Easily Testable Sequential Machines

By

Hideo FUJIWARA and Kozo KINOSHITA

Reprinted from

TECHNOLOGY REPORTS OF THE OSAKA UNIVERSITY

Vol. 24 No. 1214

Faculty of Engineering

Osaka University

Osaka Japan

1974

No.1214

(Received March 15, 1974)

Easily Testable Sequential Machines

By

Hideo FUJIWARA and Kozo KINOSHITA

(Department of Electronic Engineering)

Abstract

In this paper, two types of easily testable machines are introduced, the state-shiftable machine and the output-observable machine. Design procedures are presented in which an arbitrary machine is augmented to these easily testable machines by adding extra inputs or outputs. Efficient procedures are also described for designing checking experiments for such machines.

1. Introduction

For sequential machines several authors[1)−6)] have considered the fault detection problem as an identification problem of sequential machines, that is, finding an input-output sequence which describes a given machine uniquely. A number of these papers are based on a method given by Hennie[2)] for designing checking experiments, called *the transition checking approach*. His method yields good results for machines that possess a distinguishing sequence, and for machines that are reduced, strongly connected, and such that the actual machine has no more states than the correctly operating machine. However, for machines which do not have any distinguishing sequences, Hennie's procedure yields very long experiments, which makes it impractical. Therefore, several methods have been proposed of modifying a given sequential machine into a new one for which a short checking experiment can easily be found[3),7)−13)]. These include 1) a method of adding extra outputs[7),8)] and 2) a method of adding extra inputs[9)−11)]. For an n-state m-input symbol machine, the former gives a bound on the length of checking experiments that is approximately mn^3, and the latter gives a bound of mn^2.

In this paper, two types of easily testable machines are introduced, the *state-shiftable machine* and the *output-observable machine*. First half of this paper describes a design procedure in which an arbitrary machine is augmented to a state-shiftable machine by adding two special input symbols to the original machine. An efficient procedure is also described for designing checking experiments for the state-shiftable machines. For an n-state m-input symbol machine, this procedure gives a bound on the length of checking experiments that is approximately $mn[\log_2 n]$, where the square brackets denote "the smallest integer greater than or equal to the number inside the brackets".

The second half of this paper presents a design procedure in which an arbitrary machine is augmented to an output-observable machine by adding a minimum number of extra outputs. For the k-output-observable machines, an input-output sequence $\omega_1\omega_2$, such that ω_1 is an input-output sequence which passes through all the transitions of the given state table and ω_2 is an arbitrary inout-output sequence of length k, can be shown to be a checking experiment, and hence nearly minimum length checking experiments are obtained.

2. State-Shiftable Machines

The sequential machines considered in this paper are assumed to be finite state, synchronous, and deterministic Mealy machines, and don't require to be reduced, strongly connected, or completely specified. The machine M will be represented by a quintuple $M = (S,I,0,\delta,\lambda,)$ where $S = \{S_1,S_2,\ldots,S_n\}$ is a finite set of states, $I = \{I_1,I_2,\ldots,I_m\}$ is a finite set of input symbols, $0 = \{0_1,0_2,\ldots,0_l\}$ is a finite set of output symbols, $\delta : S\times I \rightarrow S$ is called the next state function, and $\lambda : S\times I \rightarrow 0$ is called the output function.

Definition 1: A sequential machine M is called *state-shiftable* if M contains a 2-column submachine isomorphic to a binary shift register.

Consider a p-stage binary shift register in Fig. 1. Let $Y_1, Y_2, \ldots, Y_p$ be the state

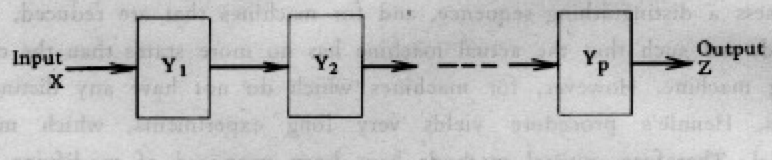

Fig. 1 The p-stage binary shift register

variables. let X be the input variable and let Z be the output variable. For the p-stage binary shift register, a p-tuple state assignment $Y_1Y_2\ldots Y_p$ can be found for each state such that

1) $Y_i(t+1) = Y_{i-1}(t)$ for $i=2,3,\ldots,p$,
2) $Y_1(t+1) = X(t)$,

and

3) $Z(t) = Y_p(t)$,

where $Y_1(t)$, $Y_2(t),\ldots,Y_p(t)$, $X(t)$, and $Z(t)$ are the values of $Y_1, Y_2,\ldots,Y_p$, X, and Z at time t, respectively.

