組み合わせ回路の実現

学籍番号 045713C：大城 和也
グループ B:メンバー
045711G:上原 祐亮
045712E:大城 和輝
045715K:大城 悠

提出期限:平成 17 年 6 月 14 日 (火)

1 実験目的

本実験では、簡単な組み合わせ回路の設計および実現を行うことによって、カルノー図などを用いた論理関数の簡単化に慣れるとともに、実際のコンピュータに使用されている演算器の設計法について習得する。

2 使用器具

- Quad 2 Input XOR Gate (XOR ゲート)
- Quad 2 Input OR Gate(OR ゲート)
- Quad 2 Input AND Gate (AND ゲート)
- ブレッドボード
- 発光ダイオード
- その他 (配線・直流電源・実験指導書)

3 実験方法

3.1 半加算器、全加算器、2 ビット加算器の真理値表を書き、それぞれの論理関数を求めよ。

半加算器は 2 つの入力を足した値を出力するものである。よって桁上がりの分の出力 C がある。そのことを踏まえて真理値表を書く。

全加算器は 2 ビット以上の加算を行うことを考え入力に桁上がりの分の C_i を考えたもので半加算器を 2 つと OR 回路でも表現できる。この回路を考えて真理値表を書く。

上の全加算器と半加算器を用いて 2 ビットの加算が行える回路を考えて真理値表を書く。

求めた真理値表から加法標準形に直すことで論理関数を求める。

3.2 上記の実験で作成した各真理値表をもとにカルノー図を描き、簡単化された論理関数を求めよ。

真理値表から実験指導書にかかれている手順でカルノー図を描き、そして簡単化を行う。

3.3 上記 2 つの実験で得られた論理関数を比較し、2.2 での実験で得られた論理関数が簡単化されていることを確認せよ。

真理値表から加法標準形として得られた式と、カルノー図を用いて簡単化されているはずの式を見比べて簡単化されているかを確かめる。

3.4 実験 3.2 で得られた論理関数から、半加算器と全加算器の回路図を書け。

先ほどの実験らでもとめた関数を参考に回路図を構成する。

3.5 実験 2.4 で描いた回路図をもとにして、ブレッドボード上に半加算器を実現し、その動作を確認せよ。

全実験での回路図を IC とブレッドボード、発光ダイオードなどを用いて回路を実現する。今回の場合、出力は 2 つなので発光ダイオードは二つ必要となる。なお、IC における使わない回路の入力は確実にグランドか電源につなぐようにする。

3.6 ブレッドボード上に、もう一つ半加算器を実現し、実験指導書の図 3.5 の構成に基づいて全加算器を実現せよ。また、その動作を確認せよ。

まず、半加算器を二つ使う全加算器の回路図を作成する。それを前実験同様に IC を用いて実現する。

4　報告事項

4.1　各実験についての結果の報告

4.1.1

半加算器、全加算器、2 ビット加算器の真理値表は下の表 1、表 2、表 3 となった。

ちなみに表 3 における A_1、B_1 は 2 ビットにおける MSB を、A_0、B_0 は LSB をそれぞれ表すとする。

表 1: 半加算器

入力		出力	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

表 2: 全加算器

入力			出力	
A	B	C_i	S	C_o
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

表 3: 2 ビット加算器

入力				出力		
A_0	A_1	B_0	B_1	S_0	S_1	C
0	0	0	0	0	0	0
0	0	0	1	0	1	0
0	0	1	0	1	0	0
0	0	1	1	1	1	0
0	1	0	0	0	1	0
0	1	0	1	0	0	1
0	1	1	0	1	1	0
0	1	1	1	1	0	1
1	0	0	0	1	0	0
1	0	0	1	1	1	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	1	0
1	1	0	1	1	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	1

これらの真理値表を見て加法標準形で表すと下記のようになる。

半加算器

$$
\begin{aligned}
S &= A\overline{B} + \overline{A}B \\
C &= AB
\end{aligned}
$$

