

Title	順序回路の故障検査に関する研究
Author(s)	村上, 伸一
Citation	大阪大学, 1970, 博士論文
Version Type	VoR
URL	<a href="https://hdl.handle.net/11094/220">https://hdl.handle.net/11094/220</a>
rights	
Note	

*Osaka University Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/>

Osaka University

# 順序回路の故障検査に関する研究

1 9 7 0 年 1 月

村 上 伸 一

# 順序回路の故障検査に関する研究

1970年1月

村上伸一

## 内 容 梗 概

本論文は筆者が大阪大学大学院工学研究科（電子工学専攻）の学生として、尾崎研究室において行った順序回路の故障検査に関する研究をまとめたものである。

本論文全体は5章より成り立っている。第1章序論では本研究の目的ならびにこの分野での研究の現状について述べ、他の研究との比較において本研究の意義を明らかにしている。第2章では論文(1)、(2)として発表した内容を中心に、与えられた順序回路の故障検出を行うための検査系列の求め方の一方法について述べている。第3章では論文(3)、(4)の内容を中心に、与えられた機能を持つ順序回路を構成する際、はじめからその故障検査が容易に行えるようにその順序回路を設計しておく方法について述べている。第4章では発表論文(5)、(6)を中心に故障検出等を行う際しばしば問題となる定められた系列を発生させる方法について述べている。第5章結論では、本研究で得られた結果と今後に残された問題とをまとめている。

最近電子機器のモジュール化や集積回路の発達につれて回路の内部接続に触れることが困難となってきている。したがって論理回路の故障検査を行う際にも、内部に触れることなく、その入力出力の対応関係だけから故障の有無および故障の個所を指摘することが要求されるようになってきた。本研究では回路のある時刻における出力が、その時加えられた入力とその時刻における回路の状態とに基づいて決定されるような記憶をもった論理回路すなわち順序回路を対象とし、その故障検査について考察した。順序回路の故障検査を組合せ回路と記憶素子部分とに分割して行うのではなく、順序回路のままに入出力の対応関係から行うことは従来あまり行われていなかった。その主な理由は一般的な順序回路に対し、その故障検査を行うための検査系列を求める能率のよい手順が得られていないこと、またこれまでに知られている方法では求める故障検査系列の長さがかかり長くなり、実際の見地からはそ

れによる故障検査の実行が困難であること等が考えられる。本論文では実用的な長さの故障検査系列を得ることに主眼をおき、それを得るための方法について考察した。

第2章では、与えられた順序回路全体を一つの回路単位とみなし、計算機を用いてその順序回路だけを一意的に表現する入出力系列（故障検査系列）を直接求めることを考えた。このようにすると従来の方法すなわち順序回路を状態遷移に分割し、そのおのおのの状態遷移を一意的に表現する部分的な入出力系列を求めそれらをつなぎ合せた形で全体の検査系列を構成するという方法よりかなり短い検査系列を得ることが可能である。これは故障検査系列の各部分は相互に密接な関連を持ち、ただ単なるつなぎ合せ以上の情報を持つことに起因するからである。第2章前半の方法ではいわゆるバックトラック・プログラミングの手法を用いており、与えられた順序回路に対する最小長の故障検査系列を求めることが可能である。また第2章後半では比較的状态数の多い順序回路にも適用できる方法について述べている。

第3章では、与えられた順序回路に対してその故障検査系列を求めるのではなく、始めから与えられた機能を持った順序回路を構成するときその順序回路の故障検査系列が容易に求められるようにもとの順序回路を少し変形して構成しておくという方法について考察した。このようにしておくとその故障検査系列が短くなる一方、それを求める手順も組織的で、また故障箇所を指摘する故障診断も比較的容易に行えること等の特長を持たせることができる。またこの変更の際には余分に必要とするハードウェアは組合せ回路のみであり、記憶素子は余分に必要としないことも知られた。

第4章では、ある定められた系列を発生させる方法について考察した。これは論理回路の故障検査を入出力の対応関係から行うときなど、その加えるべき入力系列や比較の対象とするべき出力系列を作ることが必要であり、実際の見地から重要な問題である。ここではこの系列を発生させる一つの方法として周期性をもったいくつかの系列を作っておき、与えられた系列をこれ

らの周期系列を用いて合成するという方法を考えた。ここに述べた方法は任意の系列を発生できること、また他の系列発生器に比べ必要とする記憶素子の数が最少であることなどの特長をもつものである。また周期系列を発生させる部分とそれらから与えられた系列を合成する部分とを回路的に分離できるようにすることにより、簡単に他の系列を発生するように変更できること、同時に数種の系列を発生できること等の性質を持つようにできることも示された。

## 関連発表論文

- (1) 村上、樹下、尾崎、“入出力系列を満たす順序回路の構成法とその故障検査への応用について”、信学会、オートマトン研資(昭41-09)。
- (2) 村上、樹下、尾崎、“順序回路の故障検査系列の計算機による探索”、信学誌、(昭42-10)。
- (3) 村上、樹下、尾崎、“故障検査を考慮した順序機械の構成法”、信学会オートマトン・インホメーション研資(昭43-01)。
- (4) 村上、樹下、尾崎、“故障検査を考慮した順序機械の構成法”、信学論(C)(昭43-10)。
- (5) 村上、樹下、尾崎、“定められた系列を発生させる方法について”、昭41信学会全大。
- (6) 村上、樹下、尾崎、“系列発生器の構成に関する一考察”、信学論(C)(昭45-01)採録決定。

# 順序回路の故障検査に関する研究

## 目 次

第1章 序論 .....	1
第2章 故障検査系列の計算機による探索 .....	5
2.1. 序言 .....	5
2.2. 入出力系列を満たす最少状態数の順序回路の構成 .....	6
2.3. 順序回路の故障検査系列 .....	11
2.3.1. 最小長の故障検査系列 .....	12
2.3.2. 局所的最適化法による、順序回路の故障検査系列の構成...	14
2.3.3. 入出力系列の集合による、順序回路の故障検査系列の構成...	17
2.4. 結言 .....	21
第3章 故障検査を考慮した順序回路の構成法 .....	23
3.1. 序言 .....	23
3.2. 故障検査の容易な順序回路 .....	23
3.3. 順序回路の故障検査系列 .....	28
3.3.1. 故障に対する仮定 .....	28
3.3.2. $CC$ -型順序回路に対する故障検査系列の構成 .....	28
3.3.3. 与えられた順序回路の $CC$ -型順序回路への拡大 .....	30
3.3.4. 故障検査系列の長さとは付加するハードウェアについての 考察 .....	33
3.4. 順序回路の故障診断に関する考察 .....	34
3.4.1. 定義と仮定 .....	34
3.4.2. 故障診断の方法 .....	35



3. 4. 3. 条件Ⅱに対する考察 .....	37
3. 4. 4. 故障診断の容易な他の順序回路 .....	40
3. 5. 結言 .....	41
第4章 系列発生器の一構成法 .....	42
4. 1. 序言 .....	42
4. 2. 2値系列の合成 .....	42
4. 2. 1. 定義と記号 .....	42
4. 2. 2. 長さが2の冪乗である2値系列の合成 .....	44
4. 2. 3. 長さが2の冪乗でない2値系列の合成 .....	45
4. 3. 生成ベクトルの回路の実現 .....	48
4. 4. 系列発生器の特徴 .....	53
4. 5. 結言 .....	57
第5章 結論 .....	58
付 録 .....	59
謝 辞 .....	64
参 考 文 献 .....	65

## 第1章 序 論

最近の電子機器の発達はめざましく、その機能はより高度化され複雑化されてきた。そして機器使用者の立場から稼働率、信頼性に対する厳しい要求が出されるにつれ、その故障検査が重要な問題となっている。一方回路の小型化や、集積回路の応用に伴い回路の内部接続に触れることが困難となってきたため、故障検査の仕方としては従来のような各素子当りの検査ではなく、ある大きさの回路を単位とした入力、出力の対応関係によって検査する方式が望まれるようになってきた。本研究はデジタル計算機等を構成している論理回路のうち対象を順序回路に限りその故障検査を入出力関係によって行うための検査系列の求め方、その検査系列を得やすくするための順序回路の設計方針、ならびに検査系列を作り出す方式について考察したものである。

一般に、ある入力に対しある出力を出すシステムを考えると、その出力はシステムの状態とそのとき加えられた入力に基づいて決定されると考えることができる。このようなシステムのうち状態が一つしかないようなシステム、すなわちそのとき加えられた入力のみに基づいて出力が決定されるようなシステム（回路）を組合せ回路と言ひ、状態の数が有限であるような回路を順序回路と言ひ。通常組合せ回路と順序回路を合せて論理回路と呼んでいる。前者の組合せ回路は過去に加えられた入力が現在の出力に全く影響を与えないからその意味で記憶を持たない回路であると言える。したがって組合せ回路の検査では対象とする入力は可能な入力種だけであり、本質的にはそれらすべての入力に対し期待される出力が出るか否かを調べれば十分である。しかし一般にはいくつかの入力に対してその出力を調べ、正しい出力が出ればその回路は正常であると結論できる場合が普通で、すべての入力に対し検査する必要はない。したがって組合せ回路の検査に関する研究の大半は〔13、14、15〕、上のような性質を持つできるだけ少ない入力の部分集合

を求めることに主眼がおかれている。これに対し、後者の順序回路はその出力が過去の入力に依存するからこの意味で記憶を持った回路と言える。そして順序回路では新しい入力が増えられると回路はまた別の状態に移ると考え、これをその入力による順序回路の状態遷移と呼んでいる。順序回路は一般にこの状態に関する記憶部分と現在入力によって記憶の変更を決定する組合せ回路部分とから成っている。そこで従来は順序回路の故障検査は、この記憶部分と組合せ回路部分を分割しそれぞれ別個に検査する方法が多くとられてきた。しかしこのように順序回路を2部分に分割することは、すなわち回路の内部接続に触れることであり、入出力の対応だけからの検査を考えるとときには好ましい方式ではない。本研究では順序回路を分割することなく順序回路を一体としたままその入出力の対応関係から検査することを考える。順序回路では過去に加えられた入力が増える現在の出力に影響を与えるから、対象とする入力は入力の系列を考えねばならない。そして順序回路にある入力系列を加えたとき得られる出力系列がその順序回路に特有のものであり、他の順序回路ではその出力系列を出し得ないことが言えるとき、その入力系列と出力系列の対をその順序回路を一意的に表現する入出力系列であると言う。すなわち順序回路にこのような入力系列を加えたとき、それに対応する出力系列が得られるならその順序回路は予期された順序回路と決定できるので、順序回路の故障検査を入出力の対応関係で行う問題はこのような入出力系列を求めることに帰着される。順序回路の検査問題に対する解決法の能率の良さに関する基準としては、検査系列を求める手順の簡単さとその得られる系列の長さの二つが考えられる。

この入出力系列による順序回路の検査問題に関してはいくつかの研究〔1、5、6、7、8〕があるが Hennie〔6〕は順序回路の状態遷移に着目して、おのおのの状態が満足する特徴的な入力と出力の系列の対をみつけ出し、各状態遷移毎にその入出力系列を用いてその遷移を一意的に定める部分系列を求め、それらをつなぎ合せて全体の検査系列とした。この方法は原理的には

明確で分りのよいものであるが一般的な順序回路に対しては得られる検査系列が長くなるという欠点がある。

しかし一般に入出力系列の各部分は相互に密接な関連をもち各部分系列の Concatenation が持つ状態遷移に関する情報はその各部分系列がその状態遷移に関して持つ情報の単なる和とはならず、それ以上の情報を持つのが普通である。したがって逆に言うと Hennie の考えたような部分系列を単位とし、その和として得られる検査系列よりはるかに短い系列で全状態遷移を一意的に定めることができるのが普通である。すなわち短い検査系列を求めようとするなら系列を部分系列に分割することなく、系列全体としてその系列が持つ情報を計算せねばならない。

この観点から第2章では一つの入出力系列が持つ状態遷移に関する情報の量を計算し、それが与えられた順序回路のもつ状態遷移に関する情報の量に等しいか否かを判定することによって順序回路の全状態遷移を一意的に定める系列（故障検査系列）を得ることとした。この方法を用いると与えられた順序回路に対する最小長の検査系列を得ることができる。

しかしすでに与えられた順序回路に対しその検査系列を求めることは手順が繁雑となり、また短い検査系列を得ることは本質的に困難な場合が多い。そこで順序回路の検査に関する他の解決法として、順序回路を構成するときその故障検査系列がたやすく得られるよう、また短い検査系列が得られるよう設計の段階で考慮しておくという方法が考えられる。

この立場からの順序回路の故障検査方法として Kohaviら<sup>[19]</sup> は状態の識別を容易にするため順序回路に余分の出力端子を設け、その出力値は各状態が互に区別できるようあらたに定めた。この方法によって比較的短い系列で状態が識別できその結果短い検査系列を得ることができたが、検査系列全体を構成する際の部分系列のつなぎ方は組織的でなく、検査系列を求める手順が簡単であるとは言い難い。

第3章ではやはりこの立場から検査系列を求めることを目指したがここで

は短い検査系列を得ると同時に検査系列の構成手順を組織的かつ簡単にするため Kohavi らとは逆にあらたな入力記号を導入した。そして順序回路の通常の動作においてはこの入力を使わず検査時のみにそれを用いて検査を行うことにする。これによりある程度上の二つの要求を満足する故障検査系列が得られ、さらにこの方法によると故障の有無を調べる故障検出だけでなく、故障箇所を指摘する故障診断も部分的に行えることが知られた。

入出力の対応関係で故障検査を行ったり、順序回路を駆動したり、またある標準となるデジタル的なパターンを作ったりするとき、定められた系列を作ることがしばしば必要となる。この問題に対し従来は求める系列全体を磁気テープ、紙テープあるいはシフトレジスタ<sup>[21, 22]</sup>等にたくわえておき順にシフトアウトして求める方法や、オートノマスな順序回路で求める系列を発生させる方法がとられてきた。前者の方法は必要な系列をシフトレジスタや記憶装置につめかえることにより各種の系列を発生させることができるが、系列の長さ分だけのレジスタが必要となりあまり経済的な方法ではない。また後者の方法は系列長に比べ記憶素子(レジスタ)は少なくすむが、一つの系列しか発生させることができず実際の利用の立場からみて融通性のある方法ではない。

Perlmen<sup>[23]</sup>や Cohn ら<sup>[31]</sup>は後者の立場から与えられた系列を発生させる方法を考えたが、与えられた系列をある長さを単位として観察し、その次につづく系列の値をその前の部分系列で一意的に定められるよう組合せ回路を定めた。この方法によると、得られる系列発生器の複雑さは発生させる系列の形に大きく依存し、それに要する記憶素子の数も大きく変わる。またその組合せ回路を求める操作は簡単とは言い難い。

第4章では、やはり後者の立場から系列発生器を構成することを考えたがここで得られるオートノマス順序回路は、任意の系列を発生できること、他の系列発生器に比べ必要とする記憶素子の数が最少であること、別の系列を発生するようたやすく変更できること、また同時に数種の系列を発生できること等の特長をもつものである。

## 第2章 順序回路の故障検査系列の計算機による探索

### 2.1 序 言

入出力系列による順序回路の故障診断の方法としては、順序回路の識別問題への応用としてすでにいくつかの方法〔2, 3, 4, 5〕が知られている。しかし、これらの方法では診断のための検査系列を求める手順がかなり多くなり、きわめて簡単な場合にしか有効ではなく、また診断に当たってあらかじめ故障の予測を行わなければならないなどの困難な点がある。そこでここではこれらの困難から離れて比較的簡単に検査系列を求めると言う立場から、故障個所を指摘する“故障診断の問題”ではなく、単に順序回路が正常に動作しているかどうかを決める順序回路の“故障検出の問題”について考察することにする。この方法として、Hennie〔6〕が状態識別の方法から検査系列を求める一つの方法を与えているが、Hennieの方法では求められる検査系列が長くなるという欠点がある。

この章では、それほど状態数が多い場合\*を対象にし、比較的短い検査系列を求める方法について考察するが、系列中の各個所はその系列の他のすべての個所とその形および相互の位置関係によって強く結びつき全体の系列の特徴を構成するので、短い検査系列を得ようとする場合、つねに系列全体を単位にして考えねばならない。また系列のすべての部分間の相互関係を調べることは手数の急激な増加を招く。そこでここでは計算機により、系列全体として与えられた順序回路を特徴づける入出力系列を探索することにする。このとき、ある一つの与えられた入出力系列がどのような順序回路を特徴づける系列であるかを知ることが必要であり、また与えられた入出力系列を満足する最少状態数の順序回路を構成する問題自身、順序回路の設計問題としてしばしば起る問題であるので、はじめに、与えられた入出力系列を満たす最少状態数の順序回路を求める方法について述べ、つぎにこの方法を応