Then it is easily seen that any input sequence of length p is both a distinguishing sequence and a synchronizing sequence, and that $Y_pY_{p-1}\ldots Y_1$ is a transfer sequence of length p to carry the p-stage binary shift register to state S_i with state assignment $Y_1Y_2 \dots Y_p$.

Hence, we have the following theorem.

Theorem 1: An n-state state-shiftable machine possesses 1) a distinguishing sequence of length $[\log_2 n]$ which is also a synchronizing sequence, and 2) for each state S_i, a transfer sequence of length at most $[\log_2 n]$ which transfers the machine from an arbitrary state to state S_i.

Let $M = (S,I,0,\delta,\lambda)$ be a given machine, where $S = \{S_1,S_2,\dots,S_n\}$, $I = \{I_1,I_2,\dots,I_m\}$, and $0 = \{0_1,0_2,\dots,0_l\}$. Then we can give a procedure for augmenting the given machine M so that the augmented machine M* is state-shiftable.

Augmentation Procedure:

1) Add new states $S_{n+1},S_{n+2},\dots,S_{n'}$ to M if n is not an integral power of 2, where $n' = 2^p$ and $p = [\log_2 n]$.
2) Assign the p-bit binary codes to all states such that each state has only one assignment.
3) Add new input symbols ϵ_0, ϵ_1, to M. The next state function δ and the output function λ for the new input symbols ϵ_0, ϵ_1 are defined as:

   For each state S_i, with state assignment $Y_1Y_2\dots Y_p$, $\delta(S_i,\epsilon_0) = S_j$ and $\delta(S_i,\epsilon_1) = S_k$, and $\lambda(S_i,\epsilon_0) = \lambda(S_i,\epsilon_1) = O_1$ if $Y_p = 0$

   $= O_2$ if $Y_p = 1$

   where S_j and S_k have state assignment $0Y_1Y_2\dots Y_{p-1}$ and $1Y_1Y_2\dots Y_{p-1}$, respectively.

The effect of this state transition is to shift the state assignment one digit to the right and introduce a zero or a one as new left most digit according to input ϵ_0 or ϵ_1, respectively. Thus, this 2-column submachine restricted to inputs ϵ_0,ϵ_1 is isomorphic to the p-stage binary shift register. Hence the augmented machine M* is state-shiftable.

Example: Consider machine A given by Table 1. Machine A is not strongly connected and has not any distinguishing sequence. By applying the above procedure, we obtain the augmented machine A* shown in Table 2. A* has a distinguishing sequence $\epsilon_0\epsilon_0$ which also a synchronizing sequence whose final state is S_1. Transfer sequences are shown in Table 3.

Table 1 Machine A

state \ input	0	1
S_1	S_2 (1)	S_1 (1)
S_2	——	S_3 (0)
S_3	S_2 (0)	—(1)

The dash means "don't-care."

Table 2 Augmented machine A*

state \ input		0	1	ϵ_0	ϵ_1
00	S_1	S_2 (1)	S_1 (1)	S_1 (0)	S_3 (0)
01	S_2	——	S_3 (0)	S_1 (1)	S_3 (1)
10	S_3	S_2 (0)	— (1)	S_2 (0)	S_4 (0)
11	S_4	——	——	S_2 (1)	S_4 (1)

Table 3 Transfer sequences T(i) for machine A*

T(1)	T(2)	T(3)	T(4)
Λ	$\epsilon_1\epsilon_0$	ϵ_1	$\epsilon_1\epsilon_1$

" Λ " means the null sequence.

3. Checking Experiments for State-Shiftable Machines

In this section we consider checking experiments for the state-shiftable machines. The principle of our method is mainly based on those of Hennie[2] and Hsieh[5], and we assume that readers are familiar with the principle of those methods. Assume that the class of allowable failures satisfies the following conditions:

1) Any failure which occurs is assumed to occur throughout the test.
2) Failures don't increase the number of states.

