

論 文

シフトレジスタ形しきい値素子回路網の計算機設計*

阿 江 忠**

Abstract

With respect to the design of an autonomous sequential circuit, a unified theory using linear algebra on Galois field is well-known. In this paper we discuss the design of a shift-register-type sequential circuit using a threshold element. In general the behavior of a sequential circuit using threshold elements has been rather investigated. On the other hand, the synthesis of such circuits seems to be difficult, since the synthesis of a threshold function itself requires a number of procedures.

Therefore we introduce a method using a computer and make a program for the purpose of hardware realization. One problem is how to treat "don't cares" since a required output sequence has not usually the length equal to 2^m , where m is the number of shift registers.

In order to explain this treatment we use the concept of problem solving method. The method to decide whether a function f is a threshold one or not depends on 2-asummability.

As a result a program for design is obtained within $m=7$ and it is shown to be profitable.

1. まえがき

しきい値素子回路網の順序機械としての動作については、自己組織系の動作モデル¹⁾として、あるいは、パターン認識における学習モデルとしての位置づけがもっとも適していると考えられるため、オートノマス動作に関して主に研究がなされている^{2)~4)}.

一方、出力周期系列を与えたとき、そのような出力を与えるオートノマス動作をするしきい値素子回路網を設計する問題を“構成問題”と呼ぶとき、構成問題の一般的な解法はしきい値素子自体のもつ性質の難しさのため必ずしも容易ではない⁵⁾。メモリ素子を n 個用いる回路網は 2^n 個の状態数をもつが、一般には周期系列の長さは 2 のべきであるとは限らない。したがって構成問題においてしきい値関数を決定しようとする場合、与えられた条件には閾値が 0, 1 のどちらでもよい定義域が生じる。すなわち、Don't Care (以下 D.C と略す) が存在するから、計算機設計のプログラ

ム作成にあたっては、このことを考慮に入れなければならない。

ところで、所望の出力周期系列を発生するオートノマス順序回路の設計は、シフトレジスタ形モデルに関してすでに統一的方法が確立されている⁶⁾。ただし、ガロア体上の議論であるから、ゲート回路は ring sum 演算となり、これよりしきい値ゲートへの変換法は一般にはできない。

本稿においては、構成問題の複雑さを軽減するため、ひとまず、シフトレジスタ形構成のしきい値素子回路網を対象に構成問題を論じた。上述のガロア体上の議論のような統一的表現はできないから、アルゴリズム的方法を用いなければならない。したがって、いかに能率のよい計算機設計をするかが問題となる。

本稿では Problem Solving 的な見地から問題をとりあげ、所望の周期系列に対しきい値素子を真理値表で与える計算機設計法を以下で述べる。

2. シフトレジスタ形しきい値素子回路網

本稿の設計問題はつきのように要約される。所望の周期系列を a_1, a_2, \dots, a_n ($a_i \in \{0, 1\}$ とし、あとは同じ系列を繰返す) と与えたとき Fig. 1 に示すシフト

* Computer Design of a Threshold Logic Network whose memories are Shift Resistors, by Tadashi AE (Dept. of Electronics, Fac. of Eng., Hiroshima University)

** 広島大学工学部電子工学科

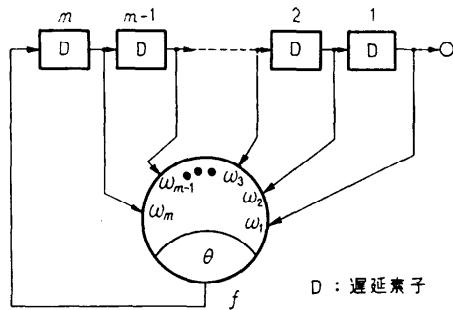


Fig. 1 Shift-type Sequential circuit with a threshold

Table 1 Relation between a output sequence and a memory one.

| 時刻 | m | $m-1 \dots$ | 3 | 2 | 1 |
|----------|-----------|------------------|-----------|-----------|-----------|
| 0 | a_m | $a_{m-1} \dots$ | a_3 | a_2 | a_1 |
| 1 | a_{m+1} | $a_m \dots$ | a_4 | a_3 | a_2 |
| 2 | a_{m+2} | $a_{m+1} \dots$ | a_5 | a_4 | a_3 |
| \vdots | \vdots | \vdots | \vdots | \vdots | \vdots |
| m | a_{2m} | $a_{2m-1} \dots$ | a_{m+4} | a_{m+3} | a_{m+1} |
| \vdots | \vdots | \vdots | \vdots | \vdots | \vdots |
| $n-1$ | a_{m-1} | $a_{m-2} \dots$ | a_3 | a_2 | a_1 |
| n | a_m | $a_{m-1} \dots$ | a_3 | a_2 | a_1 |

レジスタ形しきい値素子回路網において（初期条件を適当に与えることにより）出力端子から順次この系列を発生させるような Fig. 1 のしきい値素子を求めることがある。このとき出力周期系列の周期長 n に対し必要なシフトレジスタの数 m の下限は $\log_2 n$ であるから、 m はこの値にできるだけ近いことが望ましい。

このような問題の設定において、しきい値関数 f が満たさなければならない関係式は、時刻とともに Table 1 のようにシフトレジスタのメモリは遷移するから、

$$\left. \begin{aligned} a_{m+1} &= f(a_1, a_2, \dots, a_m) \\ a_{m+2} &= f(a_2, a_3, \dots, a_{m+1}) \\ \cdots & \\ a_{m+i} &= f(a_i, a_{i+1}, \dots, a_{i+m-1}) \\ \cdots & \\ a_n &= f(a_{n-m}, a_{n-m+1}, \dots, a_{n-1}) \\ a_1 &= f(a_{n-m+1}, a_{n-m+2}, \dots, a_n) \end{aligned} \right\} \quad (1)$$

となる。出力周期系列の長さ n が $n=2^m$ のときはよいが、一般には $n < 2^m$ となるからしきい値関数は D.C を生じる。

なお、 $m=n$ とすれば trivial に f が決められることからわかるように m を最大 n まで許すなら、しきい値関数 f の存在はつねに保証される。