\* たとえばパッケージ程度の大きさ

用して順序回路の故障検査系列を求める方法について述べることにする。

## 2.2 与えられた入出力系列を満たす最少状態数の順序回路の構成

与えられた有限長の入出力系列を満足する順序回路を求めることは、もし状態数に制限をおかなければ、順次入力を受け入れるたびに新たな状態をつけ加えて行くことによって求めることができる。すなわち、長さ  $l$  の入出力系列を満足する高々  $l + 1$  個の状態をもつ順序回路を作ることができる。したがってその中で最少状態数のものを見つけようとするならば、上のようにして求められた状態遷移が完全には定義されていない (incompletely specified の) 順序回路の状態数の最小化を行えばよいことになる。状態遷移が完全には定義されていない順序回路の状態数最小化の問題についてはすでにいくつかの方法が知られているが<sup>[16, 17, 18]</sup> それ自身としても必ずしも十分に能率のよい方法とはいえず、特にこの場合のように、長さに比例した状態数から出発する場合には、比較的簡単な問題に対しても、簡単化を行うことはそれほど容易ではない。したがって、与えられた入出力系列を満たす最少状態数の順序回路を構成する問題を、順序回路の簡単化の問題に帰着させることは、必ずしも能率の良い方法であるとは言えないので、ここでは別の観点からこの問題を考えることにする。

Rabin, Scott<sup>[8]</sup> が示しているように、最少状態数の有限オートマトンを求めることは、状態遷移関数 (および出力関数) に関して定義される右不変な同値関係による同値類を求めることである。すなわち、ある入力系列が与えられれば、状態遷移関数に関して矛盾がないようにその系列のはじめの部分で順次同値類に分類してゆくことであると考えられる。ここではこの考え方に基づき、さらに backtrack programming<sup>[9]</sup> の手法を応用する方法について考察する。

以下議論を簡単にするために、Mealy 型の順序回路について考え、2入力 2出力の場合について述べる。Moore 型の場合および多入出力の順序回

路も同様に取り扱いすることができる。

分かりやすくするために、まず簡単な例について説明する。図 2-1 の長

				$S_2$	
				$S_1$	
		$S_2$	$S_2$	$S_2$	$S_2$
(現在)状態	$S_1$	$S_1$	$S_1$	$S_1$	$S_1$
入力系列	0	0	1	0	1
出力系列	0	1	0	1	1
(遷移)状態	$S_1$	$S_1$	$S_1$	$S_1$	
	$S_2$	$S_2$	$S_2$	$S_2$	
			$S_1$		
			$S_2$		

図 2-1 長さ 5 の入出力系列

さ 5 の入出力系列を満たす 2 状態の順序回路が存在するかどうかを考えてみる。まず初期状態を  $S_1$  とし、入力 0 によって状態  $S_1$  に遷移する（出力は 0）とすれば、第 2 列においては状態  $S_1$  で入力 0 のとき出力 1 であるから出力関数に関して矛盾する。したがって第 1 列における遷移状態は新たな状態  $S_2$  でなければならない。つぎに  $S_2$  で入力 0 のとき  $S_1$  に遷移するとし第 3 列において  $S_1$  で入力 1 のとき、もしそれが  $S_1$  に遷移するとすれば第 4 列において第 1 列の遷移関数と矛盾するから、 $S_1$  からの入力 1 での遷移状態は  $S_2$  でなければならない。そうすると、第 4 列は第 2 列の状態遷移関数および出力関数から  $S_1$  に遷移しなければならない。しかしこれは第 5 列において、状態  $S_1$  で入力 1 のとき出力 1 となり第 3 列と矛盾する。これは第 2 列において遷移状態を  $S_1$  にしたこと起因するから、これを  $S_2$  としてみる。そうすると第 3 列において、状態  $S_2$  で入力 1 のとき遷移状態を  $S_1$  とすれば、第 4 列において矛盾するからこれは  $S_2$  でなければならない。第 4 列は第 2 列の条件より遷移状態は  $S_2$  となる。しかしこれは第 5 列において出力関数が矛盾する。したがってこの入出力系列を満たす 2 状態の順序回



略は存在しないことになる。

もし3状態まで許されるとすれば、上と同じようにして図2-2なる解が

				$S_3$	$S_2$
		$S_2$		$S_2$	$S_1$
(現在)状態	$S_1$	$S_1$	$S_1$	$S_1$	$S_1$
入力系列	0	0	1	0	1
出力系列	0	1	0	1	1
(遷移)状態	$S_1$	$S_1$	$S_1$	$S_1$	$S_1$
	$S_2$		$S_2$	$S_1$	
			$S_3$	$S_2$	

図2-2 長さ5の入出力系列と  
それを満たす順序回路

存在し、これが与えられた入出力系列を満足する最少状態数の順序回路の一つである。

上に述べたことを一般化して、結局つぎのようにして入出力系列を満たす最少状態数の順序回路を求めることができる(図2-3)。

説明を簡単にするためにつぎのように記号を定めておく。

- $l$  : 与えられた系列の長さ。
- $n$  : 順序回路の許容される状態数。
- $\lambda_i$  : 与えられた系列の  $i$  番目の入力記号。
- $\rho_i$  : 与えられた系列の  $i$  番目の出力記号。
- $s$  : 順序回路の状態を表わす記号。
- $S$  : 順序回路の状態  $s$  をたくわえるレジスタ。

$T(s, \lambda_i)$  : 状態遷移関数

$O(s, \lambda_i)$  : 出力関数

P D S S : 順序回路の状態を記憶するあなぐら記憶

P D S I : インデクスレジスタ I の内容を記憶するあなぐら記憶

R : 系列の途中において、そこまでの系列が必要とする状態数を示すレジスタで、重複した順序回路を避けるための検査に用いられる。

i : 着目している個所が系列の何番目の記号であるかを示す添字。

I : iをたくわえるインデクスレジスタ。

T.T. : 構成しつつある順序回路の状態遷移表。

操作 A (図 2-3 において)

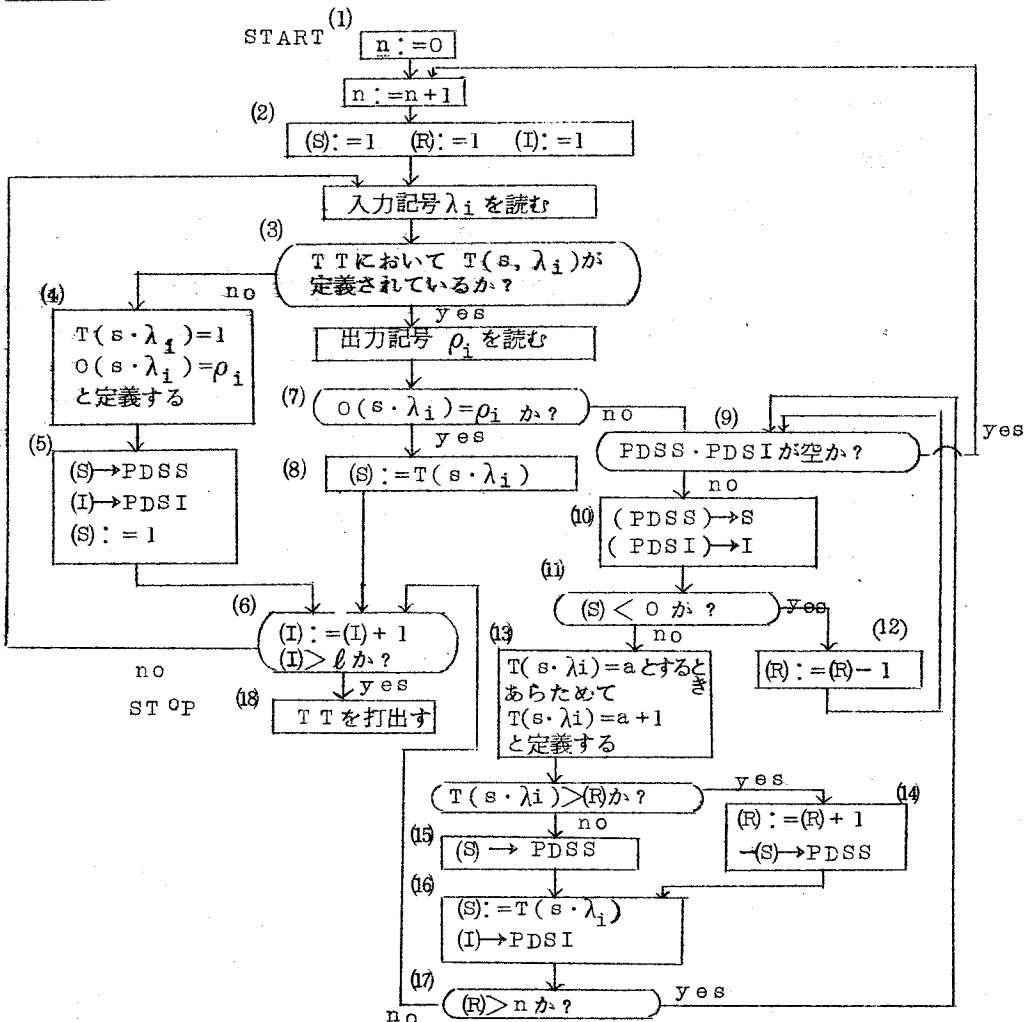


図 2-3 操作 A フローチャート

- (1) 状態数  $n = 1^*$  から調べる。
- (2) 状態記号として正整数を用いる。初期状態を 1 とする。R はこのとき 1 となり、また系列の一番最初の記号を読むので  $i = 1$ 。
- (3) すでにこれまでに作られた T.T. において、現在状態のその入力についての状態遷移が定義されているかどうかを調べる。
- (4) 定義されていないならば、その状態のその入力に対する遷移状態を 1、そのときの出力を  $\rho_i$  と定義する。
- (5) このときまだ遷移状態を変える可能性が残るので、入出力系列のこの位置とこの現在状態を記憶する。
- (6) 要求された長さ  $l$  の系列のおわりまですんだかどうか調べる。
- (7) 現在状態  $s$  の入力  $\lambda_i$  に対する遷移状態がすでに定義されていたのでそのときの出力と今度の出力が一致するかどうか調べる。
- (8) 出力が一致すれば、それまでに定義された T.T. が与えられた系列の始めから  $i$  番目までの入出力系列を満たす順序回路であることになる。このときの遷移状態をつぎの現在状態とする。
- (9) 現在状態  $s$  の入力  $\lambda_i$  に対する出力は以前の要求から  $O(s, \lambda_i)$  であり、これが今度の出力  $\rho_i$  と一致しない。すなわち、これまでに定義された T.T. ではこの  $i$  番目の入出力対を満たすことができない。そこで遷移状態を変更しうる可能性のある最も近い位置にさかのぼる。変更可能な個所がなければ状態数  $n$  の状態遷移表ではこの入出力系列を満たすことができないので  $n$  を 1 だけ増す。\* \*
- (10) 変更可能な個所にもどる。

---

\* あらかじめその系列が必要とする最少状態数が分かっている場合はその数から始めればよい。

\*\* この場合  $n$  を最小解のありそうな数まで増し、解があればまた下げるようにして両側からはさんで求めることが考えられるが、そのための基準も定めにくく、また最小解も比較的小さな所にあることが多いので必ずしもあまり有効とは思われない。

(11) (9) < 0 のとき、この変更可能な個所はたとえその遷移状態を変更しても、ただ状態の名を変更したにすぎない。(14参照)。

(12) つぎの変更可能な個所にもどるが、このときそれまでに必要とする状態数の最大値は1だけ下がる。

(13) 遷移状態を以前の遷移状態のつぎの状態に変更する。

(14) 変更先が初めての状態であるときで、これ以後この遷移先をさらに別の状態に変更するようなことが起れば、それは単なる名前の変更にすぎない。それを示すため、初めての状態への遷移を必要とする位置の現在状態を  $s$  として記憶する(11参照)。また初めての状態が出たので  $R$  は1だけ増す。

(15) まだこの遷移先を変更することが可能なのでこの現在状態を記憶する。

(16) 変更可能な位置を記憶し、このときの遷移状態をつぎの現在状態とする。

(17) それまでに必要とした状態数が状態数の上限  $n$  を越えたかどうかを調べる。越えていればつぎの変更可能な個所にもどる。

(18) ここで得られる T.T. がその入出力系列を満たす最少状態数の順序回路である。

### 2.3 順序回路の故障検査系列

前節では与えられた入出力系列を満たす状態数最少の順序回路を求める方法について述べた。ここではこの方法を用いて、順序回路に対する故障検査系列を求める方法について考察する。

ある入出力系列  $L = \{L_i, L_o\}$  が与えられたとき、もしこれを満たす最少状態数  $n$  の順序回路  $M$  がただ一つしか存在しないとすれば、この順序回路  $M$  を  $L$  によって表わすことができる。すなわち、状態数が  $n$  より多くない未知の順序回路に入力系列  $L_i$  を加えたとき、出力系列  $L_o$  が得られるならば、この未知の順序回路は  $M$  でなければならないことになる。このように与えられた順序回路  $M$  に対し  $M$  を唯一に定める入出力系列  $L = \{L_i, L_o\}$  が求まれば、

入力系列  $L_i$  に対しては、 $M$  が正常に動作する限りその出力系列は  $L_0$  であり、もし出力系列が  $L_0$  と異なっていれば、順序回路  $M$  は故障であると言える。しかし  $M$  が故障になったときでも、もとのものより状態数が増した形の故障では、入力系列  $L_i$  に対して出力系列は  $L_0$  であることもあり得る。そこで故障によって状態数が増えないと仮定すれば、順序回路を一意的に表わす入出力系列  $L = \{L_i, L_0\}$  が  $M$  の故障検査系列となる。なお検査の始めにおいて、与えられた順序回路をある特定の初期状態にするために、最終的な検査系列には特別の入力系列をつけ加えることが必要である。<sup>\*</sup> このことにより以下では強連結な順序回路のみを考える。そこで結局、故障によって状態数が増加しないと仮定すれば、強連結な順序回路に対する故障検査系列を求めることは、与えられた順序回路を一意的に表現する入出力系列を求めることになる。

### 2.3.1 最小長の故障検査系列の探索

与えられた順序回路が completely specified であり、状態数が  $n$ 、入出力記号が 0、1 であるとすれば、その状態遷移図において  $2n$  個の状態遷移の枝を含むから、この順序回路を表現する入出力系列の長さは少なくとも  $2n$  以上でなければならないが、つぎに示すように  $2n$  ではそれを一意的に表現する入出力系列が存在しないことが容易に分る。

[補題 2-1] 状態数  $n (> 1)$  の順序回路を一意的に表現する入出力系列の長さは少なくとも  $mn + 1$  以上である。ただし、与えられた順序回路は completely specified であるとし、 $m$  は入力記号の数である。

(証明) 考える順序回路は completely specified としているから状態遷移の枝は全部で  $mn$  個ある。一つの入力系列ですべての状態遷移を規定するためにはその系列は  $mn$  個のすべての状態遷移の枝を含まねばならない。したがって長さ  $mn$  の入出力系列ですべての状態遷移を規定しようとするなら、各状態遷移の枝は頂度 1 回ずつその系列に現われることになる。

---

\* この付加する系列についてはここではふれない。文献(6)参照。

しかるに、このときその系列の第  $m n$  番目の状態遷移による遷移先の状態は任意の状態に選び得るから、結局長さ  $m n$  の系列ですべての状態遷移を一意的に定めることはできない。(証明終り)

補題より、求める入出力系列の長さの下限は  $m n + 1$  以上であるから、これと前節の操作 A を合わせてつぎのようにして順序回路 M を一意的に表現する最小長の入出力系列を求めることができる。

### 操作 B

- (1) 補題 2-1 より長さ  $m n + 1$  の系列から調べれば十分であるので、 $l := m n$  とし(2)へ。
- (2) 長さ  $l$  のどのような系列でも与えられた順序回路を一意的に表現できないから  $l := l + 1$  とし(3)へ。
- (3) 長さ  $l$  のすべての可能な入力系列について(4)以下の判定を行ったかどうか調べる。すべてについて済んでいれば(2)へ、まだ済んでいない長さ  $l$  の系列があれば、その系列について(4)以下の判定を行う。
- (4) その系列で与えられた順序回路のすべての状態遷移の枝を通るかどうかを調べる。通らなければ(3)へ。すべての枝を通れば(5)へ。
- (5) その入力系列に対する与えられた順序回路の出力系列を作る。(6)へ。
- (6) (5)で作られた入出力系列を満たす順序回路の総数を求める。(操作 A の図 2-3 において STOP の代わりに(9)へとばせる方法で求めることができる)。(7)へ。
- (7) (6)で求めた順序回路の総数が 1、すなわち与えられた順序回路のみであるなら、この入出力系列が与えられた順序回路を一意的に表現する求める系列である。総数が 2 以上であるなら(3)へ。

〔例 2-1〕 この方法を表 2-1 の順序回路に適用すれば、この回路を

表 2-1 順序回路  $M_{21}$

状態 \ 入力	0	1
$S_1$	$S_2(1)$	$S_1(0)$
$S_2$	$S_1(0)$	$S_3(1)$
$S_3$	$S_0(0)$	$S_1(0)$

