3入力パラメトロンによる4変数論理関数の実現法*

坪　井　定　一**

1. まえがき

3入力パラメトロンによって，組み合わせ回路を合理的に設計する方法を考える．一般に n 変数の論理関数は**第1表**に示されている個数だけあることがわかっている[1]．そこで4変数の論理関数を扱うとすれば，真に異なる*** 224 種類の関数それぞれに対応して 224 種類の回路図を用意しておけば原理的には事足りるはずである．

第1表

n	n 変数論理関数の数	真の n 変数論理関数の数	同族* でない n 変数論理関数の数	同族でない真の n 変数論理関数の数
0	2	2	1	1
1	4	2	2	1
2	16	10	4	2
3	256	218	14	10
4	65,536	64,594	238	224

* 同族とは変数の置換，変数の否定，関数の否定によってうつり得ることをいう．

3変数の論理関数の場合には，真に異なる論理関数の個数が僅かに 10 種類であるということと，関数の立方体表示という直観的な表現方法があるなどの理由から，10 種類の回路図を用意しておくというのが有効な手段として一般に用いられている．しかしながら4変数の論理関数の場合には，たとえこのような目的で 224 種類の回路図を用意しておいたとしても，実際に与えられた関数がこの 224 種類のうちのどの関数と同族であるかを探すのはやや面倒な仕事である[2]．

そこで，この場合にはあらゆる場合を想定して回路図を用意しておくというよりも，こう考えていけば必ず回路図を導き得るという公式を作る方が実際的であろうと考え，以下このような観点に立って4変数論理関数の設計問題を考えていくことにする．

* A Logical Design of Circuit Representing 4-Variable Boolean Functions by means of 3 Fan-ins Parametrons, by Teiichi Tsuboi (Yaskawa Electric mfg. Co., Ltd., Laboratory, Yawata)

** 安川電機製作所研究所

*** 変数の置換，変数の否定，関数の否定によってうつり得ないことを真に異なるという

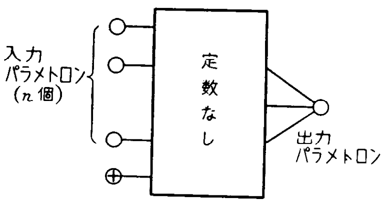
2. 自己双対関数

自己双対関数については次の定理が知られている．

【定理】

ある関数が定数なしで組み立てられるための必要かつ十分なる条件は，その関数が自己双対なることである．

この定理の裏をいえば，自己双対でない関数を実現するためには定数が必要であるということになる．ところで，定数を含む組み合わせ回路は必ず**第1図**に示すように，入力に1個の定数パラメトロンを追加すれば，あとは定数なしで組み立てることができる．したがって，n 変数の自己双対でない関数を実現するためには，ある特定の $(n+1)$ 変数の自己双対関数を実現する回路を作り，つぎにその自己双対回路のある特定の入力パラメトロンを定数に固定すればよいことになる．なお，この場合 $(n+1)$ 変数の自己双対回路が可能な最少段数で実現されていれば，この手法によって実現した n 変数の自己双対でない回路も可能な最少段数の条件を満足することになる．



第1図

いま n 個の入力変数を $(x_1, x_2, \cdots\cdots, x_n)$ とし，全部で 2^n 個ある入力ベクトルに，それぞれ X_1, X_2, $\cdots\cdots$, X_{2^n} と名前をつける．そうして

$$X_i = x_{i1}x_{i2}\cdots\cdots x_{in}\ (x_{ij}\ \text{は}\ x_j\ \text{または}\ x_j')$$

としたとき，X_i の補ベクトル X_i' を次のように定義する．

$$X_i' = x_{i1}'x_{i2}'\cdots\cdots x_{in}'$$

そこで n 変数の自己双対でない論理関数 y

$$y=X_1 \vee X_2 \vee \cdots\cdots \vee X_k$$

に対して，変数 x_0 を追加して次式に示されているような $(n+1)$ 変数の論理関数 z

$$z=x_0X_1 \vee x_0X_2 \vee \cdots\cdots \vee x_0X_k$$
$$\vee x_0'X_{k+1}' \vee x_0'X_{k+2}' \vee \cdots\cdots \vee x_0'X_{2^n}'$$

を考えてみるとこの関数は明らかに自己双対である．

〔例〕 与えられた自己双対でない関数を

$$y=X_1=x_1x_2x_3$$

とすると（**第２表**），z は

第２表

入力 $x_1\ x_2\ x_3$	出力 y	入力 $x_1\ x_2\ x_3$	出力 y
0 0 0	0	1 0 0	0
0 0 1	0	1 0 1	0
0 1 0	0	1 1 0	0
0 1 1	0	1 1 1	1

$$z=x_0x_1x_2x_3 \vee x_0'x_1'x_2'x_3' \vee x_0'x_1'x_2'x_3$$
$$\vee x_0'x_1'x_2x_3' \vee x_0'x_1x_2'x_3' \vee x_0'x_1x_2x_3'$$
$$\vee x_0'x_1x_2'x_3 \vee x_0'x_1'x_2x_3$$