所望の出力周期系列は、過渡状態の系列をもたない

ものとしたが、過渡状態の系列をもつ場合もほとんど同じ方法でとり扱える。

3. しきい値関数と Don't Care

一般にブール関数がしきい値関数であるための必要十分条件としてつきの定理が知られている。

(定理 1)⁷⁾

ブール関数が一個のしきい値関数となるための必要十分条件は、オンセットとオフセットを互いに異なる半空間に分離する線形な超平面が存在することである。

なお、オンセットとは n 変数ブール関数を 2^n 次元超立方体で表わしたとき、関数値が 1 となる頂点の集合であり、オフセットは関数値が 0 となる頂点の集合である。

与えられたブール関数が、しきい値関数であるか否かを実際に判定する方法に関しては、いろいろな方法が提案されている^{7,8)}。しかしながら、実際に計算機プログラム化するという観点からはやはりつきの asummability が適当であると思われる。

(定義)⁷⁾

ブール関数 F においてオンセットの頂点のうちから重複を許してとり出したものを x_1, \dots, x_k 、オフセットの頂点から同様にしてとり出したものを y_1, \dots, y_k とする。このとき、 $\sum_{i=1}^k x_i = \sum_{i=1}^k y_i$ となる頂点の選び方があれば F は k -summable であるという。また、頂点をどのように選んでも上式の等号が成立しないとき F は k -asummable であるという。もし、任意の k に対して k -asummable であれば単に asummable という。

(定理 2)⁷⁾

ブール関数 F がしきい値関数であるための必要十分条件は F が asummable であることである。

なお、変数の数 n が 7 以下ならば k -asummable の k は 2 で十分なことが知られている⁷⁾。（その他詳細は文献 9) に詳しい。)

関数がしきい値関数であると判定できたら、D.C をもつ関数の場合これを完全に埋めなければしきい値関数は完全には決定しないので具体的な回路設計ができない。

ところで、しきい値関数の判定と D.C がある場合の関数構造の決定を別々に行なうこととは (heuristic な方法のときもそうであるが) 計算機アルゴリズムをつくる場合には得策ではない。D.C を割当てるアルゴ

リズム、すなわち関数構造決定のアルゴリズムが完全であれば構造決定のできない関数はしきい値関数でないと判定できる。

このような観点に基づく計算機用の関数構造決定アルゴリズムを次に述べる。

D.C の処理方法は、人間が行なうと機械が行なうのとでは大きな違いがある。人間は全体を把握したのち細かい操作を行なえるから、変数の数が少ないときや極端にオンセットやオフセットが少なく直観的思考の働くときはさほど D.C の処理を考えなくてもよい。しかしながら、計算機処理においては、たとえ変数が少なくてもきっちりアルゴリズム化しておく必要が生じる。D.C に能率よく関数値 0 または 1 を割当てようとするとき次のような方法が考えられる。(まず、 $n \leq 2^m$ となる最小の m が与えられたとする。)

(1) 瞬時割当て法

D.C すべてに対し関数値 0 あるいは 1 をそれぞれ同時に割当てる。詳しくは次のような区別ができる。

(1-1) D.C すべてに対し関数値割当てを適当に(例えば、ランダム)に行ない、関数の判定を行なう。しきい値関数にならないときは割当てをやり直す。

(1-2) D.C を含む関数、すなわち不完全定義関数に対する判定法⁹⁾でしきい値関数か否かを判定し、かかるのち正確な D.C 割当てを行なう。

(1-1) の方法は D.C のない完全定義関数に対する判定法ですむという利点はあるが、最大手数は D.C の数を $p (=2^m - n)$ とするとき 2^p で与えられる。もちろん関数がしきい値に“近い”とき、すなわち、どのような D.C 割当てでもしきい値関数となるときは実際の手数はずっと少なくてすむ。しかし、最悪状態ではよい方法とはいえない。また、(1-2) の方法は不完全定義関数の asum. 判定は手数が増えること⁹⁾および、結局、正確な D.C 割当てアルゴリズムが必要になるという点で実用的でない。

そこで本稿では次の(2)の方法を用いた。

(2) 逐次割当て法

D.C を1つずつ割当て、その都度一応の判定をしておく。

(2) の方法は D.C があっても完全定義関数の判定法を適用していき、もし、不都合が生じたらもとに戻ってやり直す方法で関数決定と判定が混在している。

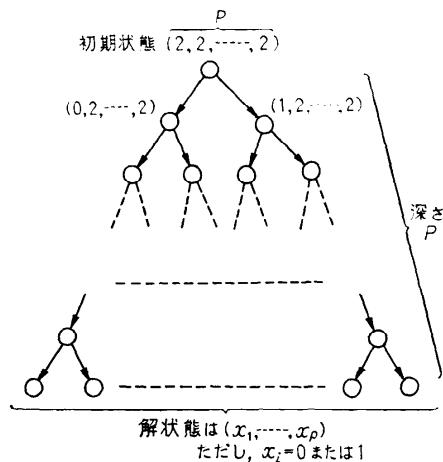


Fig. 2 Binary tree representation of don't care assignment.

すなわち、0(または1)を一つのD.Cに割当て完全定義関数に対する k -asum. 判定をする。しきい値ならば先へ進むし、非しきい値ならば逆に1(または0)とする。どちらも割当て不能ならばもとに一ステップ戻ってやり直す。(どのような割当ても不可能なら、はじめの m を $m+1$ にして繰返す。)

この方法を Problem Solving 的見地から考察するために Fig. 2 ように2進木(binary tree)表示する。このような状態空間(state-space)法¹⁰⁾の木表現で解を求める代表的方法には Depth-first 法と Breadth-first 法があるが、ともかく、しきい値関数を得るような割当て可能解を“一つ”求めることを目的とするならば Depth-first 法が望ましい。(もちろん、すべての解を求めるとするなら Breadth-first 法も考えられる。)この場合、解状態(goal state)に対応する頂点の深さ(木の根からの距離)はつねに P である。2進木の各頂点は P 次元ベクトルのラベルをもち、ベクトルの要素は未定(D.Cに相当)のときは2としておき、(2-2)の操作の一ステップごとに2を0(オフセット)または1(オンセット)に変える。操作のステップはベクトルの要素を下位から順に行なうとし、2進木において同じ深さの頂点どおりは辞書式順序で左からならべる。

オペレータは k -asum. 判定である。不完全定義関数の判定法(k^* -asum. 判定、一般には $k^* > k$)を用いればこれはつねに下向きであるが、この場合は前述のように上向きになることもある。初期状態(start-

state) の頂点のラベルは $(2, 2, \dots, 2)$ であらわし, $\overbrace{P}^{\rightarrow}$ D.C がすべて割当てられた状態, つまり, 解状態の頂点のラベルは (x_1, \dots, x_P) (ただし, $x_i=0$ または 1) で与えられる.