Let $M=(S,I,0,\delta,\lambda)$ be an n-state m-input state-shiftable machine. Let X_d be an input sequence of length $[\log_2 n]$ which is both a distinguishing sequence and a synchronizing sequence. Let S_l be the final state resulting from the application of X_d. The transfer sequence of length at most $[\log_2 n]$ to move M from state S_l to state S_i is denoted by T(i).

The checking experiment consists of two parts. The first part of the experiment verifies that X_d is both a distinguishing sequence and a synchronizing sequence, and that T(i) transfers the machine from state S_1 to S_i, and have the form:

Input:	X_d	T(i)	X_d	X_d	
State:	–	S_1	S_i	S_1	S_1
Output:	–	Z_{li}	Z_i	Z_l	

for all states S_i.

The second part of the experiment is to be designed to check all the transitions and have the form:

Input:	X_d	T(i)	I_j	X_d	
State:	–	S_1	S_i	$S_{ij} = \delta(S_i, I_j)$	S_1

Output: $-$ Z_{1i} $O_{ij} = \lambda(S_i, I_j)$ Z_{ij}

for all states S_i and inputs I_j.

Then we have the following checking experiment:

Input: X_d $T(1)$ X_dX_d $T(2)$ $X_dX_d \ldots X_d$ $T(n)$ X_dX_d

State: $-$ S_1 S_1 S_1 S_2 S_1 S_n S_1

$T(1)$ I_1X_d $T(1)$ $I_2X_d \ldots X_d$ $T(i)$ $I_jX_d \ldots X_d$ $T(n)$ I_mX_d

S_1 S_1 S_1 S_1 S_1 S_i S_1 S_n

In this checking experiment, the initializing part is preset, and hence the total checking experiment is preset, and thus is easy to be applied to the tested machine.

Let us derive the bound on the length of the checking experiment. Since the machine M is assumed to be a state-shiftable machine, $|X_d| = [\log_2 n]$ and $|T(i)| \leq [\log_2 n]$ for $i = 1,2,\ldots,n$, where $|X|$ is the length of X.

From the organization of the checking experiment, it can be seen that the total length of the checking experiment is at most

$$|X_d| + \sum_{i=1}^{n}\left(|T(i)| + 2|X_d|\right) + \sum_{j=1}^{m}\sum_{i=1}^{n}\left(|T(i)| + |I_j| + |X_d|\right)$$

$$= (2n+1)|X_d| + \sum_{i=1}^{n}|T(i)| + mn\left(|X_d| + 1\right) + m\sum_{i=1}^{n}|T(i)|$$

$$\leq (2n+1)[\log_2 n] + n[\log_2 n] + mn([\log_2 n] + 1) + mn[\log_2 n]$$

$$= (3n+1)[\log_2 n] + mn(2[\log_2 n] + 1) \simeq mn[\log_2 n]$$

Namely, the order of its length is $mn[\log_2 n]$ which is smaller than the best order mn^2 obtained in the previous methods[7)–11)].

Example: Let us construct a checking experiment for machine A^* given by Table 2. $X_d = \epsilon_0\epsilon_0$ is both a distinguishing sequence and a synchronizing sequence whose final state is S_1. Transfer sequences $T(i)$ from state S_1 to each state S_i are shown in Table 3.

The total checking experiment is:

$\epsilon_0\epsilon_0 T(1)\epsilon_0\epsilon_0\epsilon_0\epsilon_0 T(2)\epsilon_0\epsilon_0\epsilon_0\epsilon_0 T(3)\epsilon_0\epsilon_0\epsilon_0\epsilon_0 T(4)\epsilon_0\epsilon_0\epsilon_0\epsilon_0 T(1)0\epsilon_0\epsilon_0$
$T(1)1\epsilon_0\epsilon_0 T(1)\epsilon_0\epsilon_0\epsilon_0 T(1)\epsilon_1\epsilon_0\epsilon_0 T(2)1\epsilon_0\epsilon_0 T(2)\epsilon_0\epsilon_0\epsilon_0 T(2)\epsilon_1\epsilon_0\epsilon_0 T(3)$
$0\epsilon_0\epsilon_0 T(3)1\epsilon_0\epsilon_0 T(3)\epsilon_0\epsilon_0\epsilon_0 T(3)\epsilon_1\epsilon_0\epsilon_0 T(4)\epsilon_0\epsilon_0\epsilon_0 T(4)\epsilon_1\epsilon_0\epsilon_0$

4. Output-Obvervable Machines

The sequential machines considered in this section are assumed to be reduced and strongly connected Mealy machines, such that binary codes are already assigned to their output symbols, i.e., the output function is represented, by a direct product, $z_1 \times z_2 \times \ldots \times z_p$, of binary output functions $z_1 \ldots, z_p$.