全加算器

$$
\begin{aligned}
S = \overline{A}\overline{B}C_i + \overline{A}B\overline{C_i} + A\overline{B}\overline{C_i} + ABC_i \\
C = \overline{A}BC_i + A\overline{B}C_i + AB\overline{C_i} + ABC_i
\end{aligned}
$$

2 ビット加算器

$$
\begin{aligned}
S_0 = {} & \overline{A_0}\ \overline{A_1}\ B_0\ \overline{B_1}\ + \overline{A_0}\ \overline{A_1}\ B_0\ B_1\ + \overline{A_0}\ A_1\ B_0\ \overline{B_1}\ + \overline{A_0}\ A_1\ B_0\ B_1\ + \\
& A_0\ \overline{A_1}\ \overline{B_0}\ \overline{B_1}\ + A_0\ \overline{A_1}\ \overline{B_0}\ B_1\ + A_0\ A_1\ \overline{B_0}\ \overline{B_1}\ + A_0\ A_1\ \overline{B_0}\ B_1 \\
S_1 = {} & \overline{A_0}\ \overline{A_1}\ \overline{B_0}\ B_1\ + \overline{A_0}\ \overline{A_1}\ B_0\ B_1\ + \overline{A_0}\ A_1\ \overline{B_0}\ \overline{B_1}\ + \overline{A_0}\ A_1\ B_0\ \overline{B_1}\ + \\
& A_0\ \overline{A_1}\ \overline{B_0}\ B_1\ + A_0\ \overline{A_1}\ B_0\ \overline{B_1}\ + A_0\ A_1\ \overline{B_0}\ \overline{B_1}\ + A_0\ A_1\ B_0\ B_1
\end{aligned}
$$

$$
\begin{aligned}
C = {} & \overline{A_0}\ A_1\ \overline{B_0}\ B_1\ + \overline{A_0}\ A_1\ B_0\ B_1\ + A_0\ \overline{A_1}\ B_0\ B_1 + A_0\ A_1\ \overline{B_0}\ B_1 + \\
& A_0\ A_1\ B_0\ \overline{B_1} + A_0\ A_1\ B_0\ B_1
\end{aligned}
$$

4.1.2

上記の実験により求まった真理値表よりカルノー図を作成し、簡単化する。
まずは半加算器

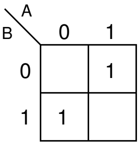

図 1: S

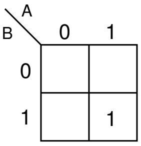

図 2: C

この 2 つはカルノー図を用いても簡単化できないのが図 1 と図 2 を見ても分かる。よって式は元のままである。

次に全加算器

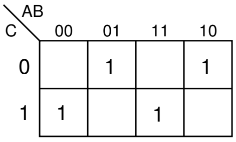

図 3: S

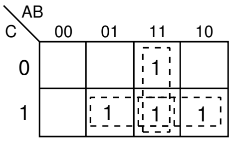

図 4: C_o

S のほうは図 3 よりどれも 2 つ以上離れているのでカルノー図でこれ以上簡単化はできない。次に C_o だが図 4 の点線で囲まれているところは一つの偏差になっており、計 3 組あることがわかる。よってこれは簡単化が可能で

