One Resettable State Variable Is Almost Sufficient for Synthesizing Synchronizable FSMs

Toshimitsu Masuzawa, Tomoo Inoue, Hiroshi Youra, and Hideo Fujiwara

Graduate School of Information Science
Nara Institute of Science and Technology
Ikoma, Nara 630-01, Japan
e-mail: masuzawa@is.aist-nara.ac.jp

Abstract

The great difficulty in sequential circuit test generation is partly caused by the fact that the initial state of the circuit is unknown. Thus, it is important to synthesize a *synchronizable* finite state machine (FSM) that can reach a specified final state by an input symbol sequence, regardless of the initial state. In this paper, we consider problems concerning synthesis of synchronizable FSMs with scannable/resettable state variables.

A state variable is called *scannable* (resp. *resettable*), if its value can be set to arbitrary value (resp. a predetermined value) by an extra input symbol. Thus, every FSM is obviously synchronizable, when all state variables are resettable. On the other hand, there exists a FSM that is not synchronizable without scannable/resettable state variables. Thus, it is interesting to determine the minimum number of the scannable/resettable state variables necessary to make the FSM synchronizable. This is our main concern in this paper.

We newly define two minimization problems concerning the number of the scannable/resettable state variables necessary to make a FSM synchronizable. We first consider the problems for a FSM that has only *a single input symbol*, and show the *exact* solutions.

We also consider the problems for a FSM with *multiple input symbols*. From the solutions for the case of a single input symbol, we obtain a sufficient condition for a FSM with multiple input symbols to be synchronizable. Since the sufficient condition is easily satisfied even for the case of only one resettable state variable, it seems that practical FSMs satisfy the condition. Actually, all MCNC'91 benchmark FSMs satisfy the condition and are found to be synchronizable with only one resettable state variable. Thus, it seems reasonable to conclude that *practical FSMs are synchronizable with only one resettable state variable.* This is the main contribution of this paper.

Furthermore, we consider the length of synchronizing sequence and show experimental results for MCNC'91 benchmark FSMs. The experimental results give considerable evidence to show that we can obtain a *short* synchronizing sequence even if we consider only a single input symbol of a FSM.

1 Introduction

Test generation for sequential circuits is much more difficult than that for combinational circuits. One of the main reasons for the great difficulty is that the initial state of the sequential circuit is unknown[4]. Thus, it is important to synthesize sequential circuits that have *synchronizing sequences*, where a synchronizing sequence[8] is an input sequence that takes a circuit to a specified final state, regardless of the initial state. In this paper, we consider problems concerning synthesis of such sequential circuits.

Most popular and simple approach is *full reset* design where every flip-flop is designed to be *resettable* (i.e., its content can be reset to a predetermined value via an extra primary port). By resetting all flip-flops, we can drive the sequential circuit to the state where every flip-flop stores the predetermined value. Thus, the circuit has a synchronizing sequence of length one, while large hardware overhead is required to feed a reset signal to every flip-flop. In *Partial scan* design, some of the flip-flops are scannable (i.e., their contents can be set to arbitrary values via an extra primary port). It is known that the partial scan design reduces the test generation cost with small overhead[1, 3]

A sequential circuit is modeled by a finite state machine (FSM), where a flip-flop of the circuit is modeled by a state variable of the FSM. When a flip-flop is scannable (resp. resettable), we say that the corresponding state variable is *scannable* (resp. *resettable*).

Cheng and Agrawal[2] show the conditions for initializability of FSMs and present a state assignment algorithm for logic minimality and initializability. Jiang *et al.*[7] point out the effectiveness of partial scan in synthesis of FSMs having short synchronizing sequences, and present a heuristic method for synthesizing such FSMs. However, they consider only synchronizing sequences where partial scan is applied *first* and *only once*. Pomeranz and Reddy[9] consider the effect of repeating partial reset on test generation. To consider the effect of repeating partial scan in synthesis of FSMs having short synchronizing sequences, Inoue *et al.*[6] introduce an *extended synchronizing sequence* where partial scan can be applied *several times*. They also present a heuristic method for synthesizing such FSMs.

In this paper, we consider the extended synchronizing sequence. Every FSM obviously has an extended synchronizing sequence (of length one), when all state variables are resettable. On the other hand, there exists a FSM that is not synchronizable without scannable/resettable state variables. Thus, it is interesting to determine the minimum number of the scannable/resettable state variables necessary to make the FSM synchronizable. This is our main concern in this paper.

We formulate the extended synchronizing sequence, and newly introduce two minimization problems concerning the number of the scannable/resettable state variables necessary to make a

FSM synchronizable. We first consider the problems for FSMs that have only *a single input symbol*, and show the *exact* solutions. In other words, we show the necessary and sufficient condition for such FSMs to be synchronizable when the number of scannable/resettable state variables is given.