Definition 2: A partition on a set of states S is a collection of disjoint subsets of S, called blocks such that their set union is S. A relation $\equiv (\pi)$ on S corresponding to a partition π is a relation such that $S_i \equiv S_j\ (\pi)$ for $S_i, S_j \epsilon S$ if and only if S_i and S_j belong to the same block of π.

Definition 3[14]**:** The *transition graph* of a partition π is a graph in which each vertex corresponds to a block of π and there is an arc from vertex v_i to vertex v_j if and only if there is a state $S_k \epsilon B_i$ (B_i is the block of corresponding to vertex v_i) and an input I_l such that $\delta(S_k, I_l) = S_m\ \epsilon B_j$. A partition π is a *shift register partition* (SRP) if and only if the transition graph of π is a subgraph of some Good's diagram. π has length of ℓ if it is a subgraph of the Good's diagram of an ℓ-stage shift register.

Table 4 Machine B

state \ input	0	1
S_1	S_4 (0)	S_2 (0)
S_2	S_3 (0)	S_2 (0)
S_3	S_4 (1)	S_3 (1)
S_4	S_5 (1)	S_5 (1)
S_5	S_5 (1)	S_1 (1)

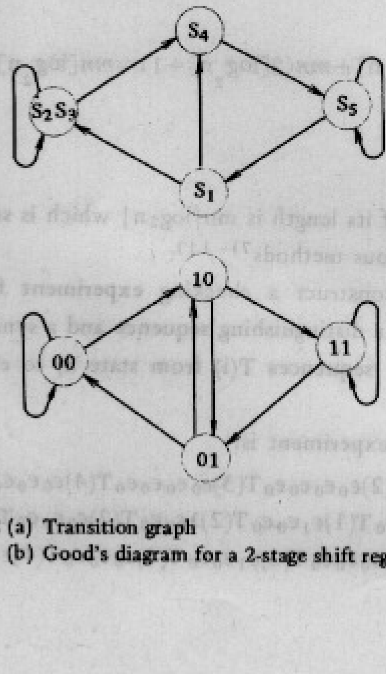

Fig. 2 (a) Transition graph
(b) Good's diagram for a 2-stage shift register

Example: Consider machine B given by Table 4 and a partition $\pi = [\overline{S_1}\ ;\ \overline{S_2S_3}\ ;\ \overline{S_4};\ \overline{S_5}]$. We obtain the transition graph shown in Fig. 2 (a). This transition graph is a subgraph of the Good's diagram of a 2-stage shift register shown in Fig. 2 (b). Therefore, the partition $\pi = [\overline{S_1}\ ;\ \overline{S_2S_3}\ ;\ \overline{S_4}\ ;\ \overline{S_5}]$ is an SRP.

Definition 4: A sequential machine M is called $k_1,k_2 \ldots, k_p$*-output-obvervable with respect to the output function* $z_1 \times z_2 \ldots \times z_p$ *and a partition* π if the following conditions are satisfied:

1) The knowledge of the present state of M is sufficient to uniquely determine the succeeding output sequence of length k_j observed at the output function z_j for every j ($1 \leq j \leq p$).
2) Let μ_{ij} be the output sequence of length k_j observed at z_j when the initial state is S_i. Then $S_i \equiv S_j\ (\pi)$ if and only if $(\mu_{i1}, \ldots, \mu_{ip}) = (\mu_{ij}, \ldots, \mu_{jp})$ for all S_i and $S_j \in S$.

When π is the zero partition, M is called *output-observable.*

Example: A sequential machine B shown in Table 4 is 1-output-observable with respect to z_1 and $\pi_1 = [\overline{S_1S_2}\ ;\ \overline{S_3S_4S_5}]$. A sequential machine B* shown in Table 6 is 1,2-output-observable with respect to $z_1 \times z_2$ and the zero partition, and thus B* is output-observable.

For a given machine M with a binary output function z, we can find a minimum partition π and a minimum integer k such that the machine M is k-output-observable with respect to z and π. This method is shown in the following.