$$C_o = BC + AC + AB$$

と表せる。

最後に 2 ビット加算器

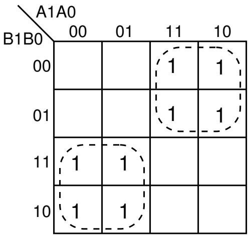

図 5: S_0

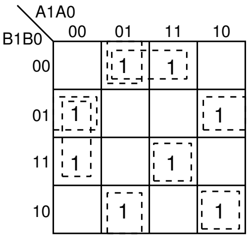

図 6: S_1

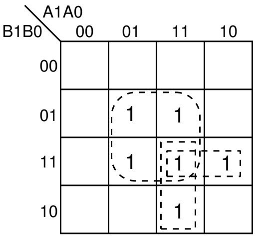

図 7: C

上記の図 5, 図 6, 図 7 より、S_0、S_1、C がそれぞれ簡単化できる事が分かる。
よって式は次のようになる。

$$
\begin{aligned}
S_0 &= A_0\,\overline{B_0}\,+\overline{A_0}\,B_0 \\
S_1 &= \overline{A_0}\,\overline{A_1}\,B_1\,+\overline{A_1}\,\overline{B_0}\,B_1\,+A_1\,\overline{B_0}\,\overline{B_1}\,+A_0\,\overline{A_1}\,B_0\,\overline{B_1}\,+A_0\,A_1\,B_0\,B_1 \\
C &= A_1\,B_1\,+A_0\,A_1\,B_0\,+A_0\,A_1\,B_1
\end{aligned}
$$

4.1.3

半加算器においては先程述べたようにカルノー図によって簡単化はできなかったので変わらない。

全加算器の方は C_o(キャリー) の論理式がカルノー図を扱うことで 1 つ 1 つの項が初めは 3 つの文字からなっていたのに対し、項は 2 つとなっており、式が簡単となっているのが確認できる。

2 ビット加算器、S_0 では初めは 8 項で 1 つの項には 4 変数であったのに対し、カルノー図での簡単化により、1 つの項は 2 変数で項数 2 だけの式となっている。S_1 も同様に項数が 8 個から 6 個へ、うち 4 つの項は 4 変数から 3 変数へと簡単化、C でも項数 6 つから 3 つへ、そのうち 1 つの項は 2 変数で、2 つの項は 3 変数とそれぞれの出力において簡単化されているのが確認できる。

4.1.4

実験で得られた式をそのまま図にするとそれぞれ図 8、図 9 のようになる。

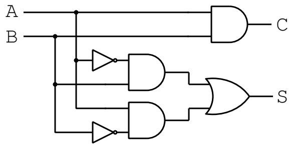

図 8: 半加算器

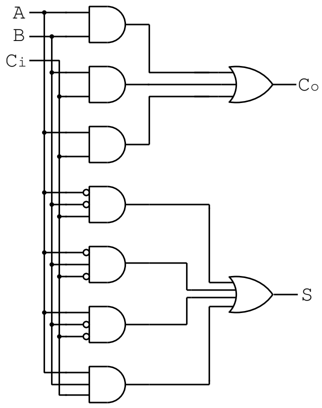

図 9: 全加算器

4.1.5

ブレッドボードで回路を実現する。半加算器は 2 入力 2 出力であるため、発光ダイオードを 2 つつけてその結果を確認した。入力をしたときには [1,1] 出力 C に相当するダイオードだけが光る。[1,0][0,1] は出力 S に相当するダイオードが発光。[0,0] のときはどちらも光らないという表 1 での真理値表と同じ結果となった。

4.1.6

実験指導書、P24 の図 3.5 を参考に全加算器を実現する。半加算器を 2 つ使用するので全実験での回路を 2 つ用意、後は OR 回路を 1 つと HA を 1 つの回路と見て計三つの回路を使い構成していく。この時の入力は Carry(桁上げ) の分も考えて 3 入力となっており、すべての入力の組み合わせを試して、表 2 の真理値表と同じ出力が確認できた。

4.2 加算器以外の実用的な組み合わせ回路を３つ挙げ、それらの真理値表と回路図を示せ。また、それらの動作や特徴について述べよ。

今回はマルチプレクサ (セレクタ)、デコーダ、多数決回路の 3 つの組み合わせ回路を挙げる。各回路図と真理値表は以下の表 4 表 5 表 6、図 10 図 11 図 12 をのせる。