4. 設計アルゴリズムの大要

設計アルゴリズムの要約は Fig. 3 のとおりである. これは主プログラムのフローチャートをあらわしているが, k -asum. 判定の k は 2 としている. ここで手数のかかるのは D.C を割当てるサブルーチンの作成である. 実際に作成したサブルーチンの理論的根拠はつきの命題による.

[命題 3]

D.C のあるブール関数 f が D.C 以外の vertex の集合については asum. であるとしたとき, D.C を適当にうめることにより D.C のないしきい値関数 f' を得ることができる. (証明) 文献 5) 参照.

D.C の割当ては, オンセット (関数値が 1) とオフセット (関数値が 0) の 2 種が存在するが, どちらを先に試みればより能率的かを知ることはむつかしい. そこで本文のプログラムでは, 試行錯誤の順序として

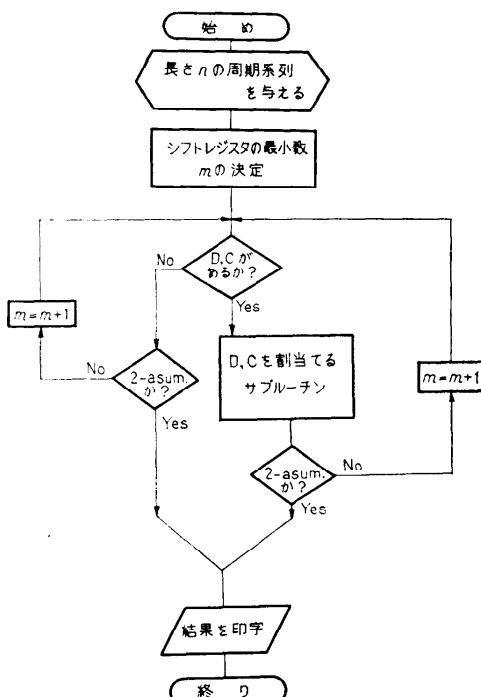


Fig. 3 Flowchart of main program.

オンセット優先, すなわち, まず関数値 1 を割当てしきい値か否かを判定し, “否” のときに関数値 0 を割当てる方法をとった*.

Fig. 2において i 番目 ($i=1, \dots, P$) の D.C がすでにオンセットに割当てられているとする. このとき, $i+1$ 番目の D.C がオンセット, オフセットともに割当不可能であれば i 番目に戻り, i 番目の D.C をオフセットに割当て可能か否かを見る. 可能ならばオフセットに割当てて $i+1$ 番目に進む. もし, 不可能なら $i-1$ 番目まで戻り同様のことを繰返す. この方法は Fig. 2 の 2 進木において最右端から解状態を探索することを意味する. 本文の設計問題では, 周期系列以外の系列には何ら条件をつけていないから, 2 進木の初期状態から解状態へ至るパスが一つ求まれば設計は完了する. しかし, そのようなパスが一つも存在しないときは, はじめに設定したシフトレジスタの数 m では実現できないから, m を $m+1$ として同じことを繰返す. 前述のように周期系列の長さ n 以下の値で必ず設計は完了する.

実際のプログラムを Fig. 4 (次頁参照) に, 結果の例を Fig. 5 (p. 28 参照) に示す, なお, Fig. 3 のフローチャートでは途中の印字は省略しているが, 実際は参考のため印字している. しきい値関数か否かの判定は 2-asum. で行なっているためシフトレジスタの数 m の上限は 7 である. 実際, このプログラムで周期系列の長さ n が 14 までの設計を行なわせたところ (すべてを試みたわけではないが) この上限内のシフトレジスタ数すべて構成可能となった. なお, 一つの周期系列に対する解を求める計算時間は長くとも数分 (TOSBAC 3400) である.

5. 考 察

しきい値関数 f を本文では真理値表として求めた. このことは変数が増すと不都合ではあるが, 実際 f をしきい値関数 1 個で実現しようとするときハードウェア的には, (i) 変数の個数には上限があること, (ii) 回路設計上真理値表が必要となることが多いこと, という理由からこれを採用した. 他の表現法 (たとえば特徴パラメータなど*) は今のところハードウェア向きではないと思われる. (ii) の理由について補足すると, 重み, しきい値で表現したしきい値関数の