Procedure A:

1) Set $\pi(0) = \mathbf{1}$ and $\ell = 1$.
2) For every state S_i, test whether all the output sequences of length ℓ observed at the output function z with the machine M initially in state S_i are the same. If "no" for some state S_i, set $\pi = \pi(\ell-1)$ and $k = \ell-1$, and stop. If "yes" for all states, then define a relation $\equiv (\pi(\ell))$ such that $S_i \equiv S_j\ (\pi(\ell))$ if and only if $\mu_i(\ell) = \mu_j(\ell)$, where $\mu_i(\ell)$ is the output sequence of length ℓ corresponding to state S_i.
3) If $\pi(\ell) = \mathbf{0}$, then set $\pi = \mathbf{0}$ and $k = \ell$, and stop. If $\pi(\ell) = \pi(\ell-1)$, then set $\pi = \pi(\ell)$ and $k = \ell-1$, and stop. Otherwise, set $\ell = \ell + 1$ and go to step 2).

Suppose that, for a given machine M, π_i and k_i ($1 \leq i \leq p$) have been obtained by means of Procedure A, then M is k_i-output-observable with respect to the output function z_i and the partition π_i for each i ($1 \leq i \leq p$). If $\pi_1\pi_2 \ldots \pi_p = \mathbf{0}$, then M is output-observable. If $\pi_1\pi_2 \ldots > \mathbf{0}$, then we have the following theorem.

Theorem 2[13]: The necessary and sufficient condition for a sequential machine M to be modified by adding a binary output functions $w_1, w_2 \ldots, w_s$ so that it will be $k_1,k_2, \ldots, k_p, l_1, l_2, \ldots, l_s$-output-observable with respect to the output function $z_1 \times z_2 \times \ldots \times z_p \times w_1 \times w_2 \times \ldots \times w_s$ is that there exist a SRP's $\tau_1, \tau_2, \ldots, \tau_s$ of length $\ell_1, \ell_2, \ldots, \ell_s$, respectively, such that $\pi_1\pi_2 \ldots \pi_p\tau_1\tau_2 \ldots \tau_s = \mathbf{0}$.

Therorem 2 shows that if we can find the least possible number of SRP's

$\tau_1, \tau_2, \ldots, \tau_s$ such that $\pi_1 \pi_2 \ldots \pi_p \tau_1 \tau_2 \ldots \tau_s = 0$, then we can modify the machine M to an output-observable one by adding a minimum number of extra outputs. the problem of generating all the SRP's for a given machine has been investigated by Nichols[14].

Suppose that we have obtained the least number of SRP's $\tau_1, \tau_2, \ldots, \tau_s$ satisfying the condition of Theorem 2. Then, we can construct binary output functions w_j ($1 \leq j \leq s$) satisfying the condition of Theorem 2 as follows: Let $Y_{j1}, Y_{j2}, \ldots, Y_{j\ell_j}$ be the state assignment variables of the ℓ_j-stage shift register corresponding to SRP τ_j, and let $(y^i_{j1}, y^i_{j2}, \ldots, y^i_{j\ell_j})$ be a binary code corresponding to state S_i. Note that each state is given a single coding. Define a binary output function w_j, such that $w_j(S_i) = y^i_{j\ell_j}$ for state S_i.

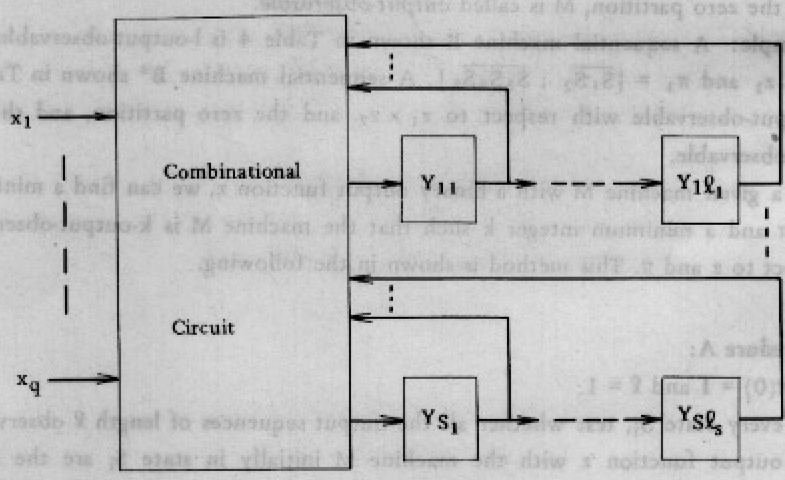

Fig. 3 Feedback shift register circuit

Summarizing this argument, we can present the following procedure for modifying a given machine so that it will be output-observable by adding a minimum number of extra outputs.