- デマルチプレクサとは 1 つの入力から複数の出力へ振り分ける回路のことであり、今回の場合には SEL が'0' のときには A の値を X に出力、SEL が'1' のときには A の値を Y に出力する。
- エンコーダとはエンコード、つまりは符号化を行う組み合わせ回路である。
- 多数決回路とはその名の通り入力された値が多いものが出力される回路である。今回の 3 入力の回路で言えば'1' が 2 つ以上入力された時は'1' を出力する。と言ったものである。

表 4: デマルチプレクサ真理値表

入力		出力	
SEL	A	X	Y
0	0	0	0
0	1	1	0
1	0	0	0
1	1	0	1

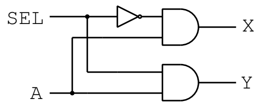

図 10: デマルチプレクサ

表 5: エンコーダ真理値表

入力				出力	
A_0	A_1	A_2	A_3	X_0	X_1
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

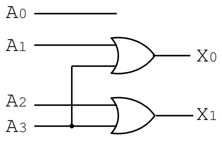

図 11: エンコーダ

表 6: 多数決回路真理値表

入力			出力
A	B	C	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

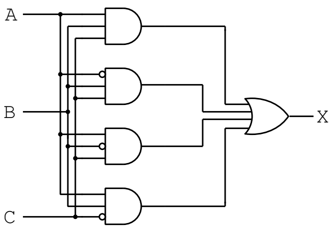

図 12: 多数決回路

4.3 クワイン-マクラスキー法とは、どのような方法か調査し報告せよ。

クワイン-マクラスキー法はカルノー図と同じく回路を論理圧縮するために用いられる方法である。論理圧縮の原理はカルノー図と同じであるが、カルノー図は通常、5 変数以上の場合には用いることは難しい。しかし、クワイン-マクラスキー方は 5 変数以上でも同じように扱えて便利である。以下にクワイン-マクラスキー法の簡単な手順をのせる。

1 まず、加法標準形にした関数を重みによって分ける

2 分けた重みの隣接する分類どうしでハミング距離が 1 となるものを探し、見つけた場合には補元則を適用して簡単化する。(1 次圧縮)

3 新しくできた分類どおしで (2) と同じステップを踏む (n 次圧縮)

4 ハミング距離が 1 となるものがなくなったら終了。圧縮できなかったそれぞれの項を取り出し、和でまとめる。

例として

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

をクワイン-マクラスキー法で圧縮する。

まず、表 7 のように分類分けする。

表 7: 一次圧縮

グループ	最小項
G_0	$\overline{A}\ \overline{B}\ \overline{C}\ \overline{D}\ *$
G_1	$A\ \overline{B}\ \overline{C}\ \overline{D}\ *$
G_2	$\overline{A}\ B\ \overline{C}\ D\ *$
	$A\ \overline{B}\ \overline{C}\ D\ *$
	$A\ \overline{B}\ C\ \overline{D}\ *$
G_3	$\overline{A}\ B\ C\ D\ *$
	$A\ B\ \overline{C}\ D\ *$
	$A\ B\ C\ \overline{D}\ *$
G_4	$A\ B\ C\ D\ *$

この表における'*' は圧縮可能なものを指しているこれらを一次圧縮したのが次の表 8 である。

表 8: 式の分類

グループ	最小項
G_0, G_1	$\overline{B}\ \overline{C}\ \overline{D}$
G_1, G_2	$A\ \overline{B}\ \overline{C}$
	$A\ \overline{B}\ \overline{D}$
G_2, G_3	$\overline{A}\ B\ D\ *$
	$B\ \overline{C}\ D\ *$
	$A\ \overline{C}\ D$
	$A\ C\ \overline{D}$
G_3, G_4	$B\ C\ D\ *$
	$A\ B\ D\ *$
	$A\ B\ C$

