問1　論理演算と加算器（ハードウェア）　（H25 秋·FE 午後問 1）

【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】

論理演算と加算器に関する問題である。このテーマは過去に数回出題されており，出題頻度が高い。今後も出題されると推測する。また，論理演算の内容は午前試験では必須事項であるから，選択した場合は，確実に解けるよう，学習しておかなければならない。

加算回路には，半加算器，全加算器がある。その回路構成まで熟知している人は少ないと思うが，基本回路（論理演算）の組合せなので，入力情報，出力情報から推測できる。

論理演算では，その真理値を確認できる真理値表を理解しておくことが大切である。主要な論理演算の真理値表が表1に示されているので，それを確認しながら正確に解いていく。

ケアレスミスに注意すれば，設問の論点も過去に出題された内容と同じであり，設問3及び設問4の空欄cが理解できれば，得点が獲得できる内容である。

［設問1］

表1から確認できるように，NAND 演算は AND 演算の否定であり，NOR 演算は OR 演算の否定である。選択肢に用意された回路図に，適当な値（0又は1）を代入して確認しても解けるが，解法上重要な真理値表を用いて検証してみる。

表A　NAND 及び NOR 回路の検証を行う真理値表

X	Y	$\overline{X}$	$\overline{Y}$	ア：$\overline{X}$ AND $\overline{Y}$	イ：$\overline{X}$ OR $\overline{Y}$	ウ：$\overline{X}$ XOR $\overline{Y}$	エ：$\overline{X \text{ XOR } Y}$
0	0	1	1	1	1	0	1
0	1	1	0	0	1	1	0
1	0	0	1	0	1	1	0
1	1	0	0	0	0	0	1

注記　NOT X と $\overline{X}$ は同義

表Aの真理値と表1の真理値を比較，確認すると，表1の NAND 演算と同じ真理値をもつ論理回路は（イ），NOR 演算と同じ真理値をもつ論理回路は（ア）である。

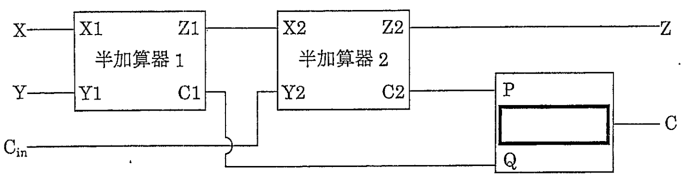
［設問2］

求めるのは加算結果の Z である。表2に Z の真理値が示されているので，その真理値と表1で示されている真理値を比較，確認すればよい。したがって，正解は「XOR」の（オ）である。

［設問3］

複数桁の加算を行うのが全加算器であるが，1桁の加算と違い，下位桁からの桁上がりを考慮しないと正確な値が求められない。したがって，入力値は加算対象の2値(X, Y)と下位桁からの桁上がり(C_{in})の三つである。その加算結果は，和の結果(Z)と上位桁への桁上がり(C)となる。入力値は三つ，それぞれが0，1の値をとるので，考えられる組合せは $2^3=8$ 通りある。これらの結果をまとめたのが表3の内容である。

求めるのは，C を出力するための演算内容である。図2と表3から，真理値表で確認すると次のようになる。なお，説明のために，図2の論理回路に入力番号を付けたものを図Aとして示す。



図A　入力番号を付けた全加算器の論理回路

C_{in} の入力値は表3から，0，1のいずれかであることが分かり，この値が Y2 の入力値となる。また，X2 の入力値は Z1 であるので，これは設問2から，X1 XOR Y1(X XOR Y)となる。また，C を出力するための入力値をそれぞれ，P，Q とすると，P=C2=X2 AND Y2，Q=C1=X1 AND Y1(X AND Y)となる。これらを理解した上で，真理値表を作成すると表B-1，B-2のようになる。

表B-1　C_{in}(Y2)=0 の真理値表

C_{in}(Y2)	X	Y	X2=X XOR Y	P=X2 AND Y2	Q=X AND Y	C
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	1	0	1	0	0	0
0	1	1	0	0	1	1