一意的に表現する最小長系列の一つ\*として長さ10の入出力系列

入力系列: 0001000110

出力系列: 1011000001

を得た。なおこの順序回路は文献〔6〕〔7〕のと同じ例題であり、文献〔6〕ではこの回路を一意的に定めるのに長さ12を要しており、また文献〔7〕においては長さ16を要している。

### 2.3.2. 局所的最適化法による順序回路の故障検査系列の構成

2.3.1.に述べた与えられた順序回路を一意的に表現する長さ最小の系列を求めることは、その方法から分るように与えられた状態図を満たす可能な入出力系列すべてについてその順序回路を一意的に表現するかどうかを調べることになる。したがって状態数が多くなればその順序回路を一意的に表現する系列も当然長くなり、可能なすべての入出力系列についてその判定を行うことは実際的には不可能である。

そこでここでは、すべての可能な入出力系列について判定を行い与えられた順序回路を一意的に表現する系列を得る代わりに、計算機で探索が実行可能な程度(主に時間的制約による)の長さの系列を単位とし、その単位の系列のおのおのはその系列を満たす順序回路の数が最も少ないような系列を選び、それらをつなぎ合せて与えられた順序回路を一意的に表現する入出力系

\* 別にあと一つ長さ10の系列が存在している。

列を得るといった方法について考察する。この方法は図2-4に示されているが、ここで記号をつぎのように定めておく。

$l_0$  : 単位系列の長さ。

$\alpha_k$  : 長さ  $l_0$  の  $k$  番目の単位系列。

$\prod_{k=1}^i \alpha_k$  : 単位系列  $\alpha_k$  の  $1 \leq k \leq i$  の  $i$  個をつなぎ合せた系列

$\alpha_1 \cdot \alpha_2 \cdots \alpha_k$

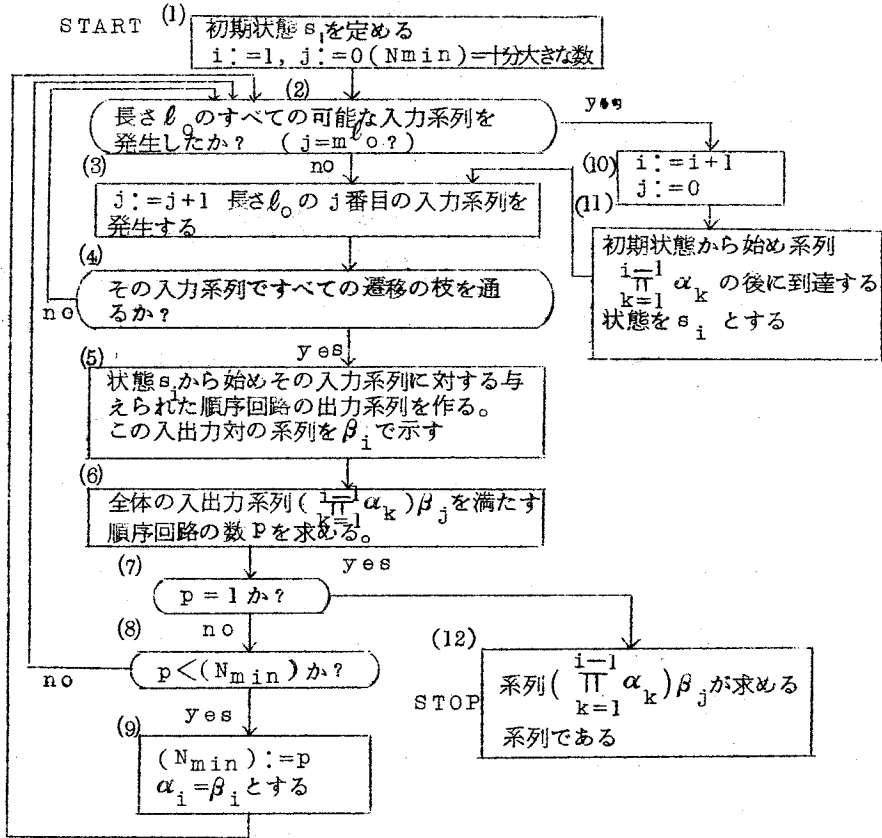


図2-4 操作Cフロチャート

操作C (図2-4において)

(1) 1番目の単位系列を表わすため、 $i := 1$ とする。また長さ  $l_0$  のすべての可能な入力系列を作りそれらに適当に順序をつけこの順序を  $j$  で示す。まず  $j := 0$  とする。



(2)  $m$ 種の入力記号の  $l_0$  個の可能な組合せ  $m^{l_0}$  個のすべての入力系列について調べたかどうか判定する。

(3) つぎの入力系列を発生する。

(4) 与えられた順序回路の状態数が多く、 $s_i$ からどのような長さ  $l_0$  の単位系列を選んでもすべての遷移の枝を通ることができない場合がある。このときには、より多くの枝を通る系列が望ましい系列であるという基準で判定する。

(5) それまでの系列で順序回路は状態  $s_i$  に達しているので、その状態にこの入力系列を加えたときの出力系列を作り、この入出力系列を  $\beta_j$  で示す。

(6) それまでの系列  $\prod_{k=1}^{i-1} \alpha_k$  に(5)で作った入出力系列  $\beta_j$  をつないだ全体の系列を満たす順序回路の総数を求める。

(7) この系列を満たす順序回路が唯一に定まるかどうかを調べる。

(8) この系列を満たす順序回路の数がいままでのどの系列よりも少ないかどうか調べる。多いときにはこの  $\beta_j$  はそれまでの  $N_{\min}$  を与える系列より結果的に長い系列を作ることになると考えられるから採用せず、つぎの系列を発生する。

(9) この系列がいままでのうちで最も短い系列を作る可能性があるから、この系列を  $i$  番目の単位系列として採用する。しかしさらにもっと短いものがないかを調べるため、つぎの系列を発生する。

(10) その単位系列については、長さ  $l_0$  のすべての系列を調べたのでつぎの単位系列を作る操作に移る。

(11) これまでの系列により到達する最終状態を求めておく。

(12) この系列が与えられた順序回路を一意的に表現する求める系列である。

〔例 2-2〕 この方法を表 2-2 に示す順序回路に適用すると、この順

表 2-2 順序回路  $M_{22}$

状態 \ 入力	0	1
$S_1$	$S_1(1)$	$S_2(0)$
$S_2$	$S_1(1)$	$S_3(0)$
$S_3$	$S_4(0)$	$S_1(0)$
$S_4$	$S_3(0)$	$S_4(0)$

序回路を一意的に表現する系列としてつぎの長さ 22 の系列

入力系列：1 0 1 0 0 1 0 1 1 0 0 0 1 1 0 1 1 0 1 0 1 0

出力系列：0 1 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1

を得た。なおこの順序回路も文献〔6〕〔7〕の例題であり、文献〔6〕では系列長 146 を要しており、また文献〔7〕では系列長 52 を要している。

### 2.3.3. 入出力系列の集合による順序回路の故障検査系列の構成

ここでは順序回路の故障検査系列を求める他の方法について考察する。与えられた順序回路を一意的に表現する入出力系列を求めるに当たり、系列を 2 つの部分に分けて考える。すなわち前半の系列は、それを作る手順が容易であり、しかも系列を満たす順序回路の数がかなり少ないであろうような系列を選ぶ。そして後半の系列は、前半の系列を満たす順序回路のうち、与えられた順序回路のみに満たされる系列を選ぶ。この後半の系列は 2.3.2 に述べた方法により構成できるので、ここでは前半の系列の構成について考える。

与えられた順序回路を一意的に表現する一つの方法として、その順序回路を特徴づけるいくつかの入出力系列の集合を用いるという方法がある〔10〕。すなわち、順序回路の状態遷移が与えられたとき、これがある特定の入出力系列を含むかどうかを調べることにより（もしある系列を含めば、+印、含まなければ

一印で表わす)、入出力系列に関する状態表を作る方法である。この入出力系列に関する状態表において、+印で示される入出力系列をすべて含み、-印で示される系列をすべて含まないということが一つの系列によって表現できれば、その系列は与えられた順序回路を一意的に表現する入出力系列ということになる。このような方法で故障検査系列を求めるとき、+印の系列を含むということは簡単に表現できるが、-印の系列を含まないことを保証するのは簡単ではない。しかし、-印の系列の大半も、+印の系列をすべて含ませたときは、含まれないと言える場合が多い。そこで結局、文献〔10〕における+印の系列をすべて含む系列を前半の系列として選べば、-印の系列中の含まれないと保証できないものについての分だけの余分な順序回路が含まれるにすぎない。ここではこのことに基づいて前半の系列を作るが、各特徴を示す系列の集合を作りやすいものとするため、それをH系列とD系列および遷移を行う入出力対の3つの結合で作ることにする。なおこの方法では必ずしも全検査系列長は短くならないが、かなり状態数の多い順序回路に対しても適用可能であると思われる。

まずH系列およびD系列をつぎのように定義する。

〔定義2-1〕 与えられた順序回路Mについてある入出力系列を考えたとき、その最終状態 $s$ がMにおいて一意的に定まるなら、その入出力系列を状態 $s$ へのH系列という。またある入出力系列について、その系列の初期状態 $s$ がMにおいて一意的に定まるなら、その入出力系列を状態 $s$ のD系列という。

順序回路の各状態が簡単化されているときには、各状態に対してH系列は常に存在するが、D系列は常に存在するとは限らない。

このとき前半の系列をつぎのようにして作る。

#### 操作D

(1) 順序回路のすべての状態について、なるべく短いそれぞれの状態へのH系列およびそれぞれの状態のD系列を作る。D系列の存在しない状態に対

しては、その系列を満たす初期状態の集合にその状態と一緒に含まれる状態の数が最も少ないような系列をD系列の代りに用いる。

(2)  $n$ 行 $m$ 列 ( $n$ :状態数、 $m$ :入力記号種の数)の表を作る。各行は各状態へのH系列のおののにおの、各列は各状態からの1回の遷移に対する入出力対のおののにおのに対応する。

(3) 状態遷移表に対応し、状態を示す各行において、その状態からの1回の遷移に対する入出力対の列の位置に+印を記入し、その遷移後の状態 $s$ をその列の名とする。

(4) 各列について、状態 $s$ の列に状態 $s$ のD系列あるいはその代りの系列を記入する。

(5) 表にしるされた各+印について、その+印のある行のH系列とその+印の状態遷移に対する入出力対、その列のD系列あるいはその代りの系列をこの順につなぐ。

(6) (5)で作られた $m$ 列の系列を、そのおののの系列の最終状態とつぎの系列の初期状態とが一致するようにしてつなぐ。\*

(6)で得られたのが前半の系列である。

〔例2-3〕 以上の操作Dを表2-3に示す順序回路に適用すると、各状態に対するH系列、遷移の入出力対、D系列とそれに対する(2)~(4)で得られる入出力系列に関する状態表は表2-4のようになる。たとえば入力00に対し出力10を出すならその最終状態は状態 $S_1$ でなければならない。したがって状態 $S_1$ へのH系列は  $\begin{matrix} 0 \\ 1 \end{matrix} / \begin{matrix} 0 \\ 0 \end{matrix}$  であり、またある状態に入力01を加え出力10を得るならその状態は状態 $S_3$ であることになる。すなわち、系列  $\begin{matrix} 0 \\ 1 \end{matrix} / \begin{matrix} 1 \\ 0 \end{matrix}$  は状態 $S_3$ のD系列となる。そして状態 $S_3$ からは入力0で出力0を出し状態 $S_3$ へ遷移するので表2-4の1行1列目に+印が記入される。

---

\* この部分系列のつなぎ方によって前半の系列の長さはかなり変化するが、その長さが最小の系列はブランチ・アンド・バウンド法を用いれば比較的簡単に得られる。



他の+印についても同様である。つぎに表の各+印につき(5)のようにして部分系列を作り、これらをつなぎ合わせると前半の系列として長さ50の系列

入力系列：0 1 1 0 1 1 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 1 0 1 0

1 0 1 0 0 1 0 1 1 0 0 0 0 1 1 0 1 1 1 0 1 0 0 1 0

出力系列：0 0 0 1 0 0 0 0 1 1 0 0 1 0 0 1 1 0 0 0 0 1 1 1 0

0 0 1 1 0 1 0 0 0 1 1 0 0 0 0 1 0 0 1 0 0 0 1 1 1

を得た。ただしこの系列においては(5)で得た各系列をつなぐとき、同じ部分は重ねて全系列長が短くなるようにした。なおこの例では、この系列を満たす順序回路は表2-3の回路のみであり、この回路を一意的に表現するには後半の系列は必要としない。

#### 2.4. 結 言

与えられた入出力系列を満たす最少状態数の順序回路の構成法を利用して順序回路の故障検査系列を探索する方法について考察した。故障検査系列の構成に当たっては順序回路の内部状態数の増加につれて三つの方法を考えた。なお2.3.2および2.3.3に述べた方法についてはさらにいくらか変更することが考えられる。たとえば2.3.2の方法については、単位系列を順次つないでゆくのに前の単位系列の前半分だけを採用し、そのつづきから次の単位系列を採すということにすれば系列長はさらに短くなると思われる。また2.3.2に述べた方法は、与えられた順序回路を特徴づけるいくつかの系列によって前半の系列を作り、これによって区別できない順序回路を後半の系列で分離する方法であるから、ここに述べた方法以外の方法によっても前半の系列を作ることができる。しかし、ここに述べた方法で求められる前半の系列は、多くの場合それだけで順序回路を一意的に表現しており、かなり能率のよい方法だと思われる。この方法の欠点は、故障検査系列を求める各段階において、でき上がった系列が与えられた順序回路を一意的に表現しているかどうかを判定せねばならないことであり、これが計算時間の大部分を占めている。

したがってこの判定なしに、あらかじめある系列の集合を含むことが順序回路を一意的に表現するための十分条件であることが求めればよいが、十分条件の制限をゆるくすると Hennie のようにかなり長くなる恐れがある。一般に、最小長の検査系列を求めることは、ある系列または系列の集合が順序回路を一意的に表現するための必要十分条件を求めることになるが、これらについてはまだ多くの問題が残されている。

## 第3章 故障検査を考慮した順序回路の構成法

### 3.1 序 言

第2章でみてきたように、与えられた順序回路に対しその回路を一意的に表現する入出力系列（故障検査系列）を求めることは複雑で、今までのところ短い検査系列を得る簡単な手順は得られていない。この順序回路の故障検査問題に対する他の解決方法として、前もって順序回路を設計するとき故障検査系列が得やすいようにもとの順序回路を変更しておくという方法がある。この一方法として Kohavi ら〔19〕は冗長な出力記号を新たに付加することを考えたが、ここでは特殊な状態遷移を行って入力変数を利用して比較的簡単に、かつ組織的に短い検査系列を作る方法について考察する。

この章の前半では故障検査の容易な順序回路と、それに対する故障検査系列の構成法ならびに与えられた順序回路をその故障検査しやすい順序回路に変更する手順について述べ、後半では前半に述べた方法によって作られた順序回路に対する故障診断の方法について考察する。

### 3.2 故障検査の容易な順序回路

順序回路の故障検査を入力、出力の対応関係から行いとき、つぎのような2つの入力系列があれば都合がよい。すなわち、一つは順序回路にその入力系列を加えたときそれに対する出力系列を見ることによってその入力系列を加えた後の順序回路の内部状態（最終状態）が一意的に決定できるような入力系列であり、もう一つはその入力系列を加えたとき、出力系列を見ることによってその入力系列を加える前の順序回路の内部状態（初期状態）が一意的に決定できるような入力系列である。なぜなら順序回路に前者の入力系列を加えその最終状態  $S_i$  を知り、つぎに状態遷移を検査したい入力  $I$  を加えその出力  $O$  を知り、さらにそのつぎに後者の入力系列を加えそのときの初期状態  $S_j$  を知れば、状態  $S_i$  からは入力  $I$  によって出力  $O$  を出して状態  $S_j$



へ遷移するということが分るからである。したがってこれらの入力系列（前者を homing sequence といい、後者を distinguishing sequence という）がたやすく求められるならその順序回路の故障検査は容易に行えることになる。ここでは、順序回路は Mealy 型であるとし、状態数を  $n$ 、内部状態を  $S_i$  ( $i=1, 2, \dots, n$ )、入力記号を  $I$ 、状態遷移関数を  $T(S_i, I)$ 、出力関数を  $O(S_i, I)$  で表わすことにする。このとき上の二つの系列が容易に求められる順序回路としてつぎの性質 (I), (II) を持った  $n$  状態の順序回路を考える。