となって（**第３表**）自己双対である．

第３表

入力 $x_0\ x_1\ x_2\ x_3$	出力 z	入力 $x_0\ x_1\ x_2\ x_3$	出力 z
0 0 0 0	0	1 0 0 0	0
0 0 0 1	1	1 0 0 1	0
0 0 1 0	1	1 0 1 0	0
0 0 1 1	1	1 0 1 1	0
0 1 0 0	1	1 1 0 0	0
0 1 0 1	1	1 1 0 1	0
0 1 1 0	1	1 1 1 0	0
0 1 1 1	1	1 1 1 1	1

このようにして，n 変数の任意の自己双対でない関数が与えられた場合にそれに対応する $(n+1)$ 変数の自己双対関数は容易に導き得るので，任意の組み合わせ回路の設計問題は自己双対回路の設計問題と等価であることがわかる．

3. 自己双対関数の T 数

自己双対関数の個数は調べられていて，$n \leqq 5$ の場合について示すと**第４表**のようになっている[3]．したがって４変数の論理関数を扱うとすれば，４変数および５変数の自己双対関数計 80 種類を調べればよいことになる．

第４表

n	n 変数自己双対関数の個数	同族でない n 変数自己双対関数の個数	同族でない真の n 変数自己双対関数の個数
1	2	1	1
2	4	1	0
3	16	3	2
4	256	7	4
5	65,536	83	76

これらの関数を実現する回路を設計する準備として，５変数以下の自己双対関数に名前を付ける必要がある．そこで，つぎのようにして T 数というものを定義する．

まず出力１を与えるすべての入力ベクトルを選び出して，各変数に現われる１の個数を数える．たとえば３変数 (x_1, x_2, x_3) の多数決関数 $m(x_1, x_2, x_3')$

$$m=x_1x_2x_3' \vee x_1x_2x_3 \vee x_1x_2'x_3' \vee x_1'x_2x_3'$$

の場合には，**第５表**に示す四つの入力ベクトルを選び出すことになり，１の個数は x_1, x_2 については３個，x_3 については１個となる．一般にこのようにして数えた１の個数がある変数に k 個あったとした場合に，q 個ある各変数について数えた結果を

第５表

$x_1\ x_2\ x_3$
1 1 0
1 1 1
1 0 0
0 1 0

$$\max(k,\ 2^{q-1}-k)$$

なる数で表現して，この q 個の数を大きい順に並べたものをその自己双対関数の T 数と定義する．いまの例 $m(x_1, x_2, x_3')$ では $q=3$ すなわち $2^{q-1}=4$ となるので，x_3 についての個数１個が

$$\max(1,\ 4-1)=3$$

３個に変換されて，T 数は (3, 3, 3) となる．

なお５変数のいくつかの自己双対関数に対しては，回路の設計上，その関数の parity が必要になることがある．関数の parity とは出力１を与える 16 個の入力ベクトルを１の個数が奇数個のものと偶数個のものとに分けたとき，その少ない方の数をいう．

〔例〕 ５変数の自己双対関数で，出力１を与える 16 個の入力ベクトルが第 13 表に示すような関数の場合には，T 数は (13, 11, 9, 9, 9) となり parity は７である．このことを T 数 (13, 11, 9, 9, 9; 7) と表現することにする．

３変数，４変数の自己双対関数の T 数を**第６表**に示す．ここに見られるように，それぞれの T 数の種類は３通り，７通りであって，第４表に示した自己双対関数の種類の数と等しくなっている．すなわち３変数，

4 変数の場合には，T 数によってすべての自己双対関数の種類を区別し得ることを示している．真に 5 変数の自己双対関数の T 数は 47 通り（第 10 表）しかなく，第 4 表に示した 76 種類よりも少ない．このことは，5 変数の自己双対関数の場合には T 数だけでは，たがいに区別することのできない関数族が存在することを示している．

第 6 表

n	真に n 変数	n 変数以下	
		実は 3 変数	実は 1 変数
3	(3, 3, 3) (2, 2, 2)	——	(4, 2, 2)
4	(7, 5, 5, 5) (6, 6, 4, 4) (6, 4, 4, 4) (5, 5, 5, 5)	(6, 6, 6, 4) (4, 4, 4, 4)	(8, 4, 4, 4)

4. 4 変数論理関数の作り方

4 変数論理関数の回路の作り方は，その関数の性質によって 5 通りに分かれる．そのそれぞれに属する関数を第 1 グループ，……，第 5 グループの関数として，以下順を追って説明する．**第 7 表**はそれぞれのグループに属する関数の代表例を示したものである．