* Problem Solving 的にいえば heuristic function が定まれば適当な探査法が示されることになるが今のところむつかしいと思われる.

```

      RUNS1000,LINE1000C
1      CRUNCH A(1)
2      CRUNCH A(2),IA(20)
3      1 READ 5,100,1 (A11),I=1,20.
4      DO 100 I=1,20
5      IF 1A1L,100,2,03 GO TO 200
6      95 CONTINUE
7      300 IL,W,I
8      DO 100 T=1,IL
9      52 I(L)=A(I)
10     CHTIC(6,100); TL
11     W=L+1,I=1,IL-1
12     WHTIE(4,100) ((A11),I=1,IL)
13     I=(IL,6,0) GO TO 200
14     DO 100 I=1,IL
15     200 WHTIE(6,100)
16     GO TO 1
17     DO 55 J=1,7
18     J=J+1
19     IF 1(J,1),100 GO TO 301
20     95 CONTINUE
21     300 IL,W,I
22     54 RTR(I,A(I))
23     DO 55 T=1,IL
24     100 I=1,IL
25     95 BTDTIA(I)
26     CALL BTDT,I,IL,W,I
27     GO 100
28     100 FORMAT(1,0)
29     101 FORMAT(1,*," GIVEN PERIODIC SEQUENCE ***,5X,* THE LENGTH
30     NOE THE GIVEN SEQUENCE *1,3)
31     102 FORMAT(1,0,5X,* POSSIBLE *1,5X,*0-VARIABLE (WITHOUT DELAY=*
32     *ELM=1)*1)
33     END
34
35     F70
36     SUMMATION DIV(J,IL)
37     CRUNCH A(1)
38     DIVIDE(1,IL,100,7,71,1D(10,7),1D(10,0),1D(10,0),1D(10,0),1D(10,0),1D(10,0),
39     1D(10,0),1D(10,0),1D(10,0),1D(10,0),1D(10,0),1D(10,0),1D(10,0),1D(10,0))
40     JU1N
41     DO 100 T=1,IL
42     If(I(J,1),1,0,1,0) GO TO 303
43     DO 10 50
44     DO 57 I=1,IL
45     K=K+1
46     57 I=(J,I),1,I,K
47     58 W=I
48     J=J+1
49     DO 59 T=1,IL
50     If(I(J,1),1,0,1,0) GO TO 304
51     DO 59 I=1,IL
52     K=K+1
53     59 I=(J,I),1,I,K
54     58 CR1J,I,E
55     K=K+1
56     J=J+1
57     DO 58 W=I,J,I
58     J=J+1
59     DO 60 T=1,IL
60     If(I(J,1),1,0,1,0) GO TO 304
61     DO 60 I=1,IL
62     K=K+1
63     60 I=(J,I),1,I,K
64     61 W=I
65     J=J+1
66     DO 67 T=1,IL
67     If(I(J,1),1,0,1,0) GO TO 304
68     DO 67 I=1,IL
69     K=K+1
70     67 I=(J,I),1,I,K
71     68 W=I
72     J=J+1
73     DO 74 T=1,IL
74     If(I(J,1),1,0,1,0) GO TO 304
75     DO 74 I=1,IL
76     K=K+1
77     74 I=(J,I),1,I,K
78     75 W=I
79     J=J+1
80     DO 81 T=1,IL
81     If(I(J,1),1,0,1,0) GO TO 304
82     DO 81 I=1,IL
83     K=K+1
84     81 I=(J,I),1,I,K
85     82 W=I
86     J=J+1
87     DO 88 T=1,IL
88     If(I(J,1),1,0,1,0) GO TO 304
89     DO 88 I=1,IL
90     K=K+1
91     88 I=(J,I),1,I,K
92     93 W=I
93     J=J+1
94     DO 95 T=1,IL
95     If(I(J,1),1,0,1,0) GO TO 304
96     DO 95 I=1,IL
97     K=K+1
98     95 I=(J,I),1,I,K
99     100 W=I
100    J=J+1
101    DO 102 T=1,IL
102    If(I(J,1),1,0,1,0) GO TO 304
103    DO 102 I=1,IL
104    K=K+1
105    104 W=I
106    J=J+1
107    DO 108 T=1,IL
108    If(I(J,1),1,0,1,0) GO TO 304
109    DO 108 I=1,IL
110    K=K+1
111    108 I=(J,I),1,I,K
112    109 W=I
113    J=J+1
114    DO 115 T=1,IL
115    If(I(J,1),1,0,1,0) GO TO 304
116    DO 115 I=1,IL
117    K=K+1
118    115 I=(J,I),1,I,K
119    120 W=I
120    J=J+1
121    DO 122 T=1,IL
122    If(I(J,1),1,0,1,0) GO TO 304
123    DO 122 I=1,IL
124    K=K+1
125    122 I=(J,I),1,I,K
126    127 W=I
127    J=J+1
128    DO 129 T=1,IL
129    If(I(J,1),1,0,1,0) GO TO 304
130    DO 129 I=1,IL
131    K=K+1
132    129 I=(J,I),1,I,K
133    134 W=I
134    J=J+1
135    DO 136 T=1,IL
136    If(I(J,1),1,0,1,0) GO TO 304
137    DO 136 I=1,IL
138    K=K+1
139    136 I=(J,I),1,I,K
140    141 W=I
141    J=J+1
142    DO 143 T=1,IL
143    If(I(J,1),1,0,1,0) GO TO 304
144    DO 143 I=1,IL
145    K=K+1
146    143 I=(J,I),1,I,K
147    148 W=I
148    J=J+1
149    DO 150 T=1,IL
150    If(I(J,1),1,0,1,0) GO TO 304
151    DO 150 I=1,IL
152    K=K+1
153    150 I=(J,I),1,I,K
154    155 W=I
155    J=J+1
156    DO 157 T=1,IL
157    If(I(J,1),1,0,1,0) GO TO 304
158    DO 157 I=1,IL
159    K=K+1
160    157 I=(J,I),1,I,K
161    162 W=I
162    J=J+1
163    DO 164 T=1,IL
164    If(I(J,1),1,0,1,0) GO TO 304
165    DO 164 I=1,IL
166    K=K+1
167    164 I=(J,I),1,I,K
168    169 W=I
169    J=J+1
170    DO 171 T=1,IL
171    If(I(J,1),1,0,1,0) GO TO 304
172    DO 171 I=1,IL
173    K=K+1
174    171 I=(J,I),1,I,K
175    176 W=I
176    J=J+1
177    DO 178 T=1,IL
178    If(I(J,1),1,0,1,0) GO TO 304
179    DO 178 I=1,IL
180    K=K+1
181    178 I=(J,I),1,I,K
182    183 W=I
183    J=J+1
184    DO 185 T=1,IL
185    If(I(J,1),1,0,1,0) GO TO 304
186    DO 185 I=1,IL
187    K=K+1
188    185 I=(J,I),1,I,K
189    190 W=I
190    J=J+1
191    DO 192 T=1,IL
192    If(I(J,1),1,0,1,0) GO TO 304
193    DO 192 I=1,IL
194    K=K+1
195    192 I=(J,I),1,I,K