性質 ある入力  $I$  に対して、

$$(I) \quad T(S_i, I) = S_{i+1}, \quad O(S_i, I) = "0" \quad (i=1, 2, \dots, n-1)$$

$$(II) \quad T(S_n, I) = S_1, \quad O(S_n, I) = "1"$$

ただし出力記号 "0", "1" は互いに異なっていればどのようなものでもよい。

いかえれば上の性質 (I), (II) を持つ順序回路は、ある同一の入力  $I$  によりすべての  $n$  個の状態を巡回的に遷移し、ある 1 つの状態遷移のときだけ異なった出力を出すような状態遷移をする順序回路である。(図 3-1 参照)

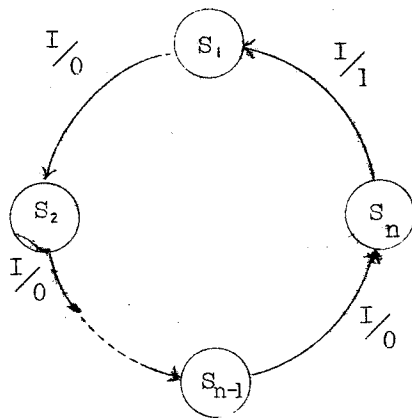


図 3-1 性質 (I), (II) をもつ順序回路の状態遷移図

ここで上の性質(I), (II)をもつ順序回路はつぎの(i)~(IV)の特徴をもつ順序回路であることが分る。

- (i) 入力記号 I を  $n$  個連ねた入力系列は、それに対する出力系列中に現れる出力記号 “1” の位置により初期状態および最終状態が知れるので、この順序回路に対する homing sequence かつ distinguishing sequence である。

すなわち出力系列の  $n-i+1$  番目に出力記号 “1” が現われれば、その初期状態は  $S_i$  であることが分る。なお検査系列の構成に当っては、この入力記号 I を  $n$  個連ねた distinguishing sequence の代りにつぎのような入出力系列を用いる。

〔定義 3-1〕 性質(I), (II)を持った  $n$  状態の順序回路について、状態  $S_i$  に対する  $D_i$ -系列とはつぎのような長さ  $(n-i+1)$  の入力系列である。

- (a) 入力系列は入力記号 I を  $(n-i+1)$  個連ねた系列であり、  
(b) 出力系列は出力記号 “0” を  $(n-i)$  個連ね、その後出力記号 “1” を一つ連ねた系列である。

もし与えられた順序回路が性質(I), (II)の状態遷移関数、出力関数をもった順序回路であり、この  $D_i$ -系列を満足するときは、その初期状態は  $S_i$  であり、その最終状態は  $i$  に関係なく  $S_1$  であることが分る。すなわちこの  $D_i$ -系列は distinguishing sequence および homing sequence の代りとして用いることができる。

- (ii) この順序回路の内部状態が状態  $S_j$  であるとき、内部状態を状態  $S_i$  ( $i > j$ ) にもってゆくには入力記号 I を  $i-j$  個加えればよい。(任意の状態から任意の状態への transfer sequence が存在する。

なお検査系列の構成に当っては、その構成手順を組織的にするため常に状態  $S_1$  から各状態  $S_i$  へ遷移させることにするので、このための transfer sequence をつぎのように定義する。

〔定義 3-2〕 性質(I), (II)を持った順序回路の状態  $S_i$  への  $T_i$ - 系列とはつぎのような長さ  $(i-1)$  の入出力系列である。

(a) 入力系列は入力記号  $I$  を  $(i-1)$  個連ねた系列であり、

(b) 出力系列は出力記号 “0” を  $(i-1)$  個連ねた系列である。

(iii) 与えられた順序回路が性質(I), (II)を持った順序回路であるかどうかを判定できる入出力系列が存在する。

この判定のための入出力系列としてつぎのような入出力系列を定義する。

〔定義 3-3〕 性質(I), (II)を持った  $n$  状態の順序回路について、 $\alpha_n$ - 系列とはつぎのような長さ  $(2n-1)$  の入出力系列である。

(a) 入力系列は入力記号  $I$  を  $(2n-1)$  個連ねた系列であり、

(b) 出力系列はつぎの三つの系列を順次つないだ系列である。

(1) 出力記号 “0” を  $(n-1)$  個連ねた系列

(2) 出力記号 “1” 1つからなる系列

(3) 出力記号 “0” を  $(n-1)$  個連ねた系列

この  $\alpha_n$ - 系列を満足する順序回路は  $n$  個以上の状態を持ち、もし  $n$  状態の順序回路がこの  $\alpha_n$ - 系列を満足するならその順序回路は性質(I), (II)の状態遷移関数、および出力関数を持った順序回路であることが証明される。

(付録 I 参照)。なお与えられた順序回路が性質(I), (II)を持った順序回路であっても、初期状態が  $S_1$  でなければこの  $\alpha_n$ - 系列を満足しない。そこで検査のはじめには状態を  $S_1$  にするため、adaptiveに入力記号  $I$  を 1 個ずつ出力に “1” が出るまで加える。出力 “1” が出れば状態が  $S_1$  になっている可能性があるのでそのつぎから  $\alpha_n$ - 系列の入力系列を加え、そのときの出力系列と  $\alpha_n$ - 系列の出力系列との一致を調べることになる\*。

(iv) 性質(I), (II)を持つ順序回路に入力記号  $I$  を連続して加えると、出力系列は  $n$  個ごとに出力 “1” を出すから、長さ  $n$  のサイクルをもつ。

---

\* Preset な検査でその順序回路が  $\alpha_n$ - 系列を満足するかどうかを判定することもできる。

以後この入力  $I$  によるサイクルをカウンタ・サイクル  $C_{\alpha_I}$  といい、この性質(I)、(II)を持つ順序回路を  $C_{\alpha_I}$  を持つ CC-型順序回路 (Counter Cycletype sequential circuit) と呼ぶことにする。

〔例 3-1〕 ここで状態数が 3 の場合 ( $n=3$ ) について上で定義した各種の入出力系列の例を示す。入出力系列は以後この論文では入力系列を上書き、その対応する出力系列をその下を書くといった形式で示すことにする。状態数が 3 の  $C_{\alpha_1}$  をもつ CC-型順序回路の各状態について  $D_i$ -,  $T_i$ -系列は表 3-1 のようになる。

表 3-1  $n=3$  の場合の  $D_i$ -,  $T_i$ -系列

状態	$D_i$ -系列	$T_i$ -系列
$S_1$	$\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$	$\wedge^*$
$S_2$	$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$
$S_3$	$\begin{pmatrix} 1 \\ 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$

\*  $\wedge$  は長さ 0 の系列

また  $\alpha_n$ -系列は

$$\alpha_3 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

となる。そしてこの  $\alpha_3$ -系列を満足する 3 状態の順序回路の入力 "1" に対する状態遷移図は図 3-2 のように一意的に定まる。

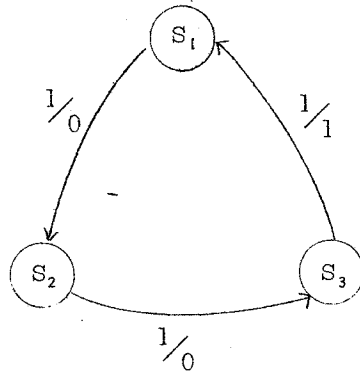


図 3-2 3 状態の  $C_{11}$  をもつ  $CC$ -型  
順序回路の状態遷移図

### 3.3. 順序回路の故障検査系列

この節では順序回路の故障検査系列の構成法について考察する。

#### 3.3.1. 故障に対する仮定

ここで考える順序回路の故障はつぎの条件を満足するものと仮定する。

条件 I

- (1) 故障によって別の状態遷移図を持った順序回路となる。
- (2) 故障によって内部状態の数は増加しない。
- (3) 故障は定常的である。

#### 3.3.2. $CC$ -型順序回路に対する故障検査系列の構成

ここではカウンタ・サイクル  $C_{11}$  をもった  $n$  状態  $m$  入力 の  $CC$ -型順序回路に対する故障検査系列の構成手順を考察する。なお検査のはじめには順序回路は状態  $S_1$  にあるものとする。

故障検査系列の構成手順とその意味

- (1)  $\alpha_n$ -系列を作り、その後に入出力対  $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$  をつけ加えた系列を  $\beta_n$ -系列と定義する。 $(\beta_n = \alpha_n \begin{pmatrix} 1 \\ 1 \end{pmatrix})$

状態  $S_1$  にあった順序回路に  $\alpha_n$ -系列の入力系列を加えると、順序回路

が正常であれば最終状態は  $S_n$  となる。以下の検査系列の構成を組織的に行うには最終状態を  $S_1$  にしておく都合がよいので、 $\alpha_n$ -系列の後に入出力対  $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$  を加えた系列  $\beta_n$  を作り、この系列で性質(I)、(II)の検査を行うと共に、そのときの最終状態を  $S_1$  にする。

- (2) 各状態に対する  $T_i$ -系列、 $D_i$ -系列を作る。さらに、与えられた順序回路において、状態  $S_i$  から入力  $I_j$  によって出力  $O_j$  を出して状態  $S_k$  へ遷移するとき、この状態遷移を検査する系列としてつぎの  $\delta_{ij}$ -系列を定義する。

$$\delta_{ij}\text{-系列} = (T_i\text{-系列}) \begin{pmatrix} I_j \\ 0_j \end{pmatrix} (D_k\text{-系列})$$

すべての状態  $S_i$  ( $i=1, 2, \dots, n$ ) とすべての入力  $I_j$  ( $j=1, 2, \dots, m$  ただし入力“1”は除く) の組合せに対し  $\delta_{ij}$ -系列を作る。この  $\delta_{ij}$ -系列は、初期状態が  $S_1$  であるとき、まず  $T_i$ -系列で状態を  $S_i$  に移し、つぎに状態  $S_i$  からの入力  $I_j$  による出力が  $O_j$  であるかを検査し、そのときの遷移先の状態  $S_k$  であるかどうかを  $D_k$ -系列で確かめるという働きをもった入出力系列である。ここで入力“1”による状態遷移は  $\beta_n$ -系列で確かめられているのでそれに対しては  $\delta_{ij}$ -系列を作らない。

- (3)  $\beta_n$ -系列の後に(2)で得たすべての  $\delta_{ij}$ -系列を任意の順序で連ねる。

ここで  $\beta_n$ -系列の後に  $\delta_{ij}$ -系列を連ねたのは、 $\beta_n$ -系列の検査で入力“1”による状態遷移が正しいと分ればはじめて  $\delta_{ij}$ -系列で各状態遷移の検査が行えることになるからである。また順序回路が正常であるなら各  $\delta_{ij}$ -系列は状態  $S_i$  から始まり状態  $S_i$  に終るから、 $\delta_{ij}$ -系列のつなぎ方は任意の順序でよい。

〔例3-2〕 表3-2に与えられた順序回路  $M_3$  に対する故障検査系列を求める。順序回路  $M_3$  はカウンタ・サイクル  $C_{n=1}$  をもった3状態のCC-型順序回路である。このとき  $n=3$  であるから各  $D_i$ -系列、 $T_i$ -系列は

表 3-2 順序回路  $M_{31}$

状態 \ 入力	0	1
$S_1$	$S_1(1)$	$S_2(0)$
$S_2$	$S_2(1)$	$S_3(0)$
$S_3$	$S_3(1)$	$S_1(1)$

表 3-1 に示されるものである。また  $\beta_n$ -系列は

$$\beta_3 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

であり、入力“0”に対する各  $\delta_{ij}$ -系列は表 3-3 となる。したがって順

表 3-3  $M_{31}$  に対する  $\delta_{ij}$ -系列

状態 \ 入力	0
$S_1$	$\begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$
$S_2$	$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$
$S_3$	$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

序回路  $M_{31}$  に対する故障検査系列はつぎのようになる。

$$\begin{aligned} \text{故障検査系列} = & \quad 1111111 \quad 0111 \quad 1011 \quad 1101 \\ & \quad 001001 \quad 1001 \quad 0101 \quad 0011 \end{aligned}$$

### 3.3.3 与えられた順序回路の CC- 型順序回路への拡大

3.3.2 では CC- 型順序回路に対する故障検査系列は比較的簡単に求めら

れることを示した。そこでつぎに、CC-型でない順序回路をそれを含む\*  
CC-列順序回路に拡大することを考える。すなわちn状態の順序回路に対  
してつぎの手順によりその順序回路がカウンタ・サイクルをもつようにでき  
る。

一般の順序回路のCC-型順序回路への拡大

状態遷移表にあらたな入力記号(以後“ $\epsilon$ ”で示す)に対応する列を  
設ける。この列の状態遷移関数  $T(S_i, \epsilon)$  および出力関数  $O(S_i, \epsilon)$   
はつぎのように定める。

$$T(S_i, \epsilon) = S_{i+1}, \quad O(S_i, \epsilon) = "0" \quad (i = 1, 2, \dots, n-1)$$

$$T(S_n, \epsilon) = S_1, \quad O(S_n, \epsilon) = "1"$$

したがって入力記号を1個つけ加えることによって一般の順序回路をCC-  
型順序回路にすることができる。この手順により拡大されたCC-型順序回  
路に対する故障検査系列は3.3.2の検査系列構成手順において入力記号“1”  
を“ $\epsilon$ ”に変えることによりただちに得られる。

[例3-3] 表3-4に与えられたCC-型でない順序回路をCC-型

表 3-4      順序回路  $M_{32}$

入 力 状 態	0	1
$S_1$	$S_3(0)$	$S_3(1)$
$S_2$	$S_3(0)$	$S_1(0)$
$S_3$	$S_2(1)$	$S_3(1)$

に拡大し、それに対する故障検査系列を求める。拡大手順により  $M_{32}$  を含む  
拡大されたCC-型順序回路  $M'_{32}$  は表3-5となる。 $M'_{32}$  に対する各  $D_{i-1}$ ,

\* 順序回路Mが満足するすべての入力系列を順序回路M'も満足するとき、順序回路M'  
は順序回路Mを含むという。



表 3-5 順序回路  $M'_{32}$

状態 \ 入力	0	1	$\epsilon$
$S_1$	$S_3(0)$	$S_3(1)$	$S_2(0)$
$S_2$	$S_3(0)$	$S_1(0)$	$S_3(0)$
$S_3$	$S_2(1)$	$S_3(1)$	$S_1(1)$

$T_i$ -系列は表 3-6 となり、 $\beta_n$ -系列は

表 3-6  $M'_{32}$  に対する  $D_i$ -,  $T_i$ -系列

状態	$D_i$ -系列	$T_i$ -系列
$S_1$	$\begin{pmatrix} \epsilon & \epsilon & \epsilon \\ 0 & 0 & 1 \end{pmatrix}$	$\wedge$
$S_2$	$\begin{pmatrix} \epsilon & \epsilon \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \epsilon \\ 0 \end{pmatrix}$
$S_3$	$\begin{pmatrix} \epsilon \\ 1 \end{pmatrix}$	$\begin{pmatrix} \epsilon & \epsilon \\ 0 & 0 \end{pmatrix}$

$$\beta_3 = \begin{pmatrix} \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

となる。各  $\delta_{ij}$ -系列は表 3-7 となり結局  $M'_{32}$  に対する故障検査系列はつ

表 3-7  $M'_{32}$  に対する  $\delta_{ij}$ -系列

状態 \ 入力	0	1
$S_1$	$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \begin{pmatrix} \epsilon \\ 1 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 1 \end{pmatrix} \begin{pmatrix} \epsilon \\ 1 \end{pmatrix}$
$S_2$	$\begin{pmatrix} \epsilon \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} \begin{pmatrix} \epsilon \\ 1 \end{pmatrix}$	$\begin{pmatrix} \epsilon \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} \epsilon & \epsilon & \epsilon \\ 0 & 0 & 1 \end{pmatrix}$
$S_3$	$\begin{pmatrix} \epsilon & \epsilon \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} \epsilon & \epsilon \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \epsilon & \epsilon \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \begin{pmatrix} \epsilon \\ 1 \end{pmatrix}$

ぎのようになる。

故障検査系列 =  $\begin{array}{cccccccc} \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & 0\epsilon & \epsilon 0\epsilon & \epsilon\epsilon 0\epsilon\epsilon & 1\epsilon & \epsilon 1\epsilon & \epsilon\epsilon\epsilon & \epsilon\epsilon 1\epsilon \\ 001001 & 01 & 001 & 00101 & 11 & 00001 & 0011 & & & & & & & \end{array}$

### 3.3.4. 故障検査系列の長さ と 付加するハードウェアについての考察

まず 3.3.2. および 3.3.3. の手順によって得られる故障検査系列の長さを概算しておく。ここでは与えられた順序回路は  $n$  状態で  $m$  種の入力記号をもつものとする。

- (1)  $\beta_n$ -系列の長さは  $2n$
- (2)  $\delta_{ij}$ -系列の長さはつぎの三つの系列の長さの和となる。
  - (i)  $T_i$ -系列の長さは最大  $n-1$
  - (ii) 各状態遷移に対する入出力対の長さは 1。
  - (iii)  $D_i$ -系列の長さは最大  $n$ 。

