問010044解説

◆解答

設問１　ａ　イ　　ｂ　ア

設問２　オ

設問３　エ

設問４　ｃ　オ　　ｄ　ウ

◆解説

論理演算に関する問題を解く方法に次の方法がある。

①　論理公式を用いる方法

②　真理値表を用いる方法

③　ベン図を用いる方法

④　ベッチ・カルノー図を用いる方法

ここでは、真理値を用いる方法で考える。

真理値表

　真理値表は、論理変数の取り得るすべての条件とコンピュータの基本演算である否定、論理和、論理積、排他的論理和などの論理演算の結果を表形式で表したものである。２つの論理変数Ａ、Ｂと論理演算の結果の間には次の関係が成立する。表中１は真、０は偽を表す。

Ａ	Ｂ	$\overline{A}$	Ａ∨Ｂ	Ａ∧Ｂ	Ａ∀Ｂ
0	0	1	0	0	0
0	1	1	1	0	1
1	0	0	1	0	1
1	1	0	1	1	0

　ＮＡＮＤは、ＡＮＤの結果の否定であり、ＮＯＲはＯＲの結果の否定であるから、次の真理値表になる。

Ａ	Ｂ	$\overline{A}$	$\overline{B}$	$\overline{A \lor B}$	$\overline{A \land B}$	$\overline{A} \lor \overline{B}$	$\overline{A} \land \overline{B}$
0	0	1	1	1	1	1	1
0	1	1	0	0	1	1	0
1	0	0	1	0	1	1	0
1	1	0	0	0	0	0	0

　上の表から、ビットパターンが一致するのは、ＮＡＮＤの場合は入力の否定の論理和$\overline{A} \lor \overline{B}$となり、ＮＯＲの場合はは入力の否定の論理積$\overline{A} \land \overline{B}$となる。

ケース	C_{in}	X	Y	C	Z
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

図２の全加算器の論理回路について考える。

ケース１、２、３については、C_{in}の入力は０であるから、後段の半加算器Ｘの入力に関係なく後段の半加算器Ｃの出力は０となり、前段の加算器の出力Ｃも０であるから、求める論理演算はＡＮＤ、ＯＲのいずれも成り立つ。

ケース７、８の２ケースについては、C_{in}の入力は１であり、後段の半加算器Ｘの入力も１であるから後段の半加算器Ｃの出力は１となり、前段の加算器の出力Ｃも１であるから、求める論理演算はＡＮＤ、ＯＲのいずれも成り立つ。

ケース４の場合、前段の半加算器の出力Ｃは１、後段の半加算器の入力Ｘは０、入力C_{in}は０であるから、後段の半加算器の出力Ｃは０となる。従って、求める論理演算は１と０の演算結果が１となる場合であるから、求める論理演算はＯＲ、ＸＯＲのいずれも成り立つ。

ケース５の場合、前段の半加算器の出力Ｃは０、後段の半加算器の入力Ｘは１、入力C_{in}は１であるから、後段の半加算器の出力Ｃは１となる。従って、求める論理演算は０と１の演算結果が０となる場合であるから、求める論理演算はＯＲ、ＸＯＲのいずれも成り立つ。

ケース６の場合、前段の半加算器の出力Ｃは０、後段の半加算器の入力Ｘは１、入力C_{in}は１であるから、後段の半加算器の出力Ｃは１となる。従って、求める論理演算は０と１の演算結果が１となる場合であるから、求める論理演算はＯＲ、ＸＯＲのいずれも成り立つ。

ケース１～８について、共通して成り立つのは論理和である。

加算結果の“あふれ”

負数を２の補数表現で表す場合の異符号の加算は、和の絶対値はどちらの演算数の絶対値よりも大にはならないため「あふれ」は生じない。

負数を２の補数表現で表す場合の同符号の加算は、演算結果の符号が被加数、加数の符号と異なると「あふれ」が生じて正しい答えを求めることができない。加算結果の値がそのビット数で表現できる範囲の値を超えるため「あふれ」が発生する。