196    197 W=I
197    J=J+1
198    DO 199 T=1,IL
199    If(I(J,1),1,0,1,0) GO TO 304
200    DO 199 I=1,IL
201    K=K+1
202    199 I=(J,I),1,I,K
203    204 W=I
204    J=J+1
205    DO 206 T=1,IL
206    If(I(J,1),1,0,1,0) GO TO 304
207    DO 206 I=1,IL
208    K=K+1
209    206 I=(J,I),1,I,K
210    211 W=I
211    J=J+1
212    DO 213 T=1,IL
213    If(I(J,1),1,0,1,0) GO TO 304
214    DO 213 I=1,IL
215    K=K+1
216    213 I=(J,I),1,I,K
217    218 W=I
218    J=J+1
219    DO 220 T=1,IL
220    If(I(J,1),1,0,1,0) GO TO 304
221    DO 220 I=1,IL
222    K=K+1
223    220 I=(J,I),1,I,K
224    225 W=I
225    J=J+1
226    DO 227 T=1,IL
227    If(I(J,1),1,0,1,0) GO TO 304
228    DO 227 I=1,IL
229    K=K+1
230    227 I=(J,I),1,I,K
231    232 W=I
232    J=J+1
233    DO 234 T=1,IL
234    If(I(J,1),1,0,1,0) GO TO 304
235    DO 234 I=1,IL
236    K=K+1
237    234 I=(J,I),1,I,K
238    239 W=I
239    J=J+1
240    DO 241 T=1,IL
241    If(I(J,1),1,0,1,0) GO TO 304
242    DO 241 I=1,IL
243    K=K+1
244    241 I=(J,I),1,I,K
245    246 W=I
246    J=J+1
247    DO 248 T=1,IL
248    If(I(J,1),1,0,1,0) GO TO 304
249    DO 248 I=1,IL
250    K=K+1
251    248 I=(J,I),1,I,K
252    253 W=I
253    J=J+1
254    DO 255 T=1,IL
255    If(I(J,1),1,0,1,0) GO TO 304
256    DO 255 I=1,IL
257    K=K+1
258    255 I=(J,I),1,I,K
259    260 W=I
260    J=J+1
261    DO 262 T=1,IL
262    If(I(J,1),1,0,1,0) GO TO 304
263    DO 262 I=1,IL
264    K=K+1
265    262 I=(J,I),1,I,K
266    267 W=I
267    J=J+1
268    DO 269 T=1,IL
269    If(I(J,1),1,0,1,0) GO TO 304
270    DO 269 I=1,IL
271    K=K+1
272    269 I=(J,I),1,I,K
273    274 W=I
274    J=J+1
275    DO 276 T=1,IL
276    If(I(J,1),1,0,1,0) GO TO 304
277    DO 276 I=1,IL
278    K=K+1
279    276 I=(J,I),1,I,K
280    281 W=I
281    J=J+1
282    DO 283 T=1,IL
283    If(I(J,1),1,0,1,0) GO TO 304
284    DO 283 I=1,IL
285    K=K+1
286    283 I=(J,I),1,I,K
287    288 W=I
288    J=J+1
289    DO 290 T=1,IL
290    If(I(J,1),1,0,1,0) GO TO 304
291    DO 290 I=1,IL
292    K=K+1
293    290 I=(J,I),1,I,K
294    295 W=I
295    J=J+1
296    DO 297 T=1,IL
297    If(I(J,1),1,0,1,0) GO TO 304
298    DO 297 I=1,IL
299    K=K+1
300    297 I=(J,I),1,I,K
301    302 W=I
302    J=J+1
303    DO 304 T=1,IL
304    If(I(J,1),1,0,1,0) GO TO 304
305    DO 304 I=1,IL
306    K=K+1
307    306 I=(J,I),1,I,K
308    309 W=I
309    J=J+1
310    DO 311 T=1,IL
311    If(I(J,1),1,0,1,0) GO TO 304
312    DO 311 I=1,IL
313    K=K+1
314    311 I=(J,I),1,I,K
315    316 W=I
316    J=J+1
317    DO 318 T=1,IL
318    If(I(J,1),1,0,1,0) GO TO 304
319    DO 318 I=1,IL
320    K=K+1
321    318 I=(J,I),1,I,K
322    323 W=I
323    J=J+1
324    DO 325 T=1,IL
325    If(I(J,1),1,0,1,0) GO TO 304
326    DO 325 I=1,IL
327    K=K+1
328    325 I=(J,I),1,I,K
329    330 W=I
330    J=J+1
331    DO 332 T=1,IL
332    If(I(J,1),1,0,1,0) GO TO 304
333    DO 332 I=1,IL
334    K=K+1
335    332 I=(J,I),1,I,K
336    337 W=I
337    J=J+1
338    DO 339 T=1,IL
339    If(I(J,1),1,0,1,0) GO TO 304
340    DO 339 I=1,IL
341    K=K+1
342    339 I=(J,I),1,I,K
343    344 W=I
344    J=J+1
345    DO 346 T=1,IL
346    If(I(J,1),1,0,1,0) GO TO 304
347    DO 346 I=1,IL
348    K=K+1
349    346 I=(J,I),1,I,K
350    351 W=I
351    J=J+1
352    DO 353 T=1,IL
353    If(I(J,1),1,0,1,0) GO TO 304
354    DO 353 I=1,IL
355    K=K+1
356    353 I=(J,I),1,I,K
357    358 W=I
358    J=J+1
359    DO 360 T=1,IL
360    If(I(J,1),1,0,1,0) GO TO 304
361    DO 360 I=1,IL
362    K=K+1
363    360 I=(J,I),1,I,K
364    365 W=I
365    J=J+1
366    DO 367 T=1,IL
367    If(I(J,1),1,0,1,0) GO TO 304
368    DO 367 I=1,IL
369    K=K+1
370    367 I=(J,I),1,I,K
371    372 W=I
372    J=J+1
373    DO 374 T=1,IL
374    If(I(J,1),1,0,1,0) GO TO 304
375    DO 374 I=1,IL
376    K=K+1
377    374 I=(J,I),1,I,K
378    379 W=I
379    J=J+1
380    DO 381 T=1,IL
381    If(I(J,1),1,0,1,0) GO TO 304
382    DO 381 I=1,IL
383    K=K+1
384    381 I=(J,I),1,I,K
385    386 W=I
386    J=J+1
387    DO 388 T=1,IL
388    If(I(J,1),1,0,1,0) GO TO 304
389    DO 388 I=1,IL
390    K=K+1
391    388 I=(J,I),1,I,K
392    393 W=I
393    J=J+1
394    DO 395 T=1,IL
395    If(I(J,1),1,0,1,0) GO TO 304
396    DO 395 I=1,IL
397    K=K+1
398    395 I=(J,I),1,I,K
399    400 W=I
400    J=J+1
401    DO 402 T=1,IL
402    If(I(J,1),1,0,1,0) GO TO 304
403    DO 402 I=1,IL
404    K=K+1
405    402 I=(J,I),1,I,K