したがって  $\delta_{ij}$ -系列の長さは高々  $2n$ 、また  $\delta_{ij}$ -系列の総数は  $n \times m$  である。

以上より求める故障検査系列の長さは高々

$$2n + 2n \times (n \times m) \approx mn^2$$

であり、故障検査系列の長さとしては従来の方法より少し短くなっていると思われる。<sup>\*</sup>

3.3.3. では与えられた順序回路が  $00$ -型でない場合には新しい入力記号“ $\epsilon$ ”をつけ加える方法を示した。そこでつぎにこの新しい入力記号を付加することによって生じるハード・ウェアの増加について考えると、

- (1) 余分の記憶素子は必要としない（∵状態数に増加はないから）
- (2) 新しい入力記号に対する状態遷移を定めるための組合せ回路が必要である。
- (3) 新しい入力線が必要な場合がある。

---

\* 文献〔19〕の方法では検査系列の長さの上限は  $mn^2$  である。

ということになる。

しかしこの(2)および(3)の要求によるハード・ウェアの増加がそれほど多くはない場合がある。たとえば要求(2)については、状態  $S_i$  から状態  $S_{i+1}$  へ入力 “ $\epsilon$ ” 以外の入力  $I$  による状態遷移がある場合を考えると、各記憶素子の入力方程式は  $I$  を含む項には “ $\epsilon$ ” も含むから、 $I$  と “ $\epsilon$ ” の入力変数割当てを符号的に近くしておくとその符号間で簡単化がなされ、この状態遷移を定めるためのハード・ウェアは少なくすむであろう。また(3)の要求については、入力の数  $2$  の冪乗でない場合を考えると、入力変数の割当てで余っている変数の組合せが存在するから、“ $\epsilon$ ” にその組合せを用いることによって、余分の入力線は不用であることになる。

さらに、順序回路を  $CC$ -型に拡大しておくで故障検出だけでなく、次節で述べるように若干の故障診断ができるなどの利点があるので、この拡大の操作はそれほど不利な条件ではないと考えることができる。また  $CC$ -型への拡大を行うと状態遷移図は必然的に強連結となるから、はじめに与えられる順序回路は必ずしも強連結である必要がないなどの特徴もある。

### 3.4. 順序回路の故障診断に関する考察

この節では前節までの考えを順序回路の故障診断に適用することを考える。

#### 3.4.1. 定義と仮定

ここでは順序回路の故障診断をつぎのように定義する。

〔定義 3-4〕 入出力系列の対応関係から与えられた順序回路の状態遷移表を決定することを順序回路の故障診断と呼ぶ。

ここで診断の対象とする順序回路は  $CC$ -型であるとし、故障は 3.3.1. の条件 I の他にさらにつぎの条件を満足するものと仮定する。

#### 条件 II

カウンタ・サイクル  $C_n$  の入力記号 “ $\epsilon$ ” に対する状態遷移は正しく

行われる。

条件Ⅱが成立しないような故障については 3.4.3. で考察する。

### 3.4.2. 故障診断の方法

ここでは adaptive な検査で故障診断を行うことを考える。<sup>\*</sup> 条件Ⅱに

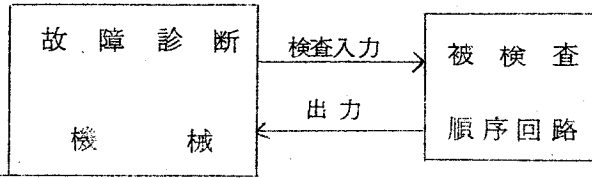


図 3-3 故障診断操作

より 3.2 で考えた  $D_i$ - 系列、 $T_i$ - 系列は故障している順序回路に対しても使用できる。したがってカウンタ・サイクル  $C_{\epsilon}$  を持った  $n$  状態  $m$  入力の  $CC$ - 型順序回路に対しては、つぎの手順により故障診断を行うことができる。

#### 故障診断手順

- (1) 入力記号 " $\epsilon$ " を出力端に出力記号 "1" が得られるまで加える。
- (2)へ。(もし  $n$  個以上の入力記号 " $\epsilon$ " を加えても出力記号 "1" が得られないなら、その順序回路は条件Ⅱを満足しない故障をしている。)
- (2)  $\beta_n$ -系列の入力系列を加えそのときの出力系列と  $\beta_n$ -系列の出力系列とを比較する。二つの出力系列が一致すれば(3)へ。(もし二つの出力系列が一致しなければその順序回路は条件Ⅱを満足しない故障をしている。)
- (3) 一般に、状態  $S_i$  の入力  $I_j$  による状態遷移を診断するにはつぎの

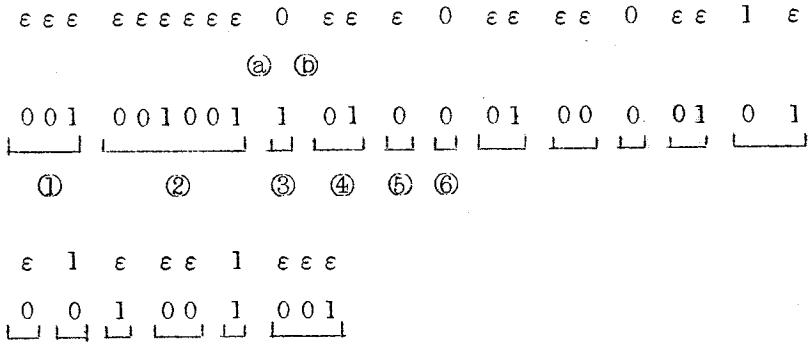
\* preset な検査によっても診断可能であることが知られている。

ようにする。

- (i)  $T_i$ -系列の入力系列を加える。
- (ii) 入力記号  $I_j$  を加える。このときもし出力  $O_j$  が得られるなら、この順序回路は状態  $S_i$  のとき入力  $I_j$  によって出力  $O_j$  を出すと結論できる。
- (iii) 入力記号 “ $\epsilon$ ” を出力端に出力 “1” が得られるまで加える。もし入力記号 “ $\epsilon$ ” を  $k$  個加えたとき初めて “1” が得られたなら、この順序回路は状態  $S_i$  から入力  $I_j$  により状態  $S_{n-k+1}$  へ遷移すると結論できる。

すべての可能な状態遷移 ( $i = 1, 2, \dots, n, j = 1, 2, \dots, m$ ) に対し、(i)~(iii)をくり返し実行する。

〔例3-4〕 条件 I、II、を満足する故障をした3状態のCC-型順序回路が与えられたとする。この順序回路に上の手順を適用してつぎの入出力系列 (図3-4) を得たとすると、この順序回路の状態遷移表は表3-8で



- ① step (1) による。
- ② step (2) による。入力 “ $\epsilon$ ” による遷移は正しい。また状態④は  $S_1$  であることが分る。
- ③ step (3) (ii) による。  $S_1$  にあるときは入力 “0” により出力 “1” を出すことが分る。
- ④ step (3) (iii) による。状態④は  $S_2$  であることが分るから、 $S_1$  からは入力 “0” により  $S_2$  へ遷移する。
- ⑤ step (3) (i) による、状態を  $S_2$  へもってゆく。
- ⑥ 以下③~⑤と同様の操作の繰返し。

図 3-4 操作 C による故障診断

表 3-8 順序回路 M

状態 \ 入力	0	1	$\epsilon$
$S_1$	$S_2(1)$	$S_3(0)$	$S_2(0)$
$S_2$	$S_2(0)$	$S_3(0)$	$S_3(0)$
$S_3$	$S_2(0)$	$S_1(1)$	$S_1(1)$

あることが分る。

### 3.4.3. 条件 II に対する考察

ここでは 3.4.1. の条件 II を満足しない故障について考察する。このときつぎの二つの場合が考えられる。

(i) 診断手順の(1)において出力“1”が得られない場合。

この場合の可能な故障はつぎの二つである。

1. 出力回路が故障している。
2. 状態遷移表が図 3-5 に示される形の故障をしている。

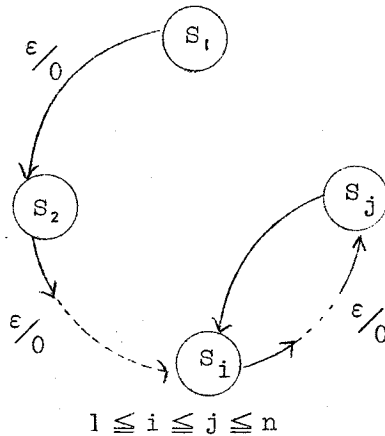


図 3-5 故障順序回路

これらの故障に対する診断の有効な方法はまだ知られていない。

(ii) 診断手順の(2)において  $\beta_n$ -系列の入力系列に対する出力系列が  $\beta_n$ -系列の出力系列と一致しない場合。

このときにも故障によって順序回路の状態数は増加しないと仮定しているから(条件Iの(2))、 $\beta_n$ -系列の入力系列に対する可能な出力系列( $n$ 状態の順序回路が出しうる出力系列)の数はそれほど多くない(表3-9\*参照)

表3-9  $\beta_n$ -系列の入力系列に対する出力系列の数

状態数 $n$	可能な出力系列 の数 $a_n$	$\epsilon$ に対する遷移を 一意的に定める出 力系列の数 $D_n$	$b_n/a_n$
2	3	2	66.6 (%)
3	9	6	66.6
4	24	15	62.5
5	63	39	61.9
6	153	90	58.8
7	369	216	58.5
8	858	489	56.9

たとえば3状態の順序回路は $\beta_3$ -系列の入力系列に対してつぎのような出力系列は出し得ない。

$$\left( \begin{array}{l} \beta_3\text{-系列の入力系列} \\ 3\text{状態の順序回路で} \\ \text{は不可能な出力系列} \end{array} \right) = \begin{pmatrix} \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

(この入出力系列を満足する順序回路の状態数は4以上である)。さらに、

\* 本学計算センター、計算機(NEAC 2206)により求めた。

$\beta_n$ -系列の入力系列に対する可能な出力系列のうちでいくらかのものは入力記号“ $\epsilon$ ”に対する状態遷移を一意的に決定することができる。たとえば  $\beta_3$ -系列の入力系列に対し、つぎの出力系列

$$\left( \begin{array}{l} \beta_3\text{-系列の入力系列} \\ \text{“}\epsilon\text{”に対する遷移を一意的に} \\ \text{定めることのできる出力系列} \end{array} \right) = \begin{pmatrix} \epsilon & \epsilon & \epsilon & \epsilon & \epsilon & \epsilon \\ 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}$$

を出す3状態の順序回路の入力記号“ $\epsilon$ ”による状態遷移は一意的に定まる。(図3-6)。この場合には、各状態は表3-10のようなあらたな  $D'_i$ -

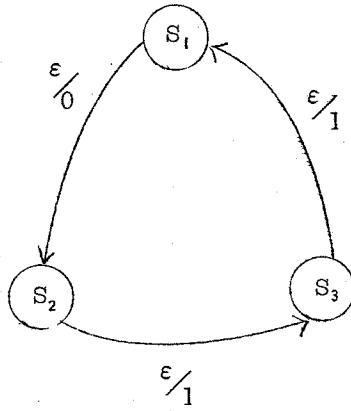


図3-6 順序回路 M

$T'_i$ -系列に当る  $D'_i$ - ,  $T'_i$ -系列を持つ。したがってこれらの  $D'_i$ - ,  $T'_i$ -系

表 3-10 Mに対する  $D'_i$ - ,  $T'_i$ -系列

状態	$D'_i$ -系列	$T'_i$ -系列
$S_1$	$\begin{pmatrix} \epsilon \\ 0 \end{pmatrix}$	$\begin{pmatrix} \epsilon & \epsilon \\ 1 & 1 \end{pmatrix}$
$S_2$	$\begin{pmatrix} \epsilon & \epsilon & \epsilon \\ 1 & 1 & 0 \end{pmatrix}$	$\wedge$
$S_3$	$\begin{pmatrix} \epsilon & \epsilon \\ 1 & 0 \end{pmatrix}$	$\begin{pmatrix} \epsilon \\ 1 \end{pmatrix}$



列を使えば、その順序回路に対する故障診断は 3.4.2. の故障診断手順とまったく同様に行うことができる。 $\beta_n$ -系列の入力系列に対し  $n$  状態の順序回路が出すことが可能な出力系列の数および入力 “ $\epsilon$ ” に対する状態遷移が一意的に決定できるような出力系列の数は表 3-9 に示されている。表 3-9 から  $\beta_n$ -系列の入力系列に対する可能な出力応答のうち、およそ 60% (すべての故障の 60% ではない) の場合は、その順序回路の故障診断が可能であるといえる。

#### 3.4.4. 故障診断の容易な他の順序回路

以上までに述べたことから、与えられた順序回路を CC-型に拡大して構成しておくことと故障検査系列が比較的簡単に求められるほか、部分的な故障診断にも有利であろうことが分った。しかし CC-型順序回路以外にもこれまでとほとんど同様の手順によって比較的簡単に故障検査系列を求めることのできる順序回路を考えることができる。ここではそのうちの二つの例をあげる。

- (1) ある一つの入力記号  $I$  に対しつぎのような状態遷移関数  $T(S_i, I)$  および出力関数  $O(S_i, I)$  をもつ順序回路。

$$T(S_i, I) = S_{i+1}, \quad O(S_i, I) = "0" \quad (i = 1, 2, \dots, n-1)$$

$$T(S_n, I) = S_n, \quad O(S_i, I) = "1"$$

この順序回路は故障検査系列の構成が簡単であるという特徴のほか、さらにつぎの特徴がある。すなわち、入力記号  $I$  を  $n$  個連ねた入力系列はこの順序回路の synchronizing sequence\* となっているから、検査の開始時において初期状態設定の操作が簡単であろうことである。

- (2) ある一つの入力記号  $I$  に対しつぎのような状態遷移関数  $T(S_i, I)$  および出力関数  $O(S_i, I)$  をもつ順序回路。

---

\* 初期状態が何であってもその入力系列を加えることによって、その順序回路をある特定の状態にもってゆくことができるような入力系列。

$$T(S_i, I) = S_i, O(S_i, I) = O_i \quad (i = 1, 2, \dots, n)$$

$$\text{ただし } O_i \neq O_j \quad (i \neq j)$$

この順序回路においては 3.2. で定義された  $D_i$ -系列に当る系列の長さが状態数に関係なく 1 であるから、故障検査系列全体の長さも短くなることが期待できる。

この(1)、(2)以外にも 2 種以上の入力記号を付加すること等により、種々の拡大方法が考えられるが、どれが最適であるかを定めることは回路の複雑さの基準等とも関連をもち、一般にはむずかしい問題である。

### 3.5 結 言

この章では故障検査を容易にするための順序回路の構成法について考察した。前半では、故障検査の容易な順序回路として  $CC$ -型の順序回路を考えその順序回路に対する故障検査系列の構成手順を示した。また与えられた順序回路が  $CC$ -型でない場合にはあらたな入力記号を 1 つ付加することによってそれを含む  $CC$ -型順序回路に変更できることを示した。後半では、 $CC$ -型順序回路に対する故障診断について考察したが、与えられた順序回路が  $CC$ -型でない場合においても、もしある特定の一つの入力記号のみからなる入力系列に対してその順序回路がその入力記号に対する状態遷移を一意的に定めるような出力応答をするなら、その順序回路の故障診断を行いうることが分った。

一般に、順序回路の故障検査系列を求める方法について考えるとき考慮すべきことは、その手順が簡単であること、およびその方法によると短い検査系列が得られること等であるが、ここで論じたように余分の入力や出力記号を付加するとこの二つの要求をある程度満足する方法が存在するものと思われる。

## 第4章 系列発生器の一構成法

### 4.1. 序 言

順序回路に対する故障検査系列が求まった後、実際にその順序回路の故障検査を行う時問題となるのはどのようにしてその検査系列を発生させるかということであろう。このようにある定められた系列を作り出すことは、順序回路の故障検査に限らず他のデジタルシステムを駆動したり、標準パターンを発生したりするときにもしばしば必要となる問題である。

順序回路の故障検査系列を発生させる時には、入力系列と出力系列を同時に作ることも、簡単に別の異なった系列を発生できるように変更できることなどが要求されるが、ここではこれらの要求を満たし、発生する系列の長さを基準としたとき必要とする記憶素子数が最少のオートノマス順序回路による系列発生器を構成することを考える。

### 4.2. 2値系列の合成

ここでは与えられた2値の時間系列を作るのに、周期性をもった2値系列をいくつか準備しておき、それらのうちのさらにいくつかを用いて与えられた2値系列を合成することにする。

#### 4.2.1. 定義と記号

まず2値系列を表現するためつぎの定義を行う。

[定義4-1]\* 行列  $M_i$  は“0”, “1” を要素とする  $2^i \times 2^i$  の正方行列で、つぎのように帰納的に定義される。

$$M_1 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \quad M_i = \begin{bmatrix} M_{i-1} & O_{i-1} \\ M_{i-1} & M_{i-1} \end{bmatrix}$$