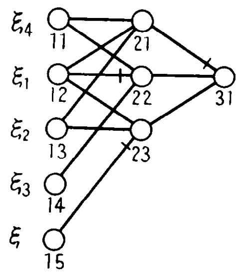
第 7 表

入力				出力				
x_1	x_2	x_3	x_4	y_1	y_2	y_3	y_4	y_5
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	1	0	0
0	0	1	1	0	1	0	1	1
0	1	0	0	0	0	1	1	0
0	1	0	1	0	1	0	1	1
0	1	1	0	0	0	0	1	1
0	1	1	1	1	1	1	1	0
1	0	0	0	0	0	1	1	0
1	0	0	1	1	1	0	1	1
1	0	1	0	1	0	0	1	1
1	0	1	1	1	1	1	1	0
1	1	0	0	1	0	0	1	1
1	1	0	1	1	0	1	0	0
1	1	1	0	1	0	1	0	0
1	1	1	1	1	0	0	0	0

4.1 第 1 グループ——自己双対関数

まず最初に，いま実現しようとしている関数が第 7 表 y_1 のように自己双対である場合について述べよう．この場合にはまずその自己双対関数の T 数を調べる．いまの例では入力変数 (x_1, x_2, x_3, x_4) に対して 1 の個数はそれぞれ (7, 5, 5, 5) となっているのでその T 数も (7, 5, 5, 5) である．

一方 4 変数の自己双対関数の作り方は**第 2 図**および**第 8 表**，**第 9 表**にまとめられている．たとえば T 数 (7, 5, 5, 5) の場合には第 8 表による $\xi=\xi_1$ にしたがって，第 2 図における $12(\xi_1)$ および $15(\xi)$ のパラメトロンを一つの同一のパラメトロンに置き換えればよく，その結果**第 3 図**に示す回路が得られる．この回路の出力を 1 にする入力ベクトルは，第 9 表に示されている 16 個の入力ベクトルのうちで $\xi=\xi_1$ の条件を



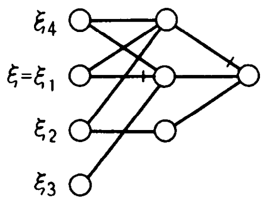
第 2 図

第 8 表

T 数	
(7, 5, 5, 5)	$\xi=\xi_1$
(6, 6, 4, 4)	$\xi=\xi_2$
(6, 4, 4, 4)	$\xi=\xi_3$
(5, 5, 5, 5)	$\xi=\xi_4$

第 9 表

$\xi\ \xi_1\ \xi_2\ \xi_3\ \xi_4$	$\xi\ \xi_1\ \xi_2\ \xi_3\ \xi_4$	$\xi\ \xi_1\ \xi_2\ \xi_3\ \xi_4$
0 0 0 0 1	0 0 1 1 1	1 0 0 1 0
0 0 0 1 0	0 1 0 0 0	1 0 0 1 1
0 0 0 1 1	0 1 0 1 0	1 0 1 1 0
0 0 1 0 0	0 1 0 1 1	1 1 1 1 1
0 0 1 0 1	0 1 1 1 1	
0 0 1 1 0	1 0 0 0 1	



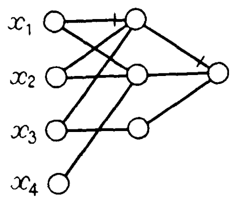
第 3 図

満足しているもの(計8個)として知ることができる．したがって，この回路の実現している関数は，入力変数 ($\xi=\xi_1, \xi_2, \xi_3, \xi_4$) に対して1の個数が (1, 5, 5, 5) となるような T 数 (7, 5, 5, 5) の自己双対関数である．

そこでこの関数を y_1 に一致させるためには

$$\begin{cases}\xi=\xi_1=x_1' \\ \xi_2=x_2 \\ \xi_3=x_3 \\ \xi_4=x_4\end{cases} \quad \text{または} \quad \begin{cases}\xi=\xi_1=x_1' \\ \xi_2=x_3 \\ \xi_3=x_4 \\ \xi_4=x_2\end{cases}$$

なる対応をつければよく，その結果 y_1 を実現する回路として**第4図**に示す回路が得られる．



第4図

以上を要約すれば，作ろうとする関数が自己双対の場合には

(1)　まずその関数の T 数を調べ

(2)　それと同じ T 数を示すある特定の回路を第2図，第8表にしたがって作ってから

(3)　入力変数 $\{\xi\}$ の間の置換およびその否定を適当にほどこして

いま作ろうとしている回路を実現するのである．

4.2　第2グループ——2段で実現可能な関数

実現しようとしている関数が自己双対でない場合には，次節**4.3**に述べる例外を除いて，2節に述べたようにして一まず5変数の自己双対関数に変換して考える．**第10表**は5変数の自己双対関数を実現する回路の作り方を，その T 数によって分類したものである．この表によれば T 数 (15, 9, 9, 9, 9) を示す回路は，その関数の parity を調べる必要はなく，本節**4.2** に述べる手法に従えば作ることができるということを示している．また T 数 (11, 11, 11, 9, 9 ; 7) なる関数については，入力変数間の置換およびその否定を適当に行なって1の個数が (11, 11, 11, 9, 9) になるようにしたとき，11100 なる入力ベクトルが含

