い方についても学ぶ。

2　使用した器具

- ダイオード
- ブレッドボード
- 直流電源 (ERS01A)
- IC(TC4011BP)

3　報告事項

3.1　各実験について結果を報告しなさい。

(1)NAND ゲートのみを用いて、NOT,AND,OR,NOR,XOR ゲートを設計せよ。

それぞれのゲートを設計するにあたって、NAND 回路の真理値表と MIL 記号を以下に表す。

表 1: NAND の真理値表

A	B	f
0	0	1
0	1	1
1	0	1
1	1	0

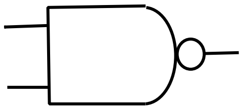

図 1: NAND の MIL 記号

NAND ゲートを用いて MIL 回路でそれぞれの設計を描くと以下のようになる。

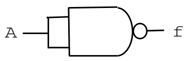

図 2: NOT 回路

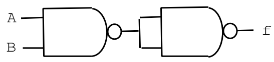

図 3: AND 回路

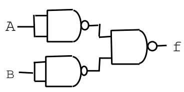

図 4: OR 回路

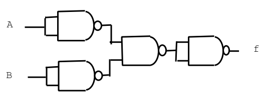

図 5: NOR 回路

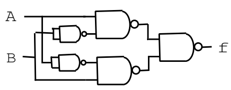

図 6: XOR 回路

真理値表を書いて確認するとこの MIL 回路で正しいことがわかる。論理式は以下のようになる。

<NOT ゲート>

$$
\begin{aligned}
f &= (\overline{A \cdot A}) \\
&= \overline{A} + \overline{A} \\
&= \overline{A}
\end{aligned}
$$

<AND ゲート>

$$
\begin{aligned}
f &= (\overline{\overline{A \cdot B}}) \\
&= A \cdot B
\end{aligned}
$$

<OR ゲート>

$$\begin{aligned} f &= (\overline{\overline{A}\cdot\overline{B}}) \\ &= \overline{\overline{A}}+\overline{\overline{B}} \\ &= A+B \end{aligned}$$

<NOR ゲート>

$$\begin{aligned} f &= (\overline{\overline{\overline{A}\cdot\overline{B}}}) \\ &= (\overline{\overline{\overline{A}}+\overline{\overline{B}}}) \\ &= (\overline{A+B}) \end{aligned}$$

<XOR ゲート>

$$\begin{aligned} f &= \overline{\overline{(\overline{A\cdot\overline{B}})\cdot(\overline{\overline{A}\cdot B})}} \\ &= (\overline{\overline{A}+B)\cdot(A+\overline{B}}) \\ &= (\overline{\overline{A}+B})+(\overline{A+\overline{B}}) \\ &= A\cdot\overline{B}+\overline{A}\cdot B \end{aligned}$$