\* 文献 [24] P. 118 参照。

ここで  $0_{i-1}$  は  $2^{i-1} \times 2^{i-1}$  の零行列を表わす。

〔例 4-1〕

$$M_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad \text{であり} \quad M_3 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad \text{である。}$$

〔定義 4-2〕 行列  $M_i$  の第  $j$  行を  $2^i$  次元の 2 元ベクトルとみなし、記号的に  $m_j^i$  で表わす。行列  $M_i$  の行ベクトル  $m_j^i$  ( $j = 1, 2, \dots, 2^i$ ) のおのおのを行列  $M_i$  の生成ベクトルと呼ぶ。

上で定義された行列  $M_i$  の生成ベクトルは  $2^i$  個の成分をもつ  $i$  次の Reed-Muller 符号<sup>[27]</sup> 全体に当るものである。本章では、与えられた長さ  $l$  の 2 値系列を発生させることを考えるとき、それを  $l$  次元の 2 元ベクトルとみなし、そのベクトルを上で定義した生成ベクトルから合成することにする。その合成の操作として、2 元ベクトル間の排他的論理和および論理積を定義する。

〔定義 4-3〕 一般に  $n$  次元の 2 つのベクトル  $m_1 = (m_{11}, m_{12}, \dots, m_{1n})$  と  $m_2 = (m_{21}, m_{22}, \dots, m_{2n})$  ここで  $m_{ij}$  は “0” または “1” に対し

(i) 各成分同志の排他的論理和を成分とするベクトルをベクトル  $m_1$  と  $m_2$  の排他的論理和と呼び、2 値変数に対するときと同じ記号  $\oplus$  を用いて表わす。

$$m_1 \oplus m_2 = (m_{11} \oplus m_{21}, m_{12} \oplus m_{22}, \dots, m_{1n} \oplus m_{2n}).$$

(ii) 各成分同志の論理積を成分とするベクトルをベクトル  $m_1$  と  $m_2$  の論理積と呼び、2 値変数に対するときと同じ記号  $\cdot$  を用いて表わす。

$$m_1 \cdot m_2 = (m_{11} \cdot m_{21}, m_{12} \cdot m_{22}, \dots, m_{1n} \cdot m_{2n})。$$

#### 4.2.2. 長さが2の冪乗である2値系列の合成

まず長さが2の冪乗である2値系列を合成することを考える。定義4-2における行列  $M_i$  の各生成ベクトルは  $i$  次の Reed-Muller 符号全体に当ることからつぎの補題が成立することが分る。

##### 補題4-1

$2^i$  次元の任意の2元ベクトル  $m$  に対し、行列  $M_i$  の生成ベクトルの中から適当なベクトル  $m_{j_1}^i, m_{j_2}^i, \dots, m_{j_k}^i$  を選び

$$m = m_{j_1}^i \oplus m_{j_2}^i \oplus \dots \oplus m_{j_k}^i$$

と表わすことができる。このとき  $m_{j_1}^i, m_{j_2}^i, \dots, m_{j_k}^i$  の選び方は一意的である。

##### 証明 付録II参照

補題4-1より、 $2^i$  次元の任意の2元ベクトルは行列  $M_i$  の適当な生成ベクトルの排他的論理和により一意的に表わせることが言える。そこでつぎに与えられた  $2^i$  次元の2元ベクトルを  $M_i$  の生成ベクトルへ分解する方法について考える。行列  $M_i$  の主対角線上の要素はすべて“1”であり、主対角線より右上はすべて“0”である。したがって与えられた2元ベクトルを  $M_i$  の生成ベクトルの排他的論理和によって表わすためには、主対角線上の“1”に着目して与えられた2元ベクトルを分解すればよい。この操作はつぎのようになる。

操作I ( $2^i$  次元の2元ベクトル  $m$  を行列  $M_i$  の生成ベクトルへ分解する操作。)

- (1) 行列  $M_i$  を作る。
- (2) あらたなベクトル  $v$  を設け、最初  $v = m$  とする。
- (3)  $v$  の要素の“1”のうち最も右にある“1”をみつける。(4)へ。

もし  $v = 0$  ( $v$  の要素がすべて "0" ) であるとき操作終る。(6)へ。

(4)  $v$  の第  $j$  桁目の "1" が最も右の "1" であったとき、 $M_i$  の生成ベクトル  $m_j^i$  は  $m$  を合成するのに必要なベクトルである。

(5)  $m_j^i \oplus v$  をあらためて  $v$  とみなし(3)へ。

(6) これまでの過程で段階(4)で得た  $m_j^i$  のすべての排他的論理和を作るとベクトル  $m$  となる。

[例 4-2]  $8 (=2^3)$  次元の 2 元ベクトル  $m = 10011010$  を行列  $M_3$  の生成ベクトルに分解することを考える。

操作 I の(2)において  $v = m = 10011010$

(3)において  $v$  はその第 7 桁目に最も右の "1" をもつ。

(4)より、生成ベクトル  $m_7^3$  は  $m$  を合成するのに必要なベクトルである。

(5)より、 $v := v \oplus m_7^3 = 00110000$

(3)において、 $v$  はその第 4 桁目に最も右の "1" をもつ。

(4)より、 $m_4^3$  は  $m$  を合成するのに必要なベクトルである。

(5)より、 $v := v \oplus m_4^3 = 11000000$

(3)において、 $v$  はその第 2 桁目に最も右の "1" をもつ。

(4)より、 $m_2^3$  は  $m$  を合成するのに必要なベクトルである。

(5)より、 $v := v \oplus m_2^3 = 00000000$

(3)において、 $v = 0$  であるから(6)へ。

(6)より  $m = m_7^3 \oplus m_4^3 \oplus m_2^3$  と表わせる。

#### 4.2.3. 長さが 2 の冪乗でない 2 値系列の合成

4.2.2. では長さが 2 の冪乗 ( $2^i$ ) である 2 値系列を行列  $M_i$  の生成ベクトルを用いて合成することを考えた。ここでは長さ  $\ell$  が 2 の冪乗でない

( $2^{i-1} < \ell < 2^i$ ) 2 値系列を行列  $M_i$  の生成ベクトルを用いて合成することを考える。まずつぎの定義を行ひ。

[定義 4-4]  $\ell$  次元 ( $2^{i-1} < \ell < 2^i$ ) の 2 元ベクトル  $m = (m_{11}, m_{12}$

……,  $m_\ell$ ) に対し、

$$P_{2^i}^i(m) = (m_1, m_2, \dots, m_\ell, p_1, p_2, \dots, p_{2^{i-\ell}})$$

ただし  $p_i$  は “0”, “1” のうちの任意の値をとってよい。

なる  $2^i$  次元の 2 元ベクトルを  $m$  の拡大ベクトルと呼ぶ。

このときつぎの補題が成立する。

補題 4-2

$\ell$  次元 ( $2^{i-1} < \ell < 2^i$ ) の 2 元ベクトル  $m$  に対し、つぎのような少なくとも一つの拡大ベクトル  $P_{2^i}^i(m)$  が存在する。

$$P_{2^i}^i(m) = m_{j_1}^i \oplus m_{j_2}^i \oplus \dots \oplus m_{j_k}^i \quad (4-1)$$

$$k \leq 2^{i-1} \quad (4-2)$$

ここで  $m_{j_h}^i$  は行列  $M_i$  の生成ベクトルを表わす。

証明  $m$  の任意の拡大ベクトルは  $2^i$  次元の 2 元ベクトルであるから、補題 4-1 より行列  $M_i$  の生成ベクトルの排他的論理和で合成することができる。そこで (4-2) 式を満たす  $k$  個以下の生成ベクトルで合成される  $m$  の拡大ベクトルが少なくとも一つ存在することを示せばよい。

$m$  のある拡大ベクトル  $v$  が  $h$  ( $h > 2^{i-1}$ ) 個の生成ベクトルの排他的論理和で表わされているとする。

$$v = m_{j_1}^i \oplus m_{j_2}^i \oplus \dots \oplus m_{j_h}^i \quad (4-3)$$

このとき行列  $M_i$  のすべての生成ベクトルのうちで (4-3) 式に用いられていない生成ベクトルすべての排他的論理和を作るとそのベクトル  $v'$  もまた  $m$  の拡大ベクトルの一つとなることが分る。なぜなら、

$$v' = m_{g_1}^i \oplus m_{g_2}^i \oplus \dots \oplus m_{g_{2^{i-h}}}^i$$

とすると

$$v \oplus v' = \overbrace{0 \ 0 \ 0 \ \dots \ 0 \ 1}^{2^i - 1 \text{ 個}}$$

であるので  $v'$  はその第  $2^i$  桁目を除き  $v$  にまったく等しい。一方  $m$  の次元は  $2^i - 1$  以下であり、 $v$  は  $m$  の拡大ベクトルであるから定義 4-4 より  $v'$  もまた  $m$  の拡大ベクトルの一つとなる。そして  $v'$  を合成している生成ベクトルの数は  $2^i - h$  であり、 $h > 2^{i-1}$  であるから結局  $v'$  は  $2^i - h < 2^i - 2^{i-1} = 2^{i-1}$  以下の生成ベクトルで合成されていることになる。

(証明終り)

補題 4-2 より、長さ  $\ell$  が 2 の冪乗でない 2 値系列を合成するのに  $\ell + 1$  時刻以後はどんな値をとってもよい\* とするなら合成に要する生成ベクトルの数は  $2^{i-1}$  個以下とできることが分る。ただし  $i = \lceil \log_2 \ell \rceil$ \*\* とする。

[例 4-3] 長さ 7 の系列  $v = 0111011$  を作ることを考えよう。これを  $8 (= 2^3)$  次元のベクトル  $v' = 01110110$  の一部とみなして  $v'$  を操作 I によって分解すると、

$$v' = m_1^3 \oplus m_2^3 \oplus m_3^3 \oplus m_4^3 \oplus m_5^3 \oplus m_7^3$$

となり 6 個の生成ベクトルの和となる。しかし  $v'$  では  $2^{3-1} = 4$  個より多くの生成ベクトルを要しているので行列  $M_3$  の全生成ベクトルのうち上式に現われない生成ベクトルの和をとると。

$$v'' = m_3^3 \oplus m_5^3 = 01110111$$

となり、第 8 桁目を無視すれば  $v''$  は  $v$  を実現している。そして  $v''$  は 2 個 ( $2^{3-1} = 4$  個以下) の生成ベクトルによって合成されている。

一般に与えられた長さ  $\ell$  が 2 の 乗でない系列  $m$  を実現するとき、その合成に要する生成ベクトルの個数を最小にするには、 $m$  の可能な拡大ベクトル

\* このようにしてもよいことは 4.3. であきらかになる。

\*\* 「 $|A|$ 」は  $A$  より小さくない最小の整数を表わす。



のおのおのについてそれに要する生成ベクトルの数を調べればよいが、\*これに関するあまり能率のよい方法はいまのところない。

### 4.3. 生成ベクトルの回路的実現

補題 4-1、4-2 より、行列  $M_i$  のすべての生成ベクトルを用意しておけば長さ  $\ell (=2^i)$  以下の任意の系列はそれらの排地的論理和として作れることが分った。そこでここでは行列  $M_i$  の生成ベクトルを時系列として発生させることのできる論理回路について考察する。そのために行列  $M_i$  の生成ベクトルのうちからさらに時系列として発生しやすいベクトルを選び、それらの論理積として他の生成ベクトルを作ることにする。まず発生しやすい生成ベクトルとしてつぎの基本生成ベクトルを定義する。

〔定義 4-4〕 行列  $M_i$  の生成ベクトルのうち  $m^j_{(2^i-2^j)}$  ( $j=0, 1, \dots, i-1$ ) および  $m^i_{2^i}$  なる  $i+1$  個の生成ベクトルを行列  $M_i$  の基本生成ベクトルと呼ぶ。

〔例 4-4〕 行列  $M_3$  に対し基本生成ベクトルとは、

$$m^3_8 = 11111111, \quad m^3_7 = 10101010,$$

$$m^3_6 = 11001100, \quad m^3_4 = 11110000.$$

の 4 つである。

定義 4-4 における行列  $M_i$  の  $i+1$  個の基本生成ベクトルは  $2^i$  個の成分の Reed-Muller 符号の基底ベクトルに当るものである。したがって符号理論における Reed-Muller 符号の性質に応じてつぎの補題が成立する。

#### 補題 4-3

行列  $M_i$  のすべての生成ベクトルは基本生成ベクトルの適当なものの論理積によって表わされる。またこの生成ベクトルの基本生成ベクトルによる合成の仕方は一意的である。

---

\* 補題 4-2 の証明より、その第  $2^i$  桁目が一定の値である系列についてだけ調べれば、他方はその  $2^i$  に関する補数として得られる。

証明 付録II参照

補題4-3より、行列  $M_1$  のすべての生成ベクトルは基本生成ベクトルの論理積として作れることが分るから、すべての生成ベクトルを発生させる回路を構成することを考えるとき、基本生成ベクトルを発生させる回路から考えてよいことになる。このとき、各基本生成ベクトルをみるとその成分の“0”、“1”の現われ方に周期性のあることが分る(例4-4参照)。さらに各基本生成ベクトルの周期については、そのおのおのの間に2の冪乗の倍数関係のあることも分る。そこでつぎのような系列変換器によって一つの基本生成ベクトルから他のすべての基本生成ベクトルを作ることにする。

〔定義4-5〕 状態遷移表4-1で与えられる順序回路(系列変換器)

表4-1 順序回路 T

状態 \ 入力	0	1	出力
$S_1$	$S_1$	$S_2$	0
$S_2$	$S_2$	$S_1$	1

Tに、初期状態を  $S_1$  として入力系列  $\alpha$  を加えたとき得られる出力系列を  $f(\alpha)$  で示す。ただし時刻0での出力は無視するものとする。

この表4-1の順序回路はT-フリップ・フロップ(T.F.F)\*そのもの

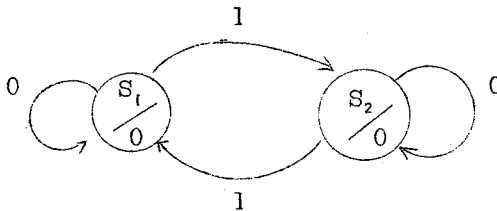


図4-1 順序回路 T

\* 文献[25]参照

であるから、T.F.F.に系列 $\alpha$ を加えたときの出力系列が $f(\alpha)$ であることになる。ここでn次元の2元ベクトルを長さnの2値系列とみなすことにすると、基本生成ベクトルの間にはつぎの関係が成立する。

補題 4-4

行列  $M_i$  の基本生成ベクトルの間にはつぎの関係式が成立する。

$$m^i_{(2^{i-1})} = f(m^i_{2^i})$$

$$m^i_{(2^{i-2^j})} = f\left(\prod_{k=0}^{j-1} \bar{m}^i_{(2^{i-2^k})}\right) \quad (j = 1, 2, \dots, i-1)$$

ここで $\bar{m}$ は $m$ の各成分の否定を成分にもつベクトルを表わし、 $\prod$ はベクトルの論理積を表わす。

証明 第一式右辺の $m^i_{2^i}$ はすべての成分が“1”の系列であり、左辺は“1”と“0”とが交互に入った系列であるから定義4-5よりあきらかに第一式は成立する。第二式右辺括弧内の系列は $2^j$ 時刻毎に値“1”をとり、他の時刻にはすべて“0”をとる系列である。一方左辺は“0”，“1”の交替の周期が $2^j$ である系列であるから、定義4-5より第二式が成立する。

補題4-4より、すべての要素が“1”である基本生成ベクトル $m^i_{2^i}$ に対応する系列(本質的には刻時パルスの系列)をT.F.F.に加えると基本生成ベクトル $m^i_{2^{i-1}}$ に対応する系列を得る。つぎに系列 $\bar{m}^i_{2^{i-1}}$ と $m^i_{2^i}$ との論理積をとった系列をT.F.F.に加えるとつぎの基本生成ベクトル $m^i_{2^{i-2}}$ に対応する系列を得る。以下帰納的に同様の操作ですべての基本生成ベクトルに対応する系列を得ることができる。そこで補題4-3、4-4をまとめて、すべての生成ベクトルに対応する系列を発生させる回路の構成方法としてつぎの操作を得る。

操作II (生成ベクトルに対応する系列を発生させる回路の構成法)。

- (1) 基本生成ベクトルのうち $m^i_{2^i}$ は刻時パルスそのものであり、 $m^i_{(2^{i-1})}$ は刻時パルスをT.F.F.一段通したもの、以下 $m^i_{(2^{i-2^j})}$ は、

$\prod_{k=0}^{j-1} m_{(2^i-2^k)}^i$  を T.E.F. 一段通したものととして得られる。(  $j = 1, 2, \dots, i-1$  )。