Augmentation Procedure:

1) Given a sequential machine M having binary output functions $z_1, z_2, \ldots, z_p$, find a minimum partition π_i and k_i for each z_i ($1 \leq i \leq p$) by means of Procedure A.

2) Set s = 1.

3) Test whether there exists SRP's $\tau_1, \tau_2, \ldots, \tau_s$, such that
$\pi_1 \pi_2 \ldots \pi_p \tau_1 \tau_2 \ldots \tau_s = 0$.
If "yes", then go to step 4).
If "no", then set s = s + 1, and repeat step 3).

4) Let $Y_{j1}, Y_{j2}, \ldots, Y_{j\ell_j}$ be the state assignment variables of the ℓ_j-stage shift register corresponding to SRP τ_j ($1 \leq j \leq s$). and let $(y^i_{j1}, \ldots, y^i_{j\ell_j})$ be a binary code

corresponding to state S_i. Add binary output functions $w_j(1 \leq j \leq s)$ to M, such that $w_j(S_i) = y_j^i \ell_j$ for each state S_i.

Example: Consider machine B given by Table 4 which is not output-observable. Let us modify machine B to an output-observable machine. The determination of a minimum number of additional output function is shown below, where each step is indicated by the corresponding number.

1) Applying Procedure A, we can obtain $k_1 = 1$, and $\pi_1 = [\overline{S_1S_2} ; \overline{S_3S_4S_5}]$.

2) $s = 1$.

3) Testing whether there exists an SRP τ_1 such that $\pi_1\tau_1 = \mathbf{0}$, we can find an SRP $\tau_1 = [\overline{S_1} ; \overline{S_2S_3} ; \overline{S_4} ; \overline{S_5}]$.. Indeed, $\pi_1\tau_1 = [\overline{S_1} ; \overline{S_2S_3} ; \overline{S_4} ; \overline{S_5}] \cdot [\overline{S_1S_2} ; \overline{S_3S_4S_5}] = [\overline{S_1} ; \overline{S_2} ; \overline{S_3} ; \overline{S_4} ; \overline{S_5}]$ = the zero partition. The transition graph of τ_1 is a subgraph of the Good's diagram for a 2-stage shift register shown in Fig. 2 (b). By giving a unique coding to each state in accordance with the labeling of the corresponding states in the Good's diagram, we can obtain a state assignment shown in Table 5.

4) By adding output function z_2 such that $z_2 = Y_2$, we can obtain the augmented machine B* shown in TAble 6 which is 1,2-output-observable with respect to $z_1 \times z_2$ and the zero partition.

Table 5 State assignment

	Y_1	Y_2
S_1	0	1
S_2	0	0
S_3	0	0
S_4	1	0
S_5	1	1

Table 6 Augmented machine B*

state \ input	0	1
S_1	S_4 (0 1)	S_2 (0 1)
S_2	S_3 (0 0)	S_2 (0 0)
S_3	S_4 (1 0)	S_3 (1 0)
S_4	S_5 (1 0)	S_5 (1 0)
S_5	S_5 (1 1)	S_1 (1 1)

5. Checking Experiments for Output-observable Machines

In this section we consider fault detection experiments for $k_1, k_2 \ldots, k_p$ -output-observable machines. Let M be the fault-free machine with the output function

$z_1 \times z_2 \times \ldots \times z_p$ and let M′ be the tested (possibly faulty) machine with the output function $z'_1 \times z'_2 \times \ldots \times z'_p$. Assume that the class of allowable failures satisfies the following conditions.

1) Any failure which occurs is assumed to occur throughout the test.
2) A failure which increases the number of states in the machine does not occur.
3) A faulty machine M′ is still $k_1, k_2, \ldots, k_p$-output-observable with respect to $z'_1 \times z'_2 \times \ldots \times z'_p$ and some partition π, i.e., the knowledge of the present state of M′ is sufficient to uniquely determine the succeeding output sequence of length k_i observed at the output function z'_i for all i ($1 \leq i \leq p$).