(2) 実験 (1) で設計した各ゲートを実際に NAND ゲート IC を用いて実現し、それらの動作を確認せよ。

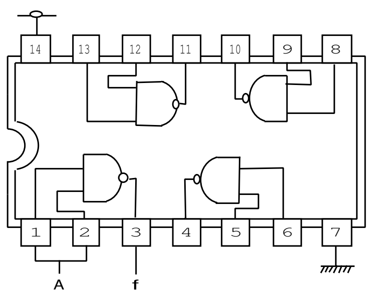

図 7: NOT 回路

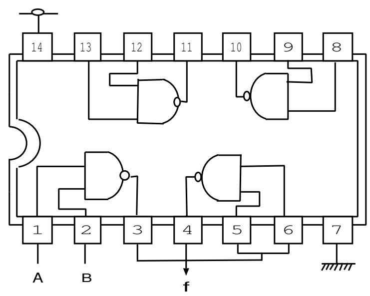

図 8: AND 回路

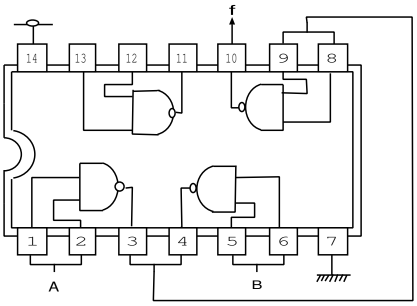

図 9: OR 回路

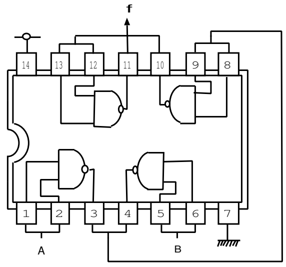

図 10: NOR 回路

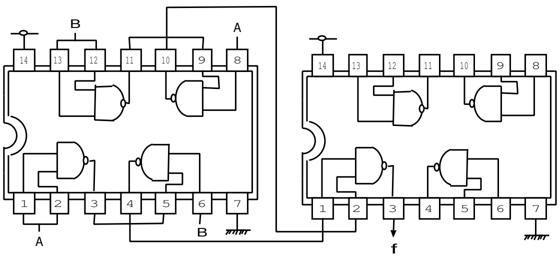

図 11: XOR 回路

NOT,AND,OR,NOR までは順調に真理値表の通りにダイオードのランプが付き正しく確認したが、最後の XOR のとき、回路の組み方は正しいのにランプがつかないという事態が起こった。よく観察してみると、ダイオードのつなぎ方が逆になっていた。この実験で NAND ゲート IC の動作を確認することと、ダイオードにもプラスとマイナスがあり、正しくつながないといけないということが確認できた。

3.2 ２変数の理論関数は全部で 16 種類ある。何故 16 種類になるか説明せよ。また、２変数の理論関数を 16 種類すべて列挙し、否定 (NOT), 論理積 (AND), および論理和 (OR) のみを用いて表現せよ。

２変数の入力方法は、(0,0),(0,1),(1,0),(1,1) と４通りある。また、４通りの出力の結果は'0' か'1' と２通りである。したがって、$2^4 = 16$ という計算になり、２変数の理論関数は全部で 16 種類あることがわかる。

表 2: ２変数の論理関数

論理式	(0,0)	(0,1)	(1,0)	(1,1)
A・B	0	0	0	0
A・B	0	0	0	1
A・$\overline{B}$	0	0	1	0
A	0	0	1	1
$\overline{A}$・B	0	1	0	0
B	0	1	0	1
A・$\overline{B}+\overline{A}$	0	1	1	0
A+B	0	1	1	1
$\overline{A}$・$\overline{A}$	1	0	0	0
A・B+$\overline{A}$・$\overline{A}$	1	0	0	1
$\overline{B}$	1	0	1	0
A+$\overline{B}$	1	0	1	1
$\overline{A}$	1	1	0	0
$\overline{A}$+B	1	1	0	1
$\overline{A}$+$\overline{B}$	1	1	1	0
A+$\overline{A}$+B+$\overline{B}$	1	1	1	1

3.3 NAND ゲート以外のゲート回路のうち、ただ１種類で NOT,AND,OR,NAND,NOR,XOR ゲートを表せるゲート回路の具体例を示せ。

ただ１種類で NOT,AND,OR,NAND,NOR,XOR ゲートを作るということは、ただ１種類で否定も表さないといけないということになる。したがって、肯定的である AND ゲートと OR ゲートでは１種類で作れないということがわかる。そこで、NOR を使用して作ることにした。式は以下の通りである。

<NOT 回路>

$$
\begin{aligned}
f &= (\overline{A+A}) \\
&= \overline{A}\cdot\overline{A} \\
&= \overline{A}
\end{aligned}
$$

<AND 回路>

$$
\begin{aligned}
f &= (\overline{\overline{A}+\overline{B}}) \\
&= \overline{\overline{A}}\cdot\overline{\overline{B}} \\
&= A\cdot B
\end{aligned}
$$

<OR 回路>

$$
\begin{aligned}
F &= (\overline{\overline{A+B}}) \\
&= A+B
\end{aligned}
$$

<NAND 回路>

$$
\begin{aligned}
f &= (\overline{\overline{\overline{A}+\overline{B}}}) \\
&= (\overline{\overline{\overline{A}}\cdot\overline{\overline{B}}}) \\
&= \overline{A\cdot B}
\end{aligned}
$$

<XOR 回路>

$$
\begin{aligned}
f &= \overline{(\overline{A+B})+(\overline{\overline{A}+\overline{B}})} \\
&= \overline{(\overline{A}\cdot\overline{B})+(A\cdot B)} \\
&= (A+B)\cdot(\overline{A}+\overline{B}) \\
&= A\cdot\overline{A}+A\cdot\overline{B}+\overline{A}\cdot B+B\cdot\overline{B} \\
&= A\cdot\overline{B}+\overline{A}\cdot B
\end{aligned}
$$