406    407 W=I
407    J=J+1
408    DO 409 T=1,IL
409    If(I(J,1),1,0,1,0) GO TO 304
410    DO 409 I=1,IL
411    K=K+1
412    409 I=(J,I),1,I,K
413    414 W=I
414    J=J+1
415    DO 416 T=1,IL
416    If(I(J,1),1,0,1,0) GO TO 304
417    DO 416 I=1,IL
418    K=K+1
419    416 I=(J,I),1,I,K
420    421 W=I
421    J=J+1
422    DO 423 T=1,IL
423    If(I(J,1),1,0,1,0) GO TO 304
424    DO 423 I=1,IL
425    K=K+1
426    423 I=(J,I),1,I,K
427    428 W=I
428    J=J+1
429    DO 430 T=1,IL
430    If(I(J,1),1,0,1,0) GO TO 304
431    DO 430 I=1,IL
432    K=K+1
433    430 I=(J,I),1,I,K
434    435 W=I
435    J=J+1
436    DO 437 T=1,IL
437    If(I(J,1),1,0,1,0) GO TO 304
438    DO 437 I=1,IL
439    K=K+1
440    437 I=(J,I),1,I,K
441    442 W=I
442    J=J+1
443    DO 444 T=1,IL
444    If(I(J,1),1,0,1,0) GO TO 304
445    DO 444 I=1,IL
446    K=K+1
447    444 I=(J,I),1,I,K
448    449 W=I
449    J=J+1
450    DO 451 T=1,IL
451    If(I(J,1),1,0,1,0) GO TO 304
452    DO 451 I=1,IL
453    K=K+1
454    451 I=(J,I),1,I,K
455    456 W=I
456    J=J+1
457    DO 458 T=1,IL
458    If(I(J,1),1,0,1,0) GO TO 304
459    DO 458 I=1,IL
460    K=K+1
461    458 I=(J,I),1,I,K
462    463 W=I
463    J=J+1
464    DO 465 T=1,IL
465    If(I(J,1),1,0,1,0) GO TO 304
466    DO 465 I=1,IL
467    K=K+1
468    465 I=(J,I),1,I,K
469    470 W=I
470    J=J+1
471    DO 472 T=1,IL
472    If(I(J,1),1,0,1,0) GO TO 304
473    DO 472 I=1,IL
474    K=K+1
475    472 I=(J,I),1,I,K
476    477 W=I
477    J=J+1
478    DO 479 T=1,IL
479    If(I(J,1),1,0,1,0) GO TO 304
480    DO 479 I=1,IL
481    K=K+1
482    479 I=(J,I),1,I,K
483    484 W=I
484    J=J+1
485    DO 486 T=1,IL
486    If(I(J,1),1,0,1,0) GO TO 304
487    DO 486 I=1,IL
488    K=K+1
489    486 I=(J,I),1,I,K
490    491 W=I
491    J=J+1
492    DO 493 T=1,IL
493    If(I(J,1),1,0,1,0) GO TO 304
494    DO 493 I=1,IL
495    K=K+1
496    493 I=(J,I),1,I,K
497    498 W=I
498    J=J+1
499    DO 500 T=1,IL
500    If(I(J,1),1,0,1,0) GO TO 304
501    DO 500 I=1,IL
502    K=K+1
503    500 I=(J,I),1,I,K
504    505 W=I
505    J=J+1
506    DO 507 T=1,IL
507    If(I(J,1),1,0,1,0) GO TO 304
508    DO 507 I=1,IL
509    K=K+1
510    507 I=(J,I),1,I,K
511    512 W=I
512    J=J+1
513    DO 514 T=1,IL
514    If(I(J,1),1,0,1,0) GO TO 304
515    DO 514 I=1,IL
516    K=K+1
517    514 I=(J,I),1,I,K
518    519 W=I
519    J=J+1
520    DO 521 T=1,IL
521    If(I(J,1),1,0,1,0) GO TO 304
522    DO 521 I=1,IL
523    K=K+1
524    521 I=(J,I),1,I,K
525    526 W=I
526    J=J+1
527    DO 528 T=1,IL
528    If(I(J,1),1,0,1,0) GO TO 304
529    DO 528 I=1,IL
530    K=K+1
531    528 I=(J,I),1,I,K
532    533 W=I
533    J=J+1
534    DO 535 T=1,IL
535    If(I(J,1),1,0,1,0) GO TO 304
536    DO 535 I=1,IL
537    K=K+1
538    535 I=(J,I),1,I,K
539    540 W=I
540    J=J+1
541    DO 542 T=1,IL
542    If(I(J,1),1,0,1,0) GO TO 304
543    DO 542 I=1,IL
544    K=K+1
545    542 I=(J,I),1,I,K
546    547 W=I
547    J=J+1
548    DO 549 T=1,IL
549    If(I(J,1),1,0,1,0) GO TO 304
550    DO 549 I=1,IL
551    K=K+1
552    549 I=(J,I),1,I,K
553    554 W=I
554    J=J+1
555    DO 556 T=1,IL
556    If(I(J,1),1,0,1,0) GO TO 304
557    DO 556 I=1,IL
558    K=K+1
559    556 I=(J,I),1,I,K
560    561 W=I
561    J=J+1
562    DO 563 T=1,IL
563    If(I(J,1),1,0,1,0) GO TO 304
564    DO 563 I=1,IL
565    K=K+1
566    563 I=(J,I),1,I,K
567    568 W=I
568    J=J+1
569    DO 570 T=1,IL
570    If(I(J,1),1,0,1,0) GO TO 304
571    DO 570 I=1,IL
572    K=K+1
573    570 I=(J,I),1,I,K
574    575 W=I
575    J=J+1
576    DO 577 T=1,IL
577    If(I(J,1),1,0,1,0) GO TO 304
578    DO 577 I=1,IL
579    K=K+1
580    577 I=(J,I),1,I,K
581    582 W=I
582    J=J+1
583    DO 584 T=1,IL
584    If(I(J,1),1,0,1,0) GO TO 304
585    DO 584 I=1,IL
586    K=K+1
587    584 I=(J,I),1,I,K
588    589 W=I
589    J=J+1
590    DO 591 T=1,IL
591    If(I(J,1),1,0,1,0) GO TO 304
592    DO 591 I=1,IL
593    K=K+1
594    591 I=(J,I),1,I,K
595    596 W=I
596    J=J+1
597    DO 598 T=1,IL
598    If(I(J,1),1,0,1,0) GO TO 304
599    DO 598 I=1,IL
600    K=K+1
601    598 I=(J,I),1,I,K
602    603 W=I
603    J=J+1
604    DO 605 T=1,IL
605    If(I(J,1),1,0,1,0) GO TO 304
606    DO 605 I=1,IL
607    K=K+1
608    605 I=(J,I),1,I,K
609    610 W=I
610    J=J+1
611    DO 612 T=1,IL
612    If(I(J,1),1,0,1,0) GO TO 304
613    DO 612 I=1,IL
614    K=K+1