第 10 表

T 数	4.2	4.4	4.5	T 数	4.2	4.4	4.5
15　9　9　9　9; 7	○			11　11　11　9　9; 7	(11100) ○	(00011) ○	
14　10　10　8　8; 6	○			11　11　9　9　9; 3		○	
14　10　8　8　8; 8	○			11　11　9　9　9; 5		○	
14　8　8　8　8; 6		○		11　11　9　9　9; 7		○	
13　11　11　9　9; 7	○			11　9　9　9　9; 3			○
13　11　9　9　9; 5		○		11　9　9　9　9; 5		○	
13　11　9　9　9; 7	○			11　9　9　9　9; 7		○	
13　9　9　9　9; 5		○		10　10　10　10　10; 6		○	
13　9　9　9　9; 7		○		10　10　10　10　8; 4		○	
12　12　10　10　10; 6	○			10　10　10　10　8; 8		○	
12　12　10　10　8; 8	○			10　10　10　8　8; 2		○	
12　12　10　8　8; 6		○		10　10　10　8　8; 6	○	○	○[1)]
12　12　8　8　8; 4		○		10　10　8　8　8; 4		○	○[2)]
12　10　10　10　10; 4		○		10　10　8　8　8; 8		○	
12　10　10　10　10; 8	○			10　8　8　8　8; 2			○
12　10　10　10　8; 6		○		10　8　8　8　8; 6		○	
12　10　10　8　8; 4		○		9　9　9　9　9; 1			○
12　10　10　8　8; 8		○		9　9　9　9　9; 3		○	
12　10　8　8　8; 6		○		9　9　9　9　9; 5		(11111) ○	(00000) ○
12　8　8　8　8; 4		○		9　9　9　9　9; 7		○	○[3)]
12　8　8　8　8; 8	○			8　8　8　8　8; 0			
11　11　11　11　11; 5		○		8　8　8　8　8; 4		○	
11　11　11　11　9; 7	○			8　8　8　8　8; 8			○
11　11　11　9　9; 5		○					

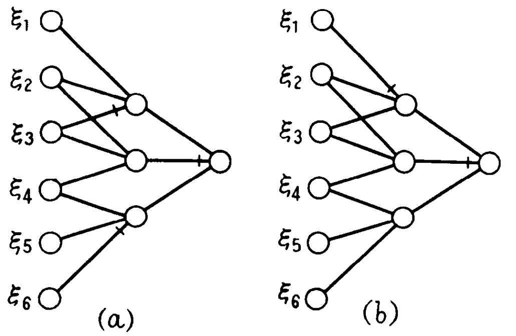
1)　422 か 331 なら 4.2，0 0 0 * * があって 332 か 322 ならば 4.5，それ以外は 4.4.

2)　0 0 * * * の二つのベクトルの距離が3なら 4.4，距離が2なら 4.5.

3)　$\left.\begin{matrix}1\ 0\ 0\ 0\ 0\\0\ 1\ 0\ 0\ 0\\0\ 0\ 1\ 1\ 1\end{matrix}\right\}$ なる関係の三つのベクトルがあれば 4.5，それ以外なら 4.4.

まれていれば **4.2**, 00011 なる入力ベクトルが含まれていれば**4.4**に述べる手法に従えばよい．さらにT数(10, 10, 10, 8, 8；6) なる関数については，T数が8の二つの入力変数が共に1であるような入力ベクトル（四つある）を選び出したときに，T 数が 10 である残りの三つの 入力変数に 含まれている1の 個数が (4, 2, 2) あるいは (3, 3, 1) なら**4.2**の手法に従えばよい．

さて全部で76種類ある 真に5変数の 自己双対関数のうちで，２段で実限することのできるものは12種類ある．そうしてこれらの回路の作り方は，４変数の場合に倣ってまとめると，**第5図**，**第 11，12 表**のようになる．



第5図

第11表

T	数				(a)	(b)
15	9	9	9	9	$\xi_3=\xi_6$	
14	10	10	8	8		$\xi_5=\xi_6$
14	10	8	8	8	$\xi_1=\xi_6$	
13	11	11	9	9	$\xi_3=\xi_4$	
13	11	9	9	9	$\xi_2=\xi_6$	
12	12	10	10	10	$\xi_2=\xi_3$	
12	12	10	10	8	$\xi_4=\xi_6$	
12	10	10	10	10	$\xi_1=\xi_4$	
12	8	8	8	8		$\xi_1=\xi_5$
11	11	11	11	9	$\xi_1=\xi_2$	
11	11	11	9	9		$\xi_2=\xi_5$
10	10	10	8	8		$\xi_1=\xi_4$

これらの図表を使って第７表 y_2 を実現する回路を導いてみよう．なお，この関数は excess-3 コードで偶数のときに出力が1となり，それ以外のときには出力が0となるような関数である．さて，まず最初にするべきことは，この関数を５変数の自己双対関数に変換することである．その結果**第13表**が得られるので，

