情報工学実験 I
ディジタル回路
実験 2 汎用ロジック IC による組み合わせ回路の実現

145705F 宮里 悠太郎

2015/04/28

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.

1　実験概要

今回の実験では, 組み合わせ回路である全加算器を設計し, それを通して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学ぶ. まず真理値表から論理関数を得る手順を行う. そして求めた論理関数を分配則などの規則によって簡単化を行い, 回路を作成する際に使用するゲートの数を最小に抑える. 本来はカルノー図を用いて論理関数の簡単化を行う予定だったが, カルノー図についての知識を持ち合わせていなかったため, 今回は使用しなかった. なお今回はAND,OR,NAND,NOR,XOR のゲートが使用できるものとして考えて簡単化を行う.

簡単化された論理関数を元に回路図を描き, ブレッドボードを用いて実際に回路を作成し, 論理関数の簡単化が正常に行われているかを確認する.

2　実験結果

2.1　全加算器の真理値表から, 論理関数を求めよ. なお全加算器の出力は 2 ビットなので, 論理関数は 2 つ得られる.

A	B	C_{in}	S	C_{out}
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 1　全加算器の真理値表

全加算器の出力は S と C_{out} の 2 つ存在するので, まずは S から求める. 表 1 より,S(A,B,C_{in}) が 1 となる A,B,C_{in} の組み合わせは,(0,0,1),(0,1,0),(1,0,0),(1,1,1) の 4 通りである. 従って S(A,B,C_{in}) は次のように OR で表現される.

$$S(A,B,C_{in}) = \overline{A}\cdot\overline{B}\cdot C_{in} + \overline{A}\cdot B\cdot\overline{C_{in}} + \cdot A\cdot\overline{B}\cdot\overline{C_{in}} + A\cdot B\cdot C_{in}$$

続いて同じように C_{out} の論理関数を求める.C_{out} が 1 となる A,B,C_{in} の組み合わせは,(0,1,1),(1,0,1),(1,1,0),(1,1,1) の 4 通りである. 従って C_{out} は次のように OR で表現される.

$$C_{out}(A,B,C_{in}) = \overline{A}\cdot B\cdot C_{in} + A\cdot\overline{B}\cdot C_{in} + A\cdot B\cdot\overline{C_{in}} + A\cdot B\cdot C_{in}$$

2.2　2.1 で得られた論理関数に対して, 以下の規則を適用して論理関数を簡単化せよ.

(a) $A \cdot C + B \cdot C = (A + B) \cdot C$ (分配則の適用)
(b) $\overline{A \cdot B} = \overline{A} + \overline{B}$ (ド・モルガンの法則 (1) の適用)
(c) $\overline{A + B} = \overline{A} + \overline{B}$ (ド・モルガンの法則 (2) の適用)
(d) $A \cdot \overline{B} + \overline{A} \cdot B = A \oplus B$ (XOR ゲートの使用)
(e) $A \cdot B + \overline{A} \cdot \overline{B} = \overline{A \oplus B}$ (XOR ゲートと NOT ゲートの使用)
(f) $A \cdot A \cdot A \cdot ...A = A$
(g) $A + A + A + ...A = A$
(h) $A + \overline{A} = 1$

まずは S(A,B,C_{in}) を簡単化する.

$$
\begin{aligned}
S(A, B, C_{in}) &= \overline{A} \cdot \overline{B} \cdot C_{in} + \overline{A} \cdot B \cdot \overline{C_{in}} + A \cdot \overline{B} \cdot \overline{C_{in}} + A \cdot B \cdot C_{in} \quad \text{(a) より} \\
&= \overline{A} \cdot (\overline{B} \cdot C_{in} + B \cdot \overline{C_{in}}) + A \cdot (\overline{B} \cdot \overline{C_{in}} + B \cdot C_{in}) \quad \text{(d) より} \\
&= \overline{A} \cdot (B \oplus C_{in}) + A \cdot (\overline{B \oplus C_{in}}) \quad \text{(d) より} \\
&= A \oplus B \oplus C_{in}
\end{aligned}
$$

続いて C_{out} を簡単化する.

$$
\begin{aligned}
C_{out} &= \overline{A} \cdot B \cdot C_{in} + A \cdot \overline{B} \cdot C_{in} + A \cdot B \cdot \overline{C_{in}} + A \cdot B \cdot C_{in} \quad \text{(a) より} \\
&= C_{in} \cdot (\overline{A} \cdot B + A \cdot \overline{B}) + A \cdot B(\overline{C_{in}} + C_{in}) \quad \text{(d),(h) より} \\
&= C_{in} \cdot (A \oplus B) + A \cdot B
\end{aligned}
$$

2.3 2.1 および 2.2 で得られた論理関数を比較し,2.2 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.