①：P と Q の演算で C の真理値となるのは OR 演算だけである。これで解答は得られるが，参考までに C_{in}(Y2)=1 の場合も表B-2で示す。

表B-2　C_{in}(Y2)=1 の真理値表

C_{in}(Y2)	X	Y	X2=X XOR Y	P=X2 AND Y2	Q=X AND Y	C
1	0	0	0	0	0	0
1	0	1	1	1	0	1
1	1	0	1	1	0	1
1	1	1	0	0	1	1

②：P と Q の演算で C の真理値となるのは OR 演算だけである。

①か②のどちらかを確認すればよい。したがって，正解は「OR」の（エ）である。

［設問4］

2の補数表現による4ビットの符号付き2進整数を加算する加算器が図3に示されている。内容は4桁の加算内容であるが，設問文にある「最上位ビットの加算において，A_4，B_4，C_3 の値が表3の全加算器の真理値表のそれぞれ X，Y，C_{in} の値の β 部分の組合せになるとき，桁あふれが生じる」の記述の意味を理解する必要がある。

注意するのは，「2の補数表現による4ビットの符号付き2進整数を加算」という記述である。2の補数表現で表現された符号体系を固定小数点数といい，最上位ビット0のときが正（+），1のときが負（−）となる。そして，桁あふれは，正の数同士か負の数同士の加算のときに発生する。

4ビットの固定小数点数で表現できる数値の範囲は −8〜+7 であることと，2進数の加算結果を理解している前提で，4ビットの固定小数点における桁あふれの例を次に示す。

例1　3と6の加算（正の数同士の加算）

$\boxed{0}$ 0 1 1　　←　$A_4=0$
) $\boxed{0}$ 1 1 0　　←　$B_4=0$
$\boxed{1}$ 0 0 1　　←　$C_3=1$

結果は9となるはずであるが，桁あふれが発生し，正しい結果とならない（符号ビットが1）。

C_4 の値は0と考える。

例2　−3と−6の加算（負の数同士の加算）

$\boxed{1}$ 1 0 1　　←　$A_4=1$
) $\boxed{1}$ 0 1 0　　←　$B_4=1$
1 $\boxed{0}$ 1 1 1　　←　$C_3=0$

結果は−9となるはずであるが，桁あふれが発生し，正しい結果とならない（符号ビットが0）。

加算処理では，5ビット目(C_4)は無視（切捨て）するが，C_4 の値は1である。

A_4，B_4，C_3 は表3でいうと，β の下段が（正の数同士の加算）例1，上段が（負の数同士の加算）例2に対応する。なお，それ以外は桁あふれとならないので，各自適当な値で検証してほしい。

・空欄c：C_3 と C_4 の値の検証なので，次の4通りを考えればよい。

(1) 正の数同士の加算…最上位ビットは正の符号を表すので，常に0 ($C_4=0$) でなければならない。また，C_3 が取りうる値は，0か1のいずれかである。

①　桁あふれなし　C_3：0　C_4：0
②　桁あふれ発生　C_3：1　C_4：0　例1

(2) 負の数同士の加算…最上位ビットは負の符号を表すので，常に1 ($C_4=1$) でなければならない。また，C_3 が取りうる値は，0か1のいずれかである。

③　桁あふれなし　C_3：1　C_4：1
④　桁あふれ発生　C_3：0　C_4：1　例2

桁あふれが生じるのは②，④の場合であり，出力結果の VF の値は1，それ以外の①，③では0となることから，正解は「XOR」の（オ）である。他の選択肢では，正しい VF の値が求まらないので，誤りである。

・空欄d：S_1〜S_4 が全て0となる場合であるから，S_1〜S_4 の入力値を0として図5で考えればよい。このとき，ZF の値が1となるので，AND 回路に入力される2値はどちらも1でなければならない。これは，空欄dの回路の出力値が1であることを意味している。間違えやすい選択肢もあるので，表Cで示す。

表C　空欄dの検証を行う真理値表

	ZF	S_1	S_2	S_3	S_4	ア：AND	イ：NAND	ウ：NOR	エ：OR	オ：XOR
①	1	0	0	0	0	0（×）	1（○）	1（○）	0（×）	0（×）
②	0	1	0	1	0	−	1（×）	0（○）	−	−

注記：○は正しい出力値，×は誤りの出力値，−は検証の必要なし

S_1〜S_4 が全て0の①では，候補として，（イ）の「NAND」，（ウ）の「NOR」が残る。ほかは ZF の出力値が1とならないので誤りである。また，②の場合，S_1 と S_2，S_3 と S_4 の NAND 演算で1が出力されるため，ZF の値が0とならず誤りである。したがって，正解は「NOR」の（ウ）である。