第12表

(a)						(b)					
ξ_1	ξ_2	ξ_3	ξ_4	ξ_5	ξ_6	ξ_1	ξ_2	ξ_3	ξ_4	ξ_5	ξ_6
0	0	0	0	1	0	0	0	0	0	1	1
0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	1	1	0	0	0	0	1	1	0
0	0	0	1	1	1	0	0	0	1	1	1
0	0	1	0	1	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	1	0	0	1
0	1	0	0	0	1	0	0	1	0	1	0
0	1	0	0	1	0	0	0	1	0	1	1
0	1	0	0	1	1	0	0	1	1	0	1
0	1	0	1	0	0	0	0	1	1	1	0
0	1	0	1	1	0	0	0	1	1	1	1
0	1	0	1	1	1	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0	1
1	0	0	0	0	1	0	1	0	0	1	0
1	0	0	0	1	0	0	1	0	0	1	1
1	0	0	0	1	1	0	1	0	1	0	1
1	0	0	1	0	0	0	1	0	1	1	0
1	0	0	1	0	1	0	1	0	1	1	1
1	0	0	1	1	0	0	1	1	0	1	1
1	0	0	1	1	1	0	1	1	1	0	1
1	0	1	0	1	0	0	1	1	1	1	0
1	1	0	0	0	0	0	1	1	1	1	1
1	1	0	0	0	1	1	0	0	0	1	1
1	1	0	0	1	0	1	0	0	1	0	1
1	1	0	0	1	1	1	0	0	1	1	0
1	1	0	1	0	0	1	0	0	1	1	1
1	1	0	1	1	0	1	0	1	0	1	1
1	1	0	1	1	1	1	1	0	0	1	1
1	1	1	0	1	0	1	1	1	0	1	1
1	1	1	1	0	0	1	1	1	1	0	1
1	1	1	1	1	0	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1

第13表

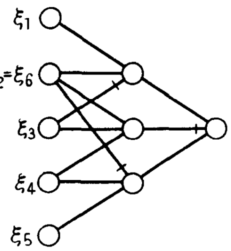
x_0	x_1	x_2	x_3	x_4	x_0	x_1	x_2	x_3	x_4	x_0	x_1	x_2	x_3	x_4
0	1	1	1	1	0	1	0	0	1	0	0	0	1	1
0	1	1	1	0	1	0	1	1	1	0	0	0	1	0
0	1	1	0	1	0	0	1	1	1	0	0	0	0	1
1	0	0	1	1	1	1	0	0	1	0	0	0	0	0
0	1	0	1	1	0	0	1	0	1					
1	0	1	0	1	1	1	0	1	1					

入力変数 (x_0, x_1, x_2, x_3, x_4) について1の個数は (5, 7, 7, 9, 13) であり，parity は7であることがわかるから，T数は (13, 11, 9, 9, 9；7) であることを知る．したがって第10表より，この関数（第７表 y_2）は第２グループに属する――すなわち２段で実現できる――ということがわかる．そこで第11表を見ると，T数 (13, 11, 9, 9, 9；7) なる自己双対関数は第5図 (a) において $\xi_2=\xi_6$ とすればよく，その結果**第6図**のように実現される．そうして第12表と第13表と

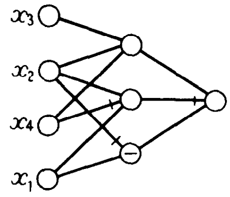
を比べることによって

$$\begin{cases} \xi_1 = x_3 \\ \xi_2 = \xi_6 = x_2 \\ \xi_3 = x_4' \\ \xi_4 = x_1 \\ \xi_5 = x_0' \end{cases}$$

なる対応関係があることがわかるので，x_0 を定数 1（+）に固定して結局第7表 y_2 を実現する回路として**第7図**に示したものが得られる．



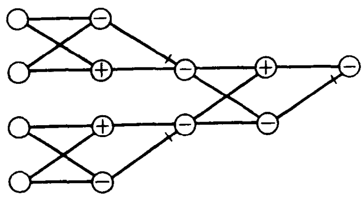
第6図



第7図

4.3 第3グループ——parity 関数

与えられた関数が第7表 y_3 のような parity 関数の場合，それは**第8図**に示す回路によって実現される．parity 関数とは，4変数16個の入力ベクトルのうち1の個数が偶数個の8個または奇数個の8個に限って出力が1となる関数であって，偶数 parity 関数と奇数 parity 関数との二通りある．全部で 64,594 通りある4変数論理関数のうちで3段以下で実現することのできない関数は，ここに掲げた二通りの関数に限り，残りの64,592通りの関数はすべて2段または3段で実現することができるのである．以下にその理由を述べる．

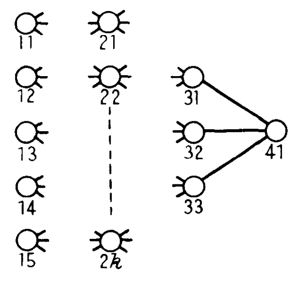

第8図

4変数および5変数の自己双対関数は，第4表によれば全部で80種類ある．そのうちで5変数の parity 関数を除く残りの79種類は，すべて2段または3段で実現する回路を導くことができる（2段で実現できるものについては **4.1** および **4.2** にその作り方を示した．また3段で実現できるものについては **4.4** および **4.5** にその作り方を示す）．