この表をさらに 2 次圧縮したものが次の表 9 となる。

表 9: 2 次圧縮

グループ	最小項
$(G_0, G_1), (G_1, G_2)$	ー
$(G_1, G_2), (G_2, G_3)$	ー
$(G_2, G_3), (G_3, G_4)$	BD

これらのうち圧縮できなくなったものが簡単化した項となる。よって例題は次のように論理圧縮できる。

$$f = \overline{B}\ \overline{C}\ \overline{D}\ + A\ \overline{B}\ \overline{C}\ + A\ \overline{B}\ \overline{D}\ + A\ \overline{C}\ D\ + A\ C\ \overline{D}\ + A\ B\ C\ + B\ D$$

4.4 同期式と非同期式の違いを調査し報告せよ。

同期式の回路というのはクロック信号の立ち上がり、あるいは立ち下がりに合わせて回路全体が動作するものである。一方、非同期式はクロック信号を用いない回路である。現在のデジタル回路では同期式の回路が主に使われている。その理由は非同期式に比べ同期式の回路の設計が容易で扱いやすいためであるが、クロックが早くなってくると配線遅延という問題があがる。クロックレスの非同期式にした場合、高速性を見いだせる。また、非同期式には消費電力の軽減や電磁波の放射強度が低いといった利点もある。

4.5 D フリップフロップおよび (同期式) カウンタとはどのような回路か調査し、報告せよ。

D フリップフロップはクロックが'0' から'1' へ変化するときには入力した値をそのままに出力し、それ以外の時には前回出力した値を保持して出力する。そのため、このフリップフロップは 1 ビットの記憶ができると考えられる。よって D フリップフロップを複数ビット並べてレジスタを作成できる。また、D フリップフロップは入力状態がクロックが変わって遅れて出力させることが可能なので遅延 (Delay) フリップフロップとも呼ばれる。

カウンタとは「数を数えるもの」という意味でその意味の通り、時とともに到来するパルス状の信号の個数をカウントする回路である。パルス (つまり入力) がくるたびに内容を 1 増加 (アップカウンタ)、あるいは 1 減少させる (ダウンカウンタ)。

4.6 本実験について考察せよ。

本実験では、真理値表からカルノー図を作成し、それを簡単化することと加算器についてを学んだ。加算器はもっとも基本的な組み合わせ回路であることが分かった。カルノー図で簡単化すると回路に必要なゲート数が減る。つまり、コストや回路自体の速度があがるので非常に有効性がある。

その他、次の実験のための予習の項目に当たる調査では、同期式と非同期式の違い、D フリップフロップおよびカウンタとはどのようなものかをある程度理解できたと思われる。

今回の実験はやはり、次の実験に繋げるための予行とも考えられた。実際、本実験の報告より、その他の次の予習の調査の方が難しかった。やはり、次の実験は今回より複雑になるだろうと考えられた。

5 その他

感想

今回、レポートの作成に非常に時間がかかりました。思った以上に資料集めに手こずりまして…。実際、友達との協力が無ければ遅刻は確定的だったと思われます。持つべきは友と改めて考えることができました。正直、毎回思うのですが Latex は中々に扱い辛くて苦労します。特に図や表が思った通りに貼れない所が。これさえどうにかなればもう少し早くできると思います。

今回も前回と同様に始めるのが遅かったおかげで徹夜するはめになりました…。次こそは月曜日を暖かい布団で過ごせるように早めに取りかかろうと思います。

参考文献

[1] 吉田たけお、尾知博：”VHDL で学ぶディジタル回路設計”，CQ 出版社.

[2] http://laputa.cs.shinshu-u.ac.jp/ yizawa/logic/,”LogicCircuits”

[3] http://www002.upp.so-net.ne.jp/latex/ , ”LaTeX コマンドシート一覧”

[4] http://www.arwil.net/ shin/tgif/index.ja.html,”Let’s draw circuit diagram with tgif”

[5] http://www.hobby-elec.org/menu.htm,” 趣味の電子回路工作”