Under these assumptions, let us design a checking experiment. Given a $k_1, k_2, \ldots, k_p$-output-observable macnine M, let ω_1 be an input-output sequence that passes through all the transitions of the state table of M, and let ω_2 be an arbitrary input-output sequence of length k, where $k = \max\{k_1, k_2, \ldots, k_p\}$. It will be proved in the following theorem that the input-output sequence $\omega_1 \omega_2$, called C-sequence, is a checking sequence.

Theorem 3[13]: Let M be an output-observable machine. Then a machine satisfying the C-sequence of M is isomorphic to M.

Theorem 2 implies that only the correctly operating machine satisfies the C-sequence of M. However the converse is not always true, i.e., the correctly operating machine does not always satisfy the C-sequence when the machine under test is not initially in the starting state of the C-sequence of M. So the machine under test should be initially in the starting state of the C-sequence when the C-sequence is to be applied. This can be done by applying a homing sequence. For $k_1, k_2, \ldots, k_p$-output-observable machine, any input sequence of length k ($k = \max\{k_1, k_2, \ldots, k_p\}$) is a homing sequence.

Example: Consider machine B*, given by Table 6, which is 1,2-output-observable with respect to output function $z_1 \times z_2$ and the zero partition. By applying an arbitrary input sequence of length 1 and 2 at the output terminals z_1 and z_2, respectively, we can establish the initial state and the final state. Suppose that the machine is in state S_1, then the shortest input-output sequence ω_1, that passes through all the transitions of B*, is obtained as follows;

Input:	0	0	0	1	1	1	0	1	0	1	
State:	S_1	S_4	S_5	S_5	S_1	S_2	S_2	S_3	S_3	S_4	S_5
Output:	0	1	1	1	0	0	0	1	1	1	
	1	0	1	1	1	0	0	0	0	0	

As the final state is S_5, the following sequence is an input-output sequence ω_2 of length 2 starting at the state S_5:

Input:	0	0	
State:	S_5	S_5	S_5
Output:	1	1	
	1	1	

Then a checking experiment for machine B* is the following:

Input:		0	0	0	1	1	1	0	1	0	1	0	0
State:	S_1	S_4	S_5	S_5	S_1	S_2	S_2	S_3	S_3	S_4	S_5	S_5	S_5
Output:		0	1	1	1	0	0	0	1	1	1	1	1
		1	0	1	1	1	0	0	0	0	0	1	1

Acknowledgement

The authors wish to acknowledge the support and encouragement of Prof. H. Ozaki of Osaka University.

References

1) A. Gill, "Introduction to the Theory of Finite-State Machines," McGraw-hill, 1962.
2) F.C. Hennie, Proc. 5th Ann. Symp. Switching Theory and Logical Design, 1964.
3) C.R. Kime, Dep. Elec. Eng., Univ. of Iowa City, Tech. Rep. 66-13, 1966.
4) G.Gönenc, IEEE Trans. Comput. (Short Notes), vol. c-19, 1970. pp. 551-558.
5) E.P.Hsieh, IEEE Trans. Comput., vol. C-20, 1971, pp.1152-1166.
6) D.E.Farmer, IEEE Trans. Comput., vol. C-22, 1973, pp.159-167.
7) Z.Kohavi and P.Lavallee, IEEE Trans. Electron. Comput., Vol. EC-16, 1967, pp.473-484.
8) R.L.Martin, Proc. Hawaii Internat. Conf. Syst. Sci., 1968.
9) S.Murakami, K.Kinoshita and H.Ozaki, IEEE Trans. Comput., vol. C-19, 1970, pp. 1079-1085.
10) J.R.Kane and S.S.Yau, Conf. Rec. 12th Ann. Sympo. on Switching and Automata Theory, 1971, pp. 38-42.
11) C.E.Holborow, IEEE Trans. Comput. (Short Notes), vol. C-21, 1972, pp. 597-598.
12) M.J.Y. Williams and J.B. Angell, IEEE Trans. Comput., vol. C-22, 1973, pp. 46-60.
13) H.Fujiwara and K.Kinoshita, IEEE Trans. Comput., vol. C-23, 1974, pp. 138-145.
14) A.J.Nichols, IEEE Trans. Electron. Comput., vol.EC-14, 1965, pp. 688-700.