MIL 記号で表すと以下のようになる。

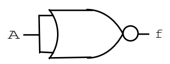

図 12: NOR で NOT 回路

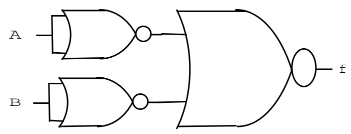

図 13: NOR で AND 回路

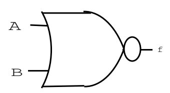

図 14: NOR 回路

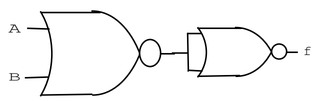

図 15: NOR で OR 回路

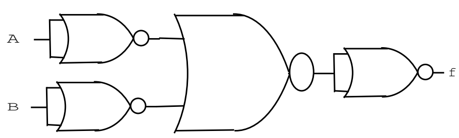

図 16: NOR で NAND 回路

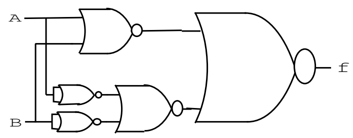

図 17: NOR で XOR 回路

3.4 ２種類のゲート回路でNOT,AND,OR,NAND, NOR,XORゲートを表せるゲート回路の組の具体例を２組以上示せ。

AND と NOT、OR と NOT の組み合わせで作ってみた。

<AND と NOT を使った場合>
2.1 の NAND ゲートを AND と NOT で表した。作ってみて感じたことは、AND と NOT で AND を作ることが無駄なような気がした。２種類のゲートで回路を作るなら、NAND と NOT など否定系のものを使ったほうが効率がいいことがわかった。

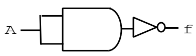

図 18: AND と NOT で NOT 回路

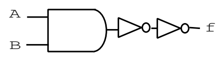

図 19: AND と NOT で AND 回路

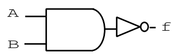

図 20: AND と NOT で NAND 回路

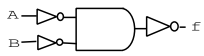

図 21: AND と NOT で OR 回路

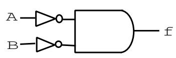

図 22: AND と NOT で NOR 回路

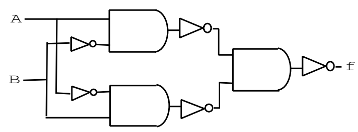

図 23: AND と NOT で XOR 回路

<OR と NOT を使った場合>
この場合は 3.3 で作った NOR 回路のものを OR と NOT で表した。

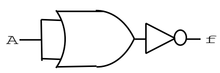

図 24: OR と NOT で NOT 回路

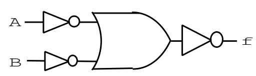

図 25: OR と NOT で AND 回路

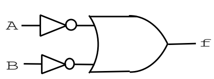

図 26: OR と NOT で NAND 回路

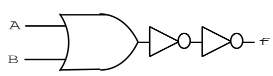

図 27: OR と NOT で OR 回路

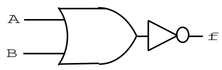

図 28: OR と NOT で NOR 回路

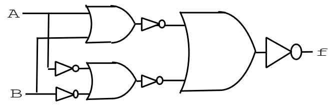

図 29: OR と NOT で XOR 回路

3.5 半加算器および全加算器とはどのような回路が調査し説明せよ。

<半加算器>
半加算器は 2 個の 1 桁の 2 進数を加算する論理回路である。半加算器はあくまで 1 桁の加算であり、下位からの桁あげは考慮しない。

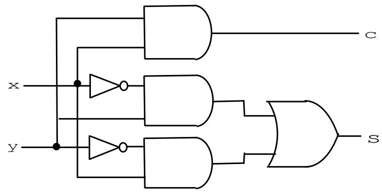

図 30: 半加算器の回路

表 3: 半加算器の真理値表

x	y	s	c
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

ここで、x と y は入力、s が出力、c は桁上げを表す。

<全加算器>
全加算器も 1 桁の 2 進数の加算を行うが、半加算器と違うのは、下位の桁からの桁上がりを含めることができる。

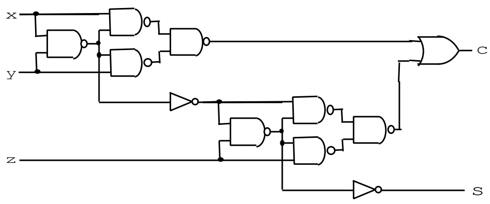

図 31: 全加算器の回路

表 4: 全加算器の真理値表

x	y	z	s	c
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	0
0	0	1	1	1
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

x、y、z が入力で、c、s が出力値である。s はその桁の計算結果となり、c は次ぎの位への桁上がりとなり、z と計算するというようになる。

3.6 本実験について考察せよ。

本実験は普段教科書で見て聞くだけであったゲート回路を実際に作るという作業を行った。それによって、実際に自らが設計した回路をもとにディジタル回路ができるという実感をもたらすことができる。今回の実験のレポートを書く際、Tgif の書き方も勉強になった。

参考文献

[1] 半加算器
http://www.infonet.co.jp/ueyama/ip/semi_cnd/adder_n.html

[2] 全加算器
http://mt-net.vis.ne.jp/ADFE_mail/0236.htm