一方5変数の parity 関数から，一つの変数を定数に固定してできる関数は4変数の parity 関数に限る．たとえば5変数の奇数 parity 関数において，一つの変数を定数1に固定したとすれば，それは4変数の偶数 parity 関数になるからである．

以上の議論から，全部で 64,594 通りある4変数の論理関数のうち，偶数奇数二通りの parity 関数を除く残りの 64,592 通りの関数はすべて2段または3段で実現できることがわかった．

つぎに4変数の parity 関数が2段または3段では実現することができないことを示そう．そのためには5変数の parity 関数について同じことを証明すればよい．そこでいま5変数の parity 関数が**第9図**に示すごとく，初段5個の入力から1段目 k 個のパラメトロンを経て2段目3個のパラメトロンへ適当に結合され，その3個からすべて順結合で出力41へ結合することによって実現されたとしよう（この形式に限っても一般性は失われていないことは明らかであろう）．そうすると，ここに現れた 31，32，33 の三つのパラメトロンの実現している関数のうちの少なくとも一つは，parity 関数との距離が5以下でなければならない．



第9図

二つの論理関数 A と B との距離とは，A を1とするすべての入力ベクトルを B に与えたとき，B の出力が0となる入力ベクトルの個数を言う．いまの例でいえば，第9図41 で偶数 parity 関数が実現されているとすれば，1の個数が偶数個である16個の入力ベクトルを 11，12，…，15 に与えたときに，いま問題にしている2段目のパラメトロンが何回0になるかということである．そうして3個のうちの少なくとも

1 個は，0 となる回数が 5 回以下であることが必要である．何故ならば，16 個の入力ベクトルのうちのある 1 個の入力によって 31，32，33 の三つのパラメトロンのうちの二つまたは三つが同時に 0 になることは許されないから，これら三つのパラメトロンに 0 が現れる回数は総計でたかだか 16 回である．したがって，三つのすべてに 0 が 6 回以上ずつ現れては 41 に望む parity 関数を実現することは不可能だからである．

さて parity 関数との距離は，その関数の parity に等しい．そこで 5 変数以下の自己双対関数で 2 段以下で実現可能な計 19 種類の関数の parity を調べてみると，それらはすべて 6 以上である．したがって，第図の 31，32，33 に該当する関数は存在しないことになる．以上で 4 変数の parity 関数が 3 段以下で実現することができないことが証明されたので，4 段で実現されている第 8 図の回路は最少段数の条件を満足しているのである．

4.4　第 4 グループ——3 段で実現可能な関数（1）

4 変数の大部分の関数は，この節に述べる方法によって実現されるのであるが，その正確な個数はまだ計算できていない．

このような関数の設計方法を第 7 表 y_4 を例にとって説明する．なお，この関数は excess-3 コードの 0～9 のときに出力が 1 となり，それ以外のときには 0 となるような関数である．まず最初にこの関数を 5 変数の自己双対関数に変換する．その結果**第 14 表**が得られるので T 数（10, 8, 8, 8, 8；6）となり，第 10 表よりこの関数は本節 **4.4** に述べる方法に従えば実現できるということがわかる．

第 14 表

x_0	x_1	x_2	x_3	x_4	x_0	x_1	x_2	x_3	x_4	x_0	x_1	x_2	x_3	x_4
0	1	1	1	1	1	0	1	1	0	1	1	1	0	0
0	1	1	1	0	1	0	1	1	1	0	0	0	1	0
0	1	1	0	1	1	1	0	0	0	0	0	0	0	1
1	0	0	1	1	1	1	0	0	1	0	0	0	0	0
1	0	1	0	0	1	1	0	1	0					
1	0	1	0	1	1	1	0	1	1					

つぎに（x_0, x_1, x_2, x_3, x_4）の五つの入力変数のうちから任意の 2 変数 x_u, x_v を選んで，いま作った 16 個の入力ベクトルを $x_u=x_v$ なるものと $x_u \neq x_v$ なるものとに分ける．第 14 表の場合 x_1 と x_2 とを選ぶと**第 15 表**（a），（b）に示す 8 個・8 個に分れる．このようにして分けた結果はかならず 8 個・8 個になるが，その各 8 個を x_u を除いた 4 変数の論理関数と見做せばその関数は 4 変数以下の自己双対関数である．第 15 表の例では（a）は T 数（6, 4, 4, 4）の（x_0, x_2, x_3, x_4）の関数であり，（b）は 1 変数 x_0 そのものである．

第 15 表

(a)　$x_1=x_2$

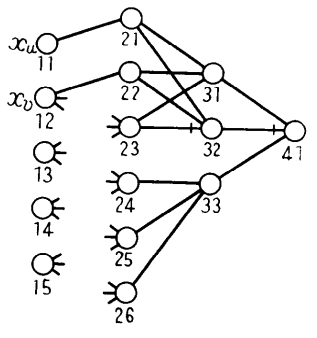
x_0	x_1	x_2	x_3	x_4
0	1	1	1	1
0	1	1	1	0
0	1	1	0	1
1	0	0	1	1
1	1	1	0	0
0	0	0	1	0
0	0	0	0	1
0	0	0	0	0
2		4	4	4