簡単化前	$S(A,B,C_{in}) = \overline{A}\cdot\overline{B}\cdot C_{in} + \overline{A}\cdot B\cdot\overline{C_{in}} + A\cdot\overline{B}\cdot\overline{C_{in}} + A\cdot B\cdot C_{in}$
簡単化後	$S(A,B,C_{in}) = A \oplus B \oplus C_{in}$

表 2　S の比較

XOR2 つだけで表現できるように簡単化された.

簡単化前	$C_{out} = \overline{A}\cdot B\cdot C_{in} + A\cdot\overline{B}\cdot C_{in} + A\cdot B\cdot\overline{C_{in}} + A\cdot B\cdot C_{in}$
簡単化後	$C_{out} = C_{in}\cdot(A \oplus B) + A\cdot B$

表 3　C_{out} の比較

AND2 つと OR1 つ,XOR1 つで表現できるように簡単化された.

2.4 2.2 で得られた論理関数から, 全加算器の回路図を描け. なお論理関数が 2 つあるので, 回路図も 2 つとなるが, それらの回路図の入力をまとめて,1 つの回路図として描くこと.

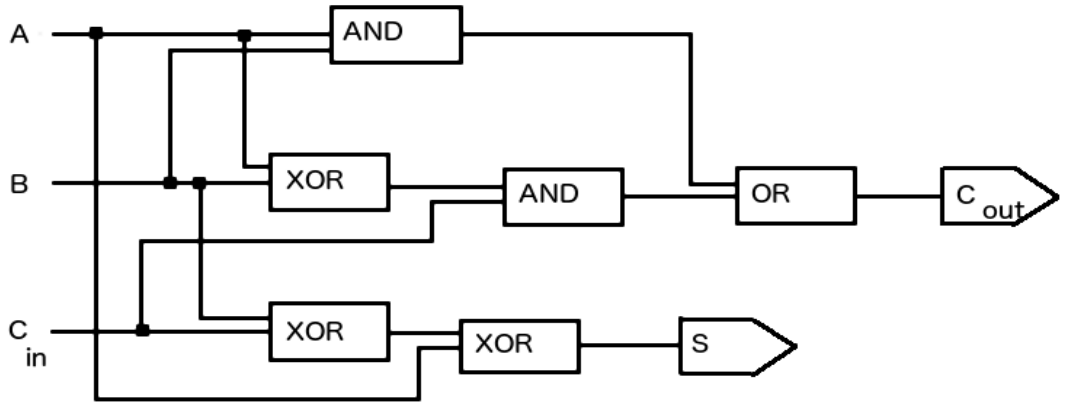

図 1　全加算器の回路図

2.5 2.4 で描いた回路図をもとにして, ブレッドボード上に全加算器を実現し, その動作を確認せよ. なお使用する IC が正常に動作することを, 事前に確認しておくこと.

表 1 の真理値表と同等の動作を行うことを確認した.

3　考察

3.1　2.2 についての考察

2.2 で簡単化された論理関数よりも使用する論理演算子を減らすことが可能かどうか考察する.
S については 3 つの入力に対して 2 つの論理演算子なのでこれ以上簡単化することは不可能だと考えられる.
C_{out} について 2.2 とは別パターンの簡単化を行ってみた.

$$\begin{aligned} C_{out} &= \overline{A}\cdot B\cdot C_{in} + A\cdot\overline{B}\cdot C_{in} + A\cdot B\cdot\overline{C_{in}} + A\cdot B\cdot C_{in} \quad \text{(a) より} \\ &= B\cdot C_{in}(\overline{A}+A) + A\cdot(\overline{B}\cdot C_{in} + B\cdot\overline{C_{in}}) \quad \text{(d),(h) より} \\ &= B\cdot C_{in} + A\cdot(B\oplus C_{in}) \end{aligned}$$

2.2 とは別の形に簡単化できたが論理演算子の数は減っていない.

$$\begin{aligned} C_{out} &= \overline{A}\cdot B\cdot C_{in} + A\cdot\overline{B}\cdot C_{in} + A\cdot B\cdot\overline{C_{in}} + A\cdot B\cdot C_{in} \quad \text{(a) より} \\ &= B\cdot(\overline{A}\cdot C_{in} + A\cdot\overline{C_{in}}) + A\cdot C_{in}\cdot(\overline{B}\cdot B) \quad \text{(d),(h) より} \\ &= B\cdot(A\oplus C_{in}) + A\cdot C_{in} \end{aligned}$$

こちらも別の形だが論理演算子は減らせていない.
以上の結果から 2.2 は最適な簡単化だったと言える.

4　感想

今回は GIMP で回路図を描きましたが、上手くかけず時間も多大にかかったので, 次からはエディタを探してそれで描きたいと思います.
いいアプリケーションがあるならば紹介をお願いします.

5　参考文献・引用文献

[1] 平成 27 年度 情報工学実験 I 実験指導書 (ディジタル回路)
[2] LaTeX コマンド集
http://www.latex-cmd.com