615    612 I=(J,I),1,I,K
616    617 W=I
617    J=J+1
618    DO 619 T=1,IL
619    If(I(J,1),1,0,1,0) GO TO 304
620    DO 619 I=1,IL
621    K=K+1
622    619 I=(J,I),1,I,K
623    624 W=I
624    J=J+1
625    DO 626 T=1,IL
626    If(I(J,1),1,0,1,0) GO TO 304
627    DO 626 I=1,IL
628    K=K+1
629    626 I=(J,I),1,I,K
630    631 W=I
631    J=J+1
632    DO 633 T=1,IL
633    If(I(J,1),1,0,1,0) GO TO 304
634    DO 633 I=1,IL
635    K=K+1
636    633 I=(J,I),1,I,K
637    638 W=I
638    J=J+1
639    DO 640 T=1,IL
640    If(I(J,1),1,0,1,0) GO TO 304
641    DO 640 I=1,IL
642    K=K+1
643    640 I=(J,I),1,I,K
644    645 W=I
645    J=J+1
646    DO 647 T=1,IL
647    If(I(J,1),1,0,1,0) GO TO 304
648    DO 647 I=1,IL
649    K=K+1
650    647 I=(J,I),1,I,K
651    652 W=I
652    J=J+1
653    DO 654 T=1,IL
654    If(I(J,1),1,0,1,0) GO TO 304
655    DO 654 I=1,IL
656    K=K+1
657    654 I=(J,I),1,I,K
658    659 W=I
659    J=J+1
660    DO 661 T=1,IL
661    If(I(J,1),1,0,1,0) GO TO 304
662    DO 661 I=1,IL
663    K=K+1
664    661 I=(J,I),1,I,K
665    666 W=I
666    J=J+1
667    DO 668 T=1,IL
668    If(I(J,1),1,0,1,0) GO TO 304
669    DO 668 I=1,IL
670    K=K+1
671    668 I=(J,I),1,I,K
672    673 W=I
673    J=J+1
674    DO 675 T=1,IL
675    If(I(J,1),1,0,1,0) GO TO 304
676    DO 675 I=1,IL
677    K=K+1
678    675 I=(J,I),1,I,K
679    680 W=I
680    J=J+1
681    DO 682 T=1,IL
682    If(I(J,1),1,0,1,0) GO TO 304
683    DO 682 I=1,IL
684    K=K+1
685    682 I=(J,I),1,I,K
686    687 W=I
687    J=J+1
688    DO 689 T=1,IL
689    If(I(J,1),1,0,1,0) GO TO 304
690    DO 689 I=1,IL
691    K=K+1
692    689 I=(J,I),1,I,K
693    694 W=I
694    J=J+1
695    DO 696 T=1,IL
696    If(I(J,1),1,0,1,0) GO TO 304
697    DO 696 I=1,IL
698    K=K+1
699    696 I=(J,I),1,I,K
700    701 W=I
701    J=J+1
702    DO 703 T=1,IL
703    If(I(J,1),1,0,1,0) GO TO 304
704    DO 703 I=1,IL
705    K=K+1
706    703 I=(J,I),1,I,K
707    708 W=I
708    J=J+1
709    DO 710 T=1,IL
710    If(I(J,1),1,0,1,0) GO TO 304
711    DO 710 I=1,IL
712    K=K+1
713    710 I=(J,I),1,I,K
714    715 W=I
715    J=J+1
716    DO 717 T=1,IL
717    If(I(J,1),1,0,1,0) GO TO 304
718    DO 717 I=1,IL
719    K=K+1
720    717 I=(J,I),1,I,K
721    722 W=I
722    J=J+1
723    DO 724 T=1,IL
724    If(I(J,1),1,0,1,0) GO TO 304
725    DO 724 I=1,IL
726    K=K+1
727    724 I=(J,I),1,I,K
728    729 W=I
729    J=J+1
730    DO 731 T=1,IL
731    If(I(J,1),1,0,1,0) GO TO 304
732    DO 731 I=1,IL
733    K=K+1
734    731 I=(J,I),1,I,K
735    736 W=I
736    J=J+1
737    DO 738 T=1,IL
738    If(I(J,1),1,0,1,0) GO TO 304
739    DO 738 I=1,IL
740    K=K+1
741    738 I=(J,I),1,I,K
742    743 W=I
743    J=J+1
744    DO 745 T=1,IL
745    If(I(J,1),1,0,1,0) GO TO 304
746    DO 745 I=1,IL
747    K=K+1
748    745 I=(J,I),1,I,K
749    750 W=I
750    J=J+1
751    DO 752 T=1,IL
752    If(I(J,1),1,0,1,0) GO TO 304
753    DO 752 I=1,IL
754    K=K+1
755    752 I=(J,I),1,I,K
756    757 W=I
757    J=J+1
758    DO 759 T=1,IL
759    If(I(J,1),1,0,1,0) GO TO 304
760    DO 759 I=1,IL
761    K=K+1
762    759 I=(J,I),1,I,K
763    764 W=I
764    J=J+1
765    DO 766 T=1,IL
766    If(I(J,1),1,0,1,0) GO TO 304
767    DO 766 I=1,IL
768    K=K+1
769    766 I=(J,I),1,I,K
770    771 W=I
771    J=J+1
772    DO 773 T=1,IL
773    If(I(J,1),1,0,1,0) GO TO 304
774    DO 773 I=1,IL
775    K=K+1
776    773 I=(J,I),1,I,K
777    778 W=I
778    J=J+1
779    DO 780 T=1,IL
780    If(I(J,1),1,0,1,0) GO TO 304
781    DO 780 I=1,IL
782    K=K+1
783    780 I=(J,I),1,I,K
784    785 W=I
785    J=J+1
786    DO 787 T=1,IL
787    If(I(J,1),1,0,1,0) GO TO 304
788    DO 787 I=1,IL
789    K=K+1
790    787 I=(J,I),1,I,K
791    792 W=I
792    J=J+1
793    DO 794 T=1,IL
794    If(I(J,1),1,0,1,0) GO TO 304
795    DO 794 I=1,IL
796    K=K+1
797    794 I=(J,I),1,I,K
798    799 W=I
799    J=J+1
800    DO 801 T=1,IL
801    If(I(J,1),1,0,1,0) GO TO 304
802    DO 801 I=1,IL
803    K=K+1
804    801 I=(J,I),1,I,K
805    806 W=I
806    J=J+1
807    DO 808 T=1,IL
808    If(I(J,1),1,0,1,0) GO TO 304
809    DO 808 I=1,IL
810    K=K+1
811    808 I=(J,I),1,I,K
812    813 W=I
813    J=J+1
814    DO 815 T=1,IL
815    If(I(J,1),1,0,1,0) GO TO 304
816    DO 815 I=1,IL
817    K=K+1
818    815 I=(J,I),1,I,K
819    820 W=I
820    J=J+1
821    DO 822 T=1,IL
822    If(I(J,1),1,0,1,0) GO TO 304
823    DO 822 I=1,IL
824    K=K+1
825   
```