(b)　$x_1 \neq x_2$

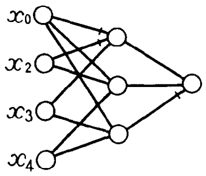
x_0	x_1	x_2	x_3	x_4
1	0	1	0	0
1	0	1	0	1
1	0	1	1	0
1	0	1	1	1
1	1	0	0	0
1	1	0	0	1
1	1	0	1	0
1	1	0	1	1
8		4	4	4

この例のように 8 個・8 個に分けた少なくとも一方が 1 変数関数になるか，または T 数（3, 3, 3）の 3 変数関数になるならば，ただちに回路図を書くことができる．すなわち $x_u=x_v$ と $x_u \neq x_v$ とに分けた少なくとも一方の関数の T 数が（8, 4, 4, 4）か（6, 6, 6, 4）かになるような 2 変数 x_u, x_v が見つかれば，その関数を実現する回路がただちに導き得るということである．

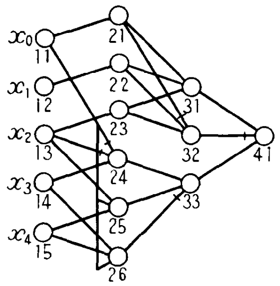
その作り方を**第 10 図**に示す．いま 8 個・8 個に分けたうちの（8, 4, 4, 4）か（6, 6, 6, 4）ではない方の関数をパラメトロン 33 で作る．4 変数以下の自己双対関数はかならず 2 段以下で実現できるから，この位置に作り得る．一方（8, 4, 4, 4）か（6, 6, 6, 4）をパラメトロン 23 で実現する．そうして，この二つの出力 23 と 33 とを，11 と 12（x_u と x_v）との一致か不一致かで図のように切り換えて，最終出力パラメトロン 41 に引き出すのである．



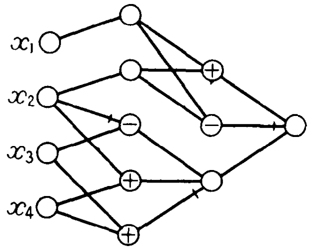
第 10 図



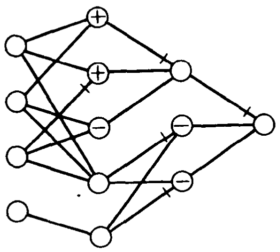
第 11 図



第 12 図



第 13 図



第 15 図

第 14 表に示した関数の場合には，その回路は**第 12 図**のように実現される．まず第15表 (a) の関数を**4.1**に示した方法に従って作ると**第 11 図**が得られる．この回路が第 12 図の 24, 25, 26, 33 の所に書かれている．一方第 15 表 (b) の関数は第 12 図の 21 によって実現されている．そうして x_1 と x_2 すなわち 12 と