(2) 基本生成ベクトルでない生成ベクトル  $m_{(2^i-2^{j-k})}^i$  (  $j = 0, 1, \dots, i-1, k = 1, 2, \dots, 2^j-1$  ) は

$$m_{(2^i-2^{j-k})}^i = m_{(2^i-2^j)}^i \cdot m_{(2^i-k)}^i$$

として得られる。

操作 I および II から最終的につぎの定理を得る。

#### 定理 4-1

長さ  $\ell$  の任意の 2 値系列は刻時パルスの系列 ( $m_{2^i}^i$ ) から、T.F.F., 論理積素子、排他的論理和素子を用いて作ることができる。ここで  $i = \lceil \log_2 \ell \rceil$  である。

[例 4-5] 長さ 16 の 2 値系列  $\alpha = 0001111011010010$  を作ることを考える。操作 I より  $\alpha$  はつぎの形に表わせる。

$$\alpha = \sum_{j \in J} \oplus m_j^4 \quad J = \{ 6, 12, 13, 15 \}$$

ここで  $\sum \oplus$  は排他的論理和を表わすものである。

操作 II より、 $M_4$  のすべての生成ベクトルを実現するオートノマスな順序回路の部分として図 4-2 の破線で囲まれた部分を得る。そして求める系列  $\alpha$  は図 4-2 の全体の回路の出力として得ることができる。

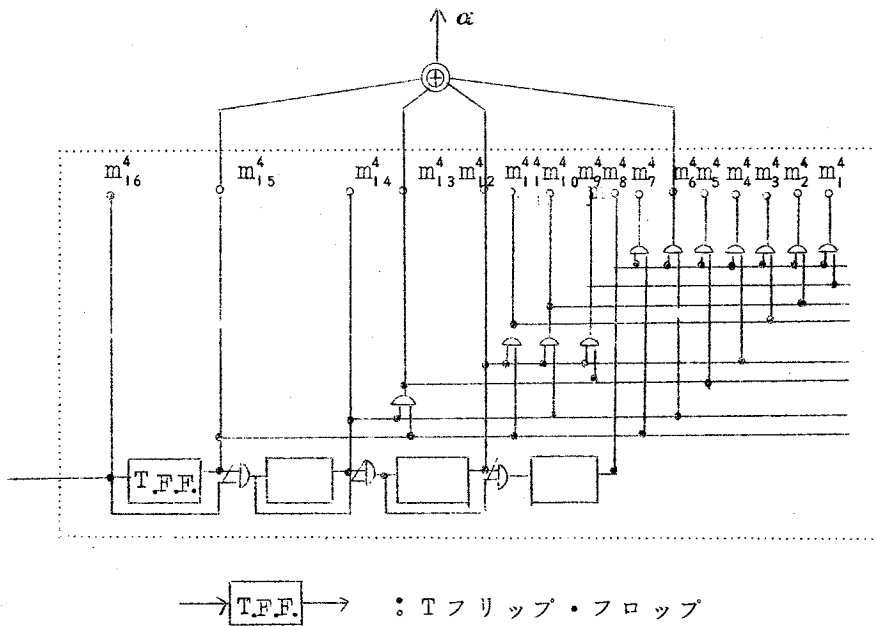


図 4-2 2 値系列の構成

ある系列を発生させるとき、その系列を繰り返して発生させることがしばしば問題となる。そこで系列を繰り返し発生させることを考えるためには、その系列の終りを検出しなければならない。ここで考えた系列発生器では各 T.F.F. のとる値の組合せは各時刻毎にすべて異なっている。したがってあらかじめその系列を発生し終った時点における各 T.F.F. の値の組合せを求めておくと、組合せ回路でそれを検出することができる。またこの系列発生器の系列発生開始前の T.F.F. の状態はすべて状態  $S_1$  になっているから、その系列の発生終了の信号ですべての T.F.F. を  $S_1$  に設定するようにすれば、求める系列を繰り返し発生することができる。(図 4-3 参照)。

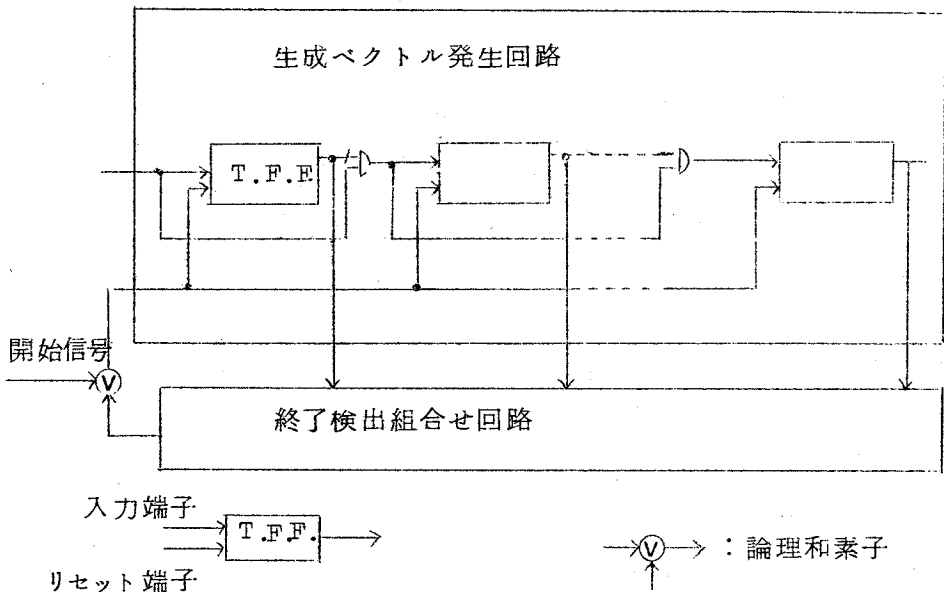


図 4-3 繰り返し系列の発生

このような組合せ回路を用いるとある系列を途中で中断させること、あるいは中止させることなども容易に行えるであろう。またもしこのような組合せ回路をまったく用いないと、 $i$  個の T.F.F. は時刻  $2^i + 1$  にはすべて  $S_i$  に戻るから、長さ  $2^i$  の全体の系列がそれ以後繰り返されることになる。

#### 4.4. 系列発生器の特徴

ここでは本章で考察した系列発生器の特徴について考察する。

前節の結果からあきらかなように、行列  $M_i$  のすべての生成ベクトルを発生させる回路があれば長さ  $2^i$  までの任意の系列を発生することができる。すなわち与えられた系列を発生させることは生成ベクトルの適当な部分集合を選び、それらの排他的論理和を作ることになる。

つぎに生成ベクトルのすべてを発生させる回路と排他的論理和のための回路を分離し、排他的論理和の回路（組合せ回路のみである）を付加回路として作ることにすると、この付加回路を取り替えることにより別の系列を発生させるより容易に変更できる。たとえば排他的論理和を作る付加回路として

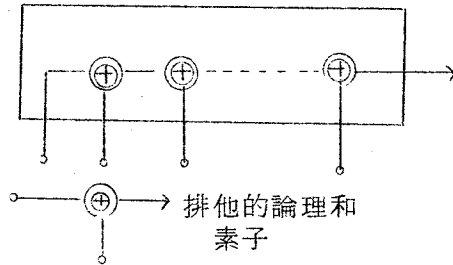


図 4-4 排他的論理和パッケージ

図 4-4 のようなパッケージを用意しておくと、発生したい系列に対する必要な生成ベクトルを操作 I で求め、そのおのおのをこのパッケージの入力端に接続してやればよい。このとき別の系列を発生させるにはその接続の仕方

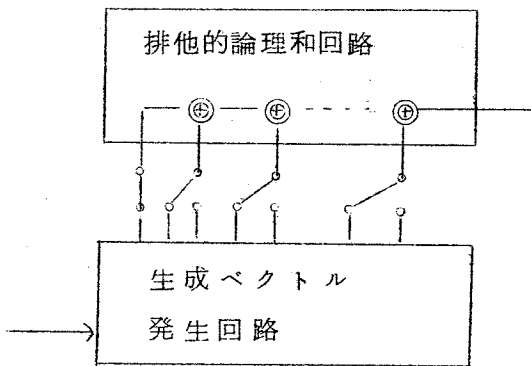


図 4-5 系列発生器の分割

を変更するだけでよい。(図 4-5)

またこのように生成ベクトルを作る回路と排他的論理和を作る回路とを分離したとき、各生成ベクトルが得られる出力端子の fan-out が十分大きければ、同時に異なった系列を数種類発生することができる。(図 4-6)

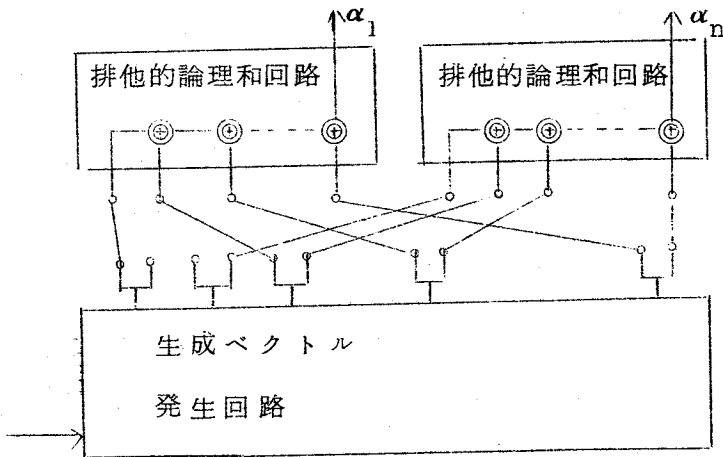


図 4-6 系列の同時発生

つぎにこの系列発生器の記憶素子については、最大長  $l$  の系列を発生させるのに  $\lceil \log_2 l \rceil$  個の記憶素子 (T.F.F.) を必要としているが、この数は長さ  $l$  の一つの 2 値系列だけを発生させる順序回路型の系列発生器が必要とする記憶素子の数に等しく、その意味でこの系列発生器は記憶素子の数に関して最少の回路である。

以上のことより、本章で考察した系列発生器の特徴はつぎのようにまとめられる。

- (1) 任意の系列を発生できる。
- (2) 付加回路の変更により別の系列を発生するようになやすく変更できる。
- (3) 同時に数種の系列を発生することができる。
- (4) 発生できる系列の長さを基準として、この回路の記憶素子は最少である。

なお与えられた系列を生成ベクトルに分解する操作 (操作 I) および基本生成ベクトルでない生成ベクトルを基本生成ベクトルに分解する操作 (操作 II-(2)) は容易にプログラム化された。

以上までに考察した系列は 2 値系列のみについてであったが、任意の多値系列は 2 値系列に分解できるので、ここに述べた系列発生器で多値系列を発



生することができる。とくにこの系列発生器では特徴(3)により同時に数種の系列を発生することができるため、一つの系列発生器で多値系列の発生が可能である。

つぎにこの系列発生器を用いて順序回路の故障検査器を構成することを考えてみる。故障検査は被検査順序回路に故障検査入力系列を加え、その入力系列に対してその順序回路が出す出力系列と故障検査系列の出力系列とが一致するかどうかを調べることによって行われる。したがって故障検査器は検査入力系列と検査出力系列とを同期させ同時に発生することが望まれる。ここで考察した系列発生器では特徴(3)によりこの要求を満たすことができる。また一つの順序回路だけでなく別の順序回路の検査をも一つの検査器で検査することができる都合がよい訳であるが、これは特徴(2)による発生系列の変更を回路的な切り換えでできるように検査器を構成しておけば容易に達成できる。組合せ回路に対する故障検査も同様の検査器で行えることは明らかであろう。したがってこの系列発生器は故障検査器の構成にとってかなり有用なものであると思われる。

なお本章で考察した系列発生器はどのような系列でも発生できるものであったが、オートノマスな順序回路を用いてある特定の系列のみを発生する系列発生器を構成することを考えるときには種々の方法が考えられる。たとえば長さ  $l$  の系列を発生するときには  $l$  まで数えるカウンタを用意し、入力として刻時パルスを数えるようにし、各時刻におけるカウンタの内容を組合せ回路で求める系列の各時刻における値に変換してやればよい。 $l$  まで数えるカウンタは  $\lceil \log_2 l \rceil$  個の2値素子を用いて構成されるから、この系列発生器が必要とする記憶素子数も  $\lceil \log_2 l \rceil$  となる。またある時刻における出力をフィードバックし、それを次の時刻に対する入力として用いたオートノマス順序回路でも与えられた系列を発生することができる。この順序回路ではもし発生しようとする長さ  $l$  の系列が各時刻において  $m$  種の値をとり、各値が同じ数だけその系列に現われるなら、必要とする状態数は  $l/m$  となる。

したがってこの順序回路を2値素子を用いて構成するなら、必要とする記憶素子数は  $\lceil \log_2 \frac{l}{m} \rceil + \lceil \log_2 m \rceil$  個となる。この他にも各種の系列発生器が考えられるであろうか、それらのうちどれを用いると最も有効であるかはそれぞれの立場によって決定される問題であろう。

#### 4.5. 結 言

本章では任意の系列を発生できること、付加回路をとりかえることにより異なる系列を発生できるより比較的簡単に変更できること、同時に数種の系列を発生できること、また記憶素子の数が最少である等の特徴をもった系列発生器について考察した。この系列発生器はこれらの特徴により、故障検査器の構成に有用なものであろうことが分った。

## 第5章 結 論

本論文で得られたおもな成果および今後に残された問題を簡単にまとめておく。

第2章では順序回路の故障検査を入力出力の対応関係のみから行う一つの方法として、与えられた順序回路を一意的に表現する最短長の入出力系列を計算機を用いて探索する方法について考察した。一般に一つの入出力系列がそれを満たす順序回路の状態遷移に課する制約はかなり厳しいもので、順序回路を一意的に表現する最短長の入出力系列は従来の観点からすると驚くほど短い場合が多いことが分った。しかしこの最短長の検査系列を求めることを目指す場合はある程度の総当り的な操作を含むことは避けられないため、ここで述べた方法は状態数の多い順序回路に対しては多くの計算時間を必要とし、あまり有効でないかも知れない。そこで順序回路の故障検査系列を求めることのみを考える場合には必ずしも最短長である必要はないから、第2章後半で述べたように近似的方法によって計算時間を短縮することが考えられる。このようにこの種の問題は検査系列の構成手順の簡単さと、検査系列の長さとの二つの要素を考慮して考えられるべきで、この両要求の妥協点をどこにおくかによって今後さらに種々の方法がなされるであろう。

第3章では入出力の対応関係による順序回路の故障検査問題を、その順序回路の構成の時から故障検査しやすいように考慮しておくという立場から考察した。ここでは故障検査時のみに使用されるあらたな入力を導入することを考えたが、これによって比較的短い故障検査系列が組織的に簡単な手順で構成できることが分った。この立場からの故障検査に関しては、回路にあらたにつけ加える冗長性をどのようにすると検査系列の構成が有利となるかということの他に、実際の回路構成にとっても有利となるような考慮がなされるべきであろう。

第4章では論理回路の故障検査時や、その他符号通信等でしばしば問題と

なるある定められた系列を発生させる系列発生器の構成について考察した。ここでは与えられた系列をいくつかの周期系列から合成する方法について述べたが、この系列発生器は任意の系列を発生できること、付加回路をとりかえることにより異なる系列を発生できるより簡単に変更できること、同時に数種の系列を発生できること、また記憶素子の数が最少であるなどの特長をもつことが分った。しかしこの系列発生器は回路的に少し複雑な嫌いがあるのでこの点の改良が望まれるであろう。

付録 I

(i) 「 $\alpha_n$ -系列を満足する順序回路の状態数は  $n$  以上である」ことの証明

$\alpha_n$ -系列はつぎの形をしている。

時刻	$t_1$	$t_2$	$t_3$	.....	$t_{n-1}$	$t_n$	$t_{n+1}$	.....	$t_{2n+1}$
入力系列	1	1	1	.....	1	1	1	.....	1
									(付-1)
順序回路 の状態	$S_{t_1}$	$S_{t_2}$	$S_{t_3}$	.....	$S_{t_{n-1}}$	$S_{t_n}$	$S_{t_{n+1}}$	.....	$S_{t_{2n+1}}$
出力系列	0	0	0	.....	0	1	0	.....	0

ある順序回路が  $\alpha_n$ -系列を満足したとすると、その順序回路の  $1 \leq t_i < t_j \leq n$  なる任意の 2 時刻  $t_i, t_j$  における状態について考える。この順序回路は状態  $S_{t_i}$  にあるときつぎの入出力系列

	長さ $(n-i+1)$									
入力系列	1 1 1 1 ..... 1 1									
	0 0 0 0 ..... 0 1									
										(付-2)

を満足するが、状態  $S_{t_j}$  にあるときはこの入出力系列 (付-2) を満足することができない。したがって  $S_{t_i}$  と  $S_{t_j}$  とは互に異なる状態である。すなわち  $n$  個の状態  $S_{t_1}, S_{t_2}, \dots, S_{t_n}$  は互に異なる状態であることになる。ゆえに、この順序回路の状態数は  $n$  以上である。