Fig. 5 An example of results

定義式からハードウェアにおける回路定数まで換算する一般的方法はまだ与えられていない。実際、RTL、DTL (TTL)などを用いしきい値素子をつくるとき、論理動作の調整は、抵抗変化や付加ダイオード数の変化で行なうことになり、結局真理値が必要となる。

本文の方法は、具体的にプログラムする段階では $m \leq 7$ とした。もし、 $m \geq 8$ となると k -asum. 判定が使えなくなり、サブルーチンを k -asum. 判定に作りかえる必要がある。このとき、設計基本方針には何ら変化はないけれども、実際の手数は極端に増し、この問題は k -asum. 判定におけるより正確な k を求めるというしきい値固有の問題⁹⁾ に還元される。

6. むすび

Fig. 1 の構成において関数 f が一般のブール関数とするならば、周期系列の長さ n に対してシフトレジスタの数 m はつねに $\lceil \log_2 n \rceil$ ($\lceil x \rceil$ は x に等しいか、 x より大の最小の整数を表わす) とする構成がつねに

可能である。しかしながら f の実際の実現はつねに簡単にできるとは言えない。

一方、ガロア体上の演算、すなわち ring sum 演算を用いる設計法は統一的になってきているが⁶⁾、これまた実際の f の実現は必ずしも容易であるとはいえない。

これらのこと考慮すれば f をしきい値関数として実現する方法は一つの実用さを有しているといえよう。

なお, f を ring sum 演算とする方法と f をしきい値関数とする方法を比較したとき, 必要とするシフトレジスタの数は両者どちらが有利であるとはいえない。すなわち, あるときは前者の方が少なく, あるときは後者の方が少ない。 f をしきい値関数とする順序回路の動作は, オートノマス動作といえども特徴化はほとんどなされていないため¹¹⁾前者と長短を論じることはまだ早計といえよう。

最後に、日頃よりご鞭撻いただく大阪大学田中幸吉教授、北橋忠宏博士、広島大学吉田典可教授、多大な協力をいただいた大阪大学永見均氏、プログラム作成に当たってくれた極東マック吉村正明氏に深謝する。なお、使用した計算機は TOSBAC 3400（広島大学計算センター）である。

参 考 文 献

- 1) 甘利: 自己組織しきい素子回路によるパターンの学習, 信学論(D), Vol. 55-D, No. 7, pp. 456~463 (1972)
 - 2) S. Yajima, T. Ibaraki and I. Kawano: On Autonomous Logic Nets of Threshold Elements, IEEE Trans. Vol. EC-17, p. 385 (1968)
 - 3) 石井, 木村: 与えられた状態列を実現するしきい素子回路網のテスト・シンセシス, 信学論(C), Vol. 51-C, No. 6, pp. 259~266 (1968)
 - 4) 石井, 宮崎: しきい素子回路網で実現できるサイクル集合の生成, 信学論(D), Vol. 55-D, No. 8, pp. 491~498 (1972)
 - 5) 永見, 阿江, 吉田: 任意周期系列のしきい値素子回路網による一実現, 信学会オートマトンと言語研究 AL 72-71 (1972)
 - 6) A. Gill: Linear Sequential Circuits-Analysis, Synthesis and Applications, McGraw-Hill Inc. (1966)
 - 7) S. T. Hu: Threshold Logic, University of California Press (1965)
 - 8) 福村: 最近のスイッチング回路網理論(その3) しきい値論理, 信学誌 Vol. 53, No. 9, pp. 1276~1284 (1970)

- 9) 佐藤, 北橋, 田中: しきい値関数判定に関する
一考察, 信学論(C), Vol. 54-C, No. 9, pp. 789
~795 (1971)
- 10) N. J. Nilsson: Problem-Solving Methods in
Artificial Intelligence, McGraw-Hill (1971)
- 11) 阿江, 永見: しきい値素子を用いるシフトレジ
スタ形順序回路, 信学会オートマトンと言語研
究 AL 73-38 (1973)
(昭和 49 年 1 月 8 日受付)
(昭和 49 年 5 月 31 日再受付)