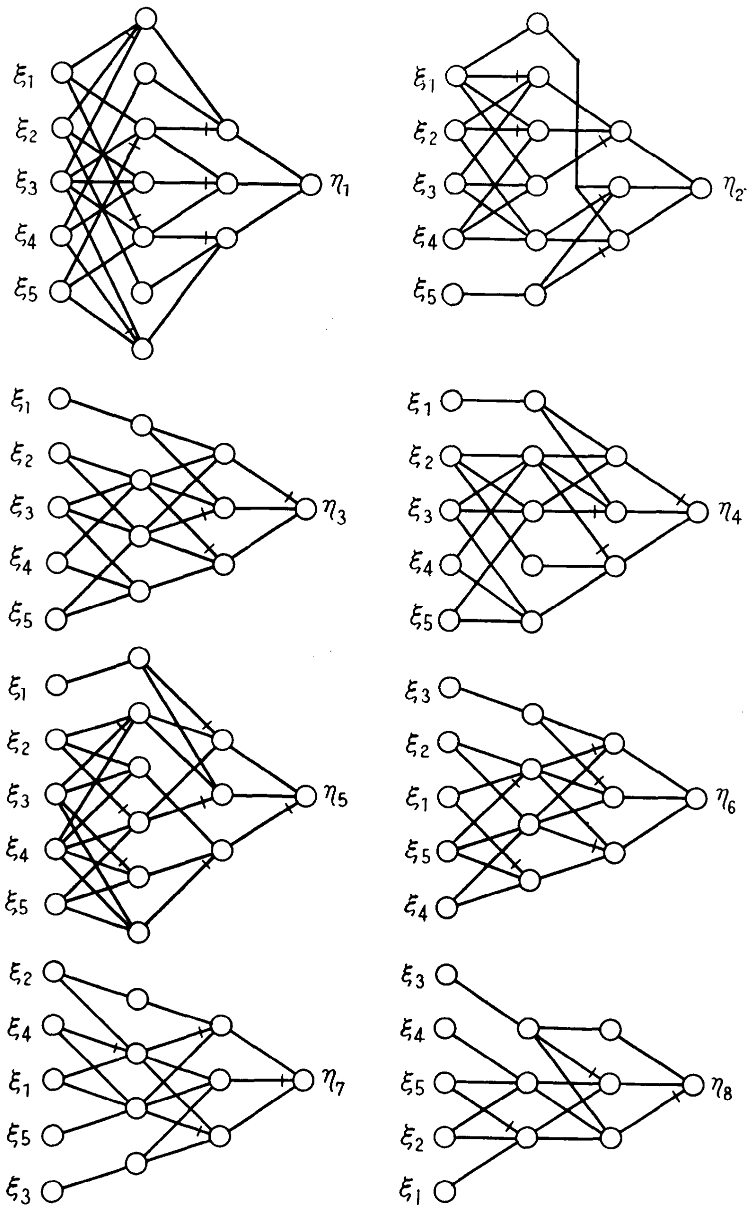
第 16 表

入力					出力							
ξ_1	ξ_2	ξ_3	ξ_4	ξ_5	η_1	η_2	η_3	η_4	η_5	η_6	η_7	η_8
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	1	1
0	1	0	0	0	0	0	0	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1
0	1	1	0	0	1	1	0	0	0	0	0	0
1	0	0	0	1	1	1	1	1	0	0	0	0
1	0	1	0	0	1	1	1	1	1	0	0	0
0	1	0	1	0	1	1	1	1	1	1	1	0
0	0	0	1	1	1	1	1	1	1	1	1	1
1	1	0	0	0	1	1	1	1	1	1	1	1
1	0	0	1	0	1	1	1	1	1	1	1	1
0	1	0	0	1	1	1	1	1	1	1	1	1
0	0	1	0	1	1	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	1	1	1	1	1
1	1	0	0	1	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0	0
0	1	1	0	1	0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0
1	0	1	0	1	0	0	0	0	0	0	0	1
0	1	0	1	1	0	0	0	0	0	1	1	1
0	1	1	1	0	0	0	0	0	1	1	1	1
1	0	0	1	1	0	0	1	1	1	1	1	1
0	1	1	1	1	1	0	0	0	0	0	0	0
1	0	1	1	1	1	1	1	0	0	0	0	0
1	1	0	1	1	1	1	1	1	1	1	0	0
1	1	1	0	1	1	1	1	1	1	1	1	1
1	1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1
					(9, 9, 9, 9; 1)	(10, 8, 8, 8; 2)	(11, 9, 9, 9; 3)	(10, 10, 8, 8; 4)	(9, 9, 9, 9; 5)	(10, 10, 10, 8; 6)	(9, 9, 9, 9; 7)	(8, 8, 8, 8; 8)

13 とが等しければ，33 が無条件で 41 に出るように，また 12 と 13 とが異なれば 21 が 41 に出るようにするのである．このようにして第 14 表に示した関数を実現する回路ができたらあとは x_0（パラメトロン 11）を 1(+) に固定すればよく，その結果第 7 表 y_4 を実現する回路として**第 13 図**を導くことができる．

4.5　第 5 グループ——3 段で実現可能な関数（2）

3 段で実現可能な 5 変数の自己双対関数（計 63 種類）のうち，これから示す 8 種類の関数は，前節 **4.4** に述べた（x_u, x_v）が存在しない．**第 16 表**および**第**



第 14 図

14 図にこれら 8 種類の関数の代表関数とその実現回路を示す.

第 7 表 y_5(2 out of 4) を例にとってみると，5 変数自己双対関数に変換して T 数 (10, 8, 8, 8, 8; 2) を得る. この関数は第 16 表および第 14 図に η_2 として用意されているので，ξ_1 を定数一に固定して**第 15 図**を得，これが 2 out of 4 を実現する回路である.

5. あとがき

3 入力パラメトロンによって 4 変数の論理関数を，可能な最少段数で実現する方法について述べた. 全部で 64,594 通りある 4 変数の論理関数のうちで，3 段以下で実現不能なものは偶数奇数二通りの parity 関数に限り，残りの 64,592 通りの関数は 2 段あるいは 3 段で実現できることがわかった.

ここに述べた方法が，関係各位に御利用いただければ幸甚である.

参考文献

1) 高橋秀俊：計算機械，岩波講座，現代応用数学，B 14-aⅡ，岩波書店，東京 (1958)

2) 2 段で実現できる 4 変数関数については次の文献がある.
喜安善一，池野信一：パラメトロン回路集(4)，電子科学，Vol. 9, No. 10 (1959)

3) 戸田　巌：自己双対論理関数の型の個数について，情報処理，Vol. 2, No. 1 (1961)

4) 喜安善一：ディジタル回路の数学，現代エレクトロニクス選書 34，共立出版，東京 (1961)

(昭和 38 年 6 月 3 日受付)