(ii) 「 $n$  状態の順序回路が  $\alpha_n$ -系列を満足するならこの順序回路は性質(I)

(II) を持つ」ことの証明

$S_{t_i}$  ( $i = 1, 2, \dots, n$ ) がすべて異なることと式(付-1)とから  $i = 1, 2, \dots, n-1$  に対しては  $T(S_{t_i}, 1) = S_{t_{i+1}}$ ,  $O(S_{t_i}, 1) = "0"$  が成立することが分る。…………… 性質(I)

また  $n$  状態の順序回路が  $\alpha_n$ -系列を満足するなら、式(付-1)において状態  $S_{t_{n+1}}$  は状態  $S_{t_1} \sim S_{t_n}$  のどれかに等しくなければならない。ところが状態  $S_{t_{n+1}}$  はつぎの入出力系列(付-3)を満足する状態であり、 $S_{t_1} \sim S_{t_n}$  の状態のうちで(付-3)を満足できるのは  $S_{t_1}$  だけである。

$$\begin{array}{rcc}
 & \text{長さ}(n-1) & \\
 \text{入力系列} & \overbrace{1 \quad 1 \quad 1 \quad \dots \quad 1} & \\
 & 0 \quad 0 \quad 0 \quad \dots \quad 0 & \text{(付-3)}
 \end{array}$$

したがって、 $n$  状態の順序回路が  $\alpha_n$ -系列を満足するとすれば、 $S_{t_{n+1}} = S_{t_1}$  とならねばならない。すなわち、 $T(S_{t_{n+1}}, 1) = S_{t_1}$ ,

$$O(S_{t_n}, 1) = "1" \quad \dots \quad \text{性質(II)}$$

そしてこのように  $S_{t_n}$  の状態遷移関数、出力関数を定めると、式(付-1)より  $S_{t_{n+1}} = S_{t_1}$  なることが要求されるが、このようにしても  $S_{t_{n+1}}$  の出力関数に矛盾は生じない。

付録 II

補題 4-1 の証明

(1) まず  $2^i$  次の単位行列を  $E_i$  で表わすとき、 $E_i$  の各行ベクトルは行列  $M_i$  の行ベクトルの排他的論理和によって表わされることを示す。 $i$  に関する帰納法で行う。

i)  $E_1$  の第 1 行は  $m_1^1$  そのものであり、第 2 行は  $m_1^1 \oplus m_2^1$  として表わせるので、 $i = 1$  に関して補題は成立する。

ii) 一般に  $E_{i-1}$  が  $M_{i-1}$  の行ベクトル間の排他的論理和により表わされる(行列  $M_{i-1}$  が行ベクトル間の排他的論理和の操作で対角化される)と仮定する。行列  $M_i$  は定義より、



が一意的であることを示す。

ある2元ベクトル  $\alpha$  がつぎのように2様に表わされたと仮定する。

$$\alpha = \sum_{j \in J_1} \oplus m_j^i, \quad \alpha = \sum_{j \in J_2} \oplus m_j^i, \quad J_1 \neq J_2$$

ここで  $\sum \oplus$  は排他的論理和を表わし、 $J_1, J_2$  はインデックスの集合を表わす。

$$\text{このとき } \alpha \oplus \alpha = 0 = \left( \sum_{j \in J_1} \oplus m_j^i \right) \oplus \left( \sum_{j \in J_1} \oplus m_j^i \right)$$

ここで  $0$  は  $2^i$  次元の零ベクトルを表わす

$J_1$  または  $J_2$  の要素でかつ  $J_1, J_2$  の共通の要素ではないものの集合を  $J'$  とすると上式右辺は  $\sum_{j \in J'} \oplus m_j^i$  となる。また仮定より  $J' \neq \emptyset$  であるから、 $J'$  の要素のうち最大のインデックスを  $k$  ( $1 \leq k \leq 2^i$ ) とすると、あきらかに  $\sum_{j \in J'} \oplus m_j^i$  はその第  $k$  要素に "1" をもつ。これは上式の形が  $0$  であることに矛盾する。ゆえに  $\alpha$  の表現は一意的である。(証明終り)

#### 補題 4-3 の証明

(1) 補題の前半を  $i$  についての数学的帰納法によって証明する。

(i)  $M_1$  については生成ベクトル  $m_1^1, m_2^1$  はともに基本生成ベクトルであるからあきらかに補題は成立する。

(ii) 一般に  $M_{i-1}$  のすべての生成ベクトルがその  $i$  個の基本生成ベクトルの適当なものの論理積で表わされていたと仮定する。

$M_i$  は定義より

$$M_i = \begin{bmatrix} M_{i-1} & 0_{i-1} \\ M_{i-1} & M_{i-1} \end{bmatrix} \quad \text{なる形をしているので、}$$

(a) 行列  $M_i$  の生成ベクトルのうち  $2^{i-1} < j \leq 2^i$  なる生成ベクトル  $m_j^i$  は  $M_{i-1}$  に対する帰納法の仮定より、 $M_i$  の基本生成ベクトル  $m_{(2^{i-2}j)}^i$  ( $j = 0.1 \dots \dots, 2^{i-2}$ ) および  $m_{2^i}^i$  の  $i$  個の適当なものの論理積として表わすことができる。(  $\because M_i$  の基本生成ベクトルのうち  $m_{(2^{i-2}j)}^i$  ( $j=0,$

$1, \dots, 2^{i-2}$ ) および  $m_{2^i}^i$  は  $M_{i-1}$  の  $i$  個の基本生成ベクトル  $m_{(2^{i-1} - 2^j)}^{i-1}$  ( $j = 0, 1, \dots, 2^{i-2}$ ) および  $m_{2^{i-1}}^{i-1}$  をそれぞれ 2 倍にひきのぼした形のベクトルである。) )

(b) 行列  $M_i$  の生成ベクトルのうち  $1 \leq j \leq 2^{i-1}$  なるすべての生成ベクトルは

$$m_j^i = m_{2^{i-1}}^i \cdot m_{(2^{i-1} + j)}^i$$

として表わせる。ここで右辺第一項は  $M_i$  の基本生成ベクトルであり、第二項は(a)により  $M_i$  の基本生成ベクトルの論理積として表わされている生成ベクトルである。

(i), (ii)より、すべての  $i$  について、行列  $M_i$  のすべての生成ベクトルは基本生成ベクトルの論理積として表わせる。

(2) つぎに各生成ベクトルの論理積による表現の仕方が一意的であることを示す。

行列  $M_i$  の基本生成ベクトルの個数は  $i + 1$  個である。このうち  $m_{2^i}^i$  なる基本生成ベクトルは  $2^i$  個のすべての成分が “1” であるので他の生成ベクトルの論理積による合成は関与しない。そこで実際他の生成ベクトルの合成に関与する基本生成ベクトルの数は  $i$  個である。一方  $m_{2^i}^i$  を除く生成ベクトルの個数は  $2^i - 1$  個であり、(1)ではそれらのすべてが基本生成ベクトルの組合せで作られることが示された。  $i$  個の基本生成ベクトルの組合せの仕方は  $\sum_{j=1}^i \binom{i}{j} = 2^i - 1$  通りであるから、基本生成ベクトルの組合せのおのおの、生成ベクトルのおのおのとは一対一の対応がなければならない。

(証明終り)



## 謝

## 辞

本研究の全過程を通じて、直接理解ある御指導を賜わり、つねに励ましていただいた尾崎弘教授ならびに樹下行三助教授に衷心より感謝の意を表す。

また基礎工学部嵩忠雄教授には適切な御指導、御討論をいただき、心から感謝する。

大学院修士、博士両課程において電子工学一般および各専門分野に関し御指導、御教示を賜わった電子工学教室菅田栄治教授、喜田村善一教授、宮脇一男教授、中井順吉教授、故寺田正純教授、電子ビーム研究施設裏克己教授産業科学研究所松尾幸人教授、中村勝吾教授に深謝する。

第2章、第3章における計算機利用に際し、御指導、御配慮いただいた城憲三名誉教授、牧之内三郎助教授ならびに安井裕講師、また第4章について適切な御教示、御助言いただいた基礎工学部都倉信樹助教授に厚く感謝する。

本研究に関し、基礎工学部保田豊助教授、電信電話公社電気通信研究所橋本昭洋博士、通産省電気試験所鳥居宏次博士、日本電気岡本光曜氏には本学大学院在学中に有益な御助言、御討論をいただき、心から謝意を表す。

また白川功講師、愛媛大学工学部有吉弘助教授ならびに大阪大学基礎工学部大学院学生谷口健一氏にはいろいろ御助言、御援助いただき、厚く御礼申し上げます。

筆者の属している尾崎研究室の谷口慶治助手、戸松重一技官、大学院学生高橋浩光氏、山本勝氏、細見輝政氏、岡本務氏、谷口尙氏、また同研究室の川端信賢氏、岡田真理氏には種々の面で御協力いただいた。ここに記して感謝する次第である。

## 参 考 文 献

- [ 1 ] E.E.Moore : " Gedanken-experiments of sequential machines", in Automata Studies, Princeton Univ. Press, P129 (1956)。
- [ 2 ] S.Seshu, D.N. Freeman: "The diagnosis of asynchronous sequential switching systems", IRE Trans., EC-11, P459 (Aug.1962)。
- [ 3 ] A. Gill : " Introduction to the theory of finite-state machines", Mc Graw-Hill (1962)。
- [ 4 ] 樹下行三: " 順序回路網の故障診断法に關する一考察", 信学誌, 46 9, P1211 ( 昭 38-09)。
- [ 5 ] J.F. Poage, E.J. Mc Cluskey: "Derivation of optimum test sequence for sequential machines", Proc. of the 5-th Ann. Sympo. on Switching Circuit Theory and Logical Design, Princeton N.J. (Oct.1964)。
- [ 6 ] F.C. Hennie: " Fault detecting experiments for sequential circuit", Proc.of the 5-th Ann. Sympo. on Switching Circuit Theory and Logical Design, Princeton N.J. (Oct. 1964)。
- [ 7 ] 福井徹、中西義郎: " 有限状態システムのテスト入力決定法", 信学誌, 50.10, P.1911 ( 昭 42-10)。
- [ 8 ] M.O. Rabin, Scott: "Finite automata and their decision problems", IBMJ., 3 (1959)。
- [ 9 ] S.W. Golomb, L.D. Baumert: "Backtrack programming", JACM, 12, 4, P.516 (Oct.1965)。
- [10] A.A.Tal: "Questionnaire language and the abstract synthesis of minimal sequential machines", Automatika i Telemekhanika, 25, NO.6, P.946, (June 1964)。

- [11] 村上伸一、樹下行三、尾崎弘：“入出力系列を満たす順序回路の構成法とその故障検査への応用について”、信学会オートマトンと自動制御、インホメーション理論研資、(昭41-09)。
- [12] 村上伸一、樹下行三、尾崎弘：“順序回路の故障検査系列の計算機による探索”、信学誌、50、10、P.1925 (昭42-10)。
- [13] J.P.Roth：“Diagnosis of automata failures:a Calculus and a method”，IBMJ. P278 (July 1966)。
- [14] J.P. Roth, W.G. Bouricius, P.R. Schneider：  
“ Programmed Algorithm. to compute tests to detect and distinguish between failures in logic circuit”  
IEEE Trans, EC-16, 5, P.567, (Oct. 1967)。
- [15] 橋本昭洋、嵩忠雄、尾崎弘：“組合せ論理回路の検査に関する一考察”、信学誌 47, 608 (昭39 )。
- [16] M.G. Paul, S.H. Unger, “Minimizing the number of states in incompletely specified sequential switching functions”, IRE Trans., EC-8, (Sept.1959)。
- [17] S. Ginsberg, “A technique for the reduction of a given machine to a minimal state machine”, IRE Trans., EC-8 (Sept. 1959)。
- [18] A. Grassell, F.Luccio, “ A method for minimizing the number of internal states in incompletely specified sequential network”, IEEE Trans., EC-14 (June, 1965)。
- [19] Z. Kohavi, P. Lavallee: “ Design of sequential machines with fault-detection Capabilites ”, IEEE Trans., EC-16, 4, P.473, (Aug. 1967)。
- [20] 村上伸一、樹下行三、尾崎弘：“故障検査を考慮した順序機械の構

- 成法”、信学誌、51、10、P.455 (昭43-10)。
- [21] S.W. Golomb: “Shift register sequences”, San Francisco, Calif. Holden Day, (1967)。
- [22] 村上伸一、都倉信樹、嵩忠雄、尾崎弘: “シフトレジスタ形回路による順序機械の実現について”、信学誌、49、12、P2462 (昭41-12)。
- [23] M. Perlman: “The synthesis of binary sequence detectors”, IEEE Trans., C-17 P.873, (Sept.1968)
- [24] 尾崎弘、樹下行三: “デジタル代数学”、共立出版(昭41)。
- [25] フィスター(尾崎弘訳): “デジタル計算機の論理設計”、朝倉書店(昭35)。
- [26] 村上伸一、樹下行三、尾崎弘: “定められた系列を発生させる方法について”、昭41-信学会全大。
- [27] W.W. Peterson, “Error-correcting codes”, the M.I.T. Press, and John Wiley & Sons, Inc, (1961)。
- [28] 村上伸一、樹下行三、尾崎弘: “系列発生器の構成に関する一考察” 信学論C (昭45-01) 掲載予定。
- [29] P.Weiner, J.E. Hopcroft: “Modular decomposition of synchronous sequential machines”, Conf. record of 1967 eighth Ann. Sympo. on Switching and Automata Theory P.233, (Oct.1967)。
- [30] 村上伸一、樹下行三、尾崎弘: “万能型順序回路について”、信学論C、52-C、4、P.240 (昭44-04)。
- [31] M. Cohn, S. Even, “Design of shift register generator for finite sequences”, IEEE Trans on Computers, C-18 P.660 (July 1969)。

- [32] F.C. Hennie: "Finite-state models for logical machines", John Wiley & Sons, Inc, New York, (1968)。
- [33] M.L. Minsky: "Computation: finite and infinite machines", Prentice-Hall, Inc, N.J. (1967)。
- [34] A. Gill: "Realization of input-output relations by sequential machines", JACM, vol.13, No.1, P.33, (Jan. 1966)。
- [35] 河田亨、樹下行三、尾崎弘: "シフトレジスタ形順序回路の故障検査について", 信学論C、52C、7、P.378(昭44-07)。
- [36] R.L. Martin: "The design of diagnosible sequential machines", Proc. of the Hawaii International Conf on System Sciences, Univ. of Hawaii Press, P.619 (1968)。
- [37] S.Murakami, K.Klnoshita, H.Ozaki, "Universal type binary sequence generator", Technology Reports of the Osaka University, vol.19, No.904, (Oct.1969)。

# 正誤表

誤

正

P. (5) ↓ 6      2.2 入出力…      2.2 与えられた入出力…

↓ 8      …… 検査系列      …… 検査系列の探索

P. 8 ↑ 13      …… 73の5行にして入出力…      …… 73の5行にして与えられた

入出力系列…

P. 13 ↑ 6      の図 2-31において STOP…      の図 2-31において (18) STOP…

P. 14 ↑ 10      …… 与えられた状態図を…      …… 与えられた順序回路が…

↑ 9      …… についてその      …… についてそれがその…

P 15 ↓ 6      ……  $d_R$       ……  $d_R$  を意味する。

P 23 ↑ 2      その出力 0 を…      その出力 0 を…

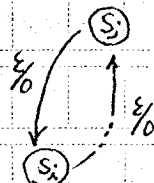
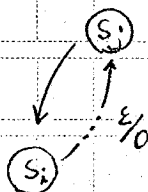
↑ 1      …… 出力 0 を…      …… 出力 0 を…

P 25 ↑ 4      …… 存在する。      …… 存在する)

P 29 ↓ 9      ……  $\begin{pmatrix} I_j \\ 0_j \end{pmatrix}$  ……      ……  $\begin{pmatrix} I_j \\ 0_j \end{pmatrix}$  ……

P 31 脚註      …… 入力系列…      …… 入出力系列…

P 37 図 3-5



P 46 ↑ 2       $V' = m_j^i \oplus \dots$

$V' = m_{g_1}^i \oplus \dots$

P 47 ↑ 2      …… 2 の 乗…

…… 2 の 冪乗…

誤

正

P 51 ↓ 1

--- ㄉ T.E.F. ---

--- ㄉ T.F.F. ---

P 64 ↓ 8

--- 裏克己 ---

--- 裏克巴 ---

↓ 13

--- 電信電話 ---

--- 日本電信電話 ---

P 65 ↓ 2

E. E. Moore ---

E. F. Moore ---

↑ 13

--- Ann. ---

--- Ann. ---

P 66 ↑ 3

macbine ---

machine ---

P 67 ↑ 7

--- Ann ---

--- Ann. ---

P 68 ↑ 3

--- K1oshita ---

--- Kinoshita ---

↑ 3

--- Universal type

--- Universal-type

↑ 2

--- Teclmology ---

--- Technology ---