具　体　例

２進数0101と0010の和は0111となる。２進数0101は10進数の５であり、２進数0010は10進数の２であるから５＋２＝７となり、２進数で表すと0111となる。

２進数0101と0100の和は1001となる。２進数0101は10進数の５であり、２進数0100は10進数の４であるから５＋４＝９となり、２進数で表すと1001となる。

しかし、負数を２の補数で表す２進数の加算の場合、1001は10進数の－７となり、10進数の加算の結果と一致しなくなる。これは、同符号の加算で、加算結果の符号が被加数、加数の符号と異なる結果となっており、あふれが発生したためである。

$$\begin{array}{r} 0101 \\ +0010 \\ \hline 0111 \end{array} \quad \cdots\cdots \quad \begin{array}{r} 5 \\ +2 \\ \hline 7 \end{array}$$

↑ 符号が変わらない

$$\begin{array}{r} 0101 \\ +0100 \\ \hline 1001 \end{array} \quad \cdots\cdots \quad \begin{array}{r} 5 \\ +4 \\ \hline 9 \end{array}$$

↑ 符号が変わる

設問１

A	B	$\overline{A}$	$\overline{B}$	$\overline{A \vee B}$	$\overline{A \wedge B}$	$\overline{A} \vee \overline{B}$	$\overline{A} \wedge \overline{B}$
0	0	1	1	1	1	1	1
0	1	1	0	0	1	1	0
1	0	0	1	0	1	1	0
1	1	0	0	0	0	0	0

上の真理値表から、ビットパターンが一致するのは、ＮＡＮＤの場合は入力の否定の論理和$\overline{A} \vee \overline{B}$となり、ＮＯＲの場合はは入力の否定の論理積$\overline{A} \wedge \overline{B}$となる。

ａの求める答えはイ、ｂの求める答えはアとなる。

設問２

表２の半加算器の真理値表から、Ｘ、Ｙのいずれか一方が１の場合、加算した結果のＺは１となり、Ｘ、Ｙが同じビットの場合は加算した結果のＺは０となるため、Ｚの値はＸ、Ｙの排他的論理和になる。答えはＸＯＲとなり、求める答えはオとなる。

設問３

ケース１～８について、論理演算の前段半加算器の出力Ｃ、後段半加算器の出力Ｃの値を求めて、その値での演算結果が１になる論理演算を求めればよい。

解説の項での説明によると、ケース１～３、７、８はＡＮＤ、ＯＲのいずれも成り立ち、ケース４、５、６はＯＲ、ＸＯＲのいずれも成り立つ。従って、ケース１～８について共通して成り立つのは論理和ＯＲである。求める答えはエとなる。

設問４

負数を２の補数表現で表す場合、同符号の加算は、演算結果の符号が被加数、加数の符号と異なると「あふれ」が生じる。ケース４の場合、Ｘ＝１、Ｙ＝１、Ｚ＝０、ケース５の場合、Ｘ＝０、Ｙ＝０、Ｚ＝１とにりあふれが発生する。最上位ビット$Ａ_4$、$Ｂ_4$、$Ｃ_3$の加算結果の$Ｃ_4$に関して、$Ｃ_3$、$Ｃ_4$が異符号になるとあふれが発生する。$Ｃ_3$、$Ｃ_4$が異符号になるのは排他的論理和の場合である。従って、図４の桁あふれ検出の論理回路でＶＦが１となるのは、異なるビット$Ｃ_3$、$Ｃ_4$が排他的論理和になる場合である。ｃの求める答えはオとなる。

図５のゼロ検出の論理回路において、論理積の演算結果ＺＦが１になるのは、入力ｄが１の場合である。ｓ１、ｓ２、ｓ３、ｓ４がすべて０であり、ｓ１とｓ２、ｓ３とｓ４の演算結果がそれぞれ１となるのは、ｄがＮＯＲの場合である。ｄの求める答えはウとなる。