We also consider the problems for FSMs with *multiple input symbols*. If a FSM satisfies the above mentioned condition for some input symbol, the FSM has an extended synchronizing sequence where no other input symbol appears. Thus, from the necessary and sufficient condition for the case of a single input symbol, we obtain a sufficient condition for FSMs with multiple input symbols to be synchronizable. Since the sufficient condition is easily satisfied even for the case of only one resettable state variable, it seems that practical FSMs satisfy the condition. Actually, all MCNC'91 benchmark FSMs[10] satisfy the condition and are found to be synchronizable with one resettable state variable. Thus, it seems reasonable to conclude that *practical FSMs are synchronizable with only one resettable state variable.* This is the main contribution of this paper.

Furthermore, we consider the length of extended synchronizing sequence and show experimental results for MCNC'91 benchmark FSMs. The experimental results give considerable evidence to show that we can obtain a *short* extended synchronizing sequence even if we consider only a single input symbol of a FSM.

The paper is organized as follows. In Section 2, we formulate the extended synchronizing sequence and define two minimization problems concerning the number of scannable/resettable state variables. We completely solve the problems for FSMs with a single input symbol in Section 3. In Section 4, we consider the problems for FSMs with multiple input symbols. We also show the experimental results for MCNC'91 benchmark FSMs. Section 5 provides conclusions.

2 Preliminaries

2.1 Finite state machines

A *finite state machine* (FSM) M is defined by a triplet (S, I, δ) where S is a finite set of *states*, I is a finite set of *input symbols* and δ is a *state transition function*. The state transition function δ is a total function of $S \times I$ into S, and $\delta(s, i)$ is the state that M passes to when M is at state s and input symbol i is applied to M. The state transition function is naturally extended to a function $\hat{\delta}$ of $2^S \times I^+$ into 2^S as follows[1]:

1. For any state $s \in S$ and any input symbol $i \in I$, $\hat{\delta}(\{s\}, i) = \{\delta(s, i)\}$.
2. For any state $s \in S$ and any input symbol sequence $\tilde{i} = i_1 i_2 \dots i_m \in I^+$ $(m \geq 2)$, $\hat{\delta}(\{s\}, \tilde{i}) = \hat{\delta}(\hat{\delta}(\{s\}, i_1), \tilde{i}')$ where $\tilde{i}' = i_2 i_3 \dots i_m$.

[1] I^+ is a set of all finite and non-empty sequences of input symbols.

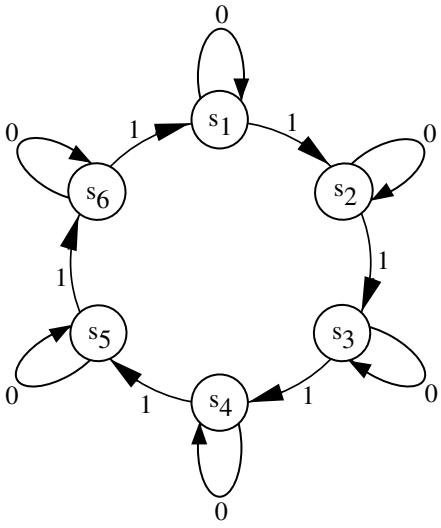

Figure 1: The state diagram of FSM M_1

3. For any state subset $S' \subseteq S$ and any input symbol sequence $\tilde{i} \in I^+$, $\hat{\delta}(S', \tilde{i}) = \bigcup_{s \in S'} \hat{\delta}(\{s\}, \tilde{i})$.

In the rest of this paper, the extended state transition function $\hat{\delta}$ is simply denoted by δ if no confusion occurs. Notice that we omit initial states, output symbols and an output function of a FSM for simplicity, since we consider only a (extended) synchronizing sequence that takes a FSM to a specified final state, *regardless* of the outputs or the initial state.

When a FSM is implemented by a sequential circuit, its current state is stored in flip-flops. In several approaches to design for testability, some of the flip-flops are designed to be *scannable* (i.e., their contents can be set to arbitrary values via an extra primary port) or *resettable* (i.e., their contents can be set to predetermined values via an extra primary port). A flip-flop of the circuit is modeled by a binary *state variable* of the FSM. A state variable is called *scannable* (resp. *resettable*) if it can be set to any value (resp. a predetermined value) by an extra input. We say that a *scan/reset operation* is executed, when the scannable/resettable state variables are set to some values by the extra inputs.

A *synchronizing sequence* of a FSM $M = (S, I, \delta)$ is defined to be an input symbol sequence $\tilde{i} \in I^+$ satisfying $\delta(S, \tilde{i}) = \{s\}$ for some state $s \in S$. We say that a FSM is *synchronizable* if it has a synchronizing sequence. Notice that a FSM is not necessarily synchronizable. For example, a FSM M_1 in Fig. 1 is not synchronizable.

However, it is obvious that any FSM can be taken to a specified final state regardless of the initial state, if all state variables are scannable or resettable. Thus, it is interesting to determine the minimum number of the scannable/resettable state variables necessary to make the FSM syn-

Table 1: State assignment f_1 of FSM M_1

state	y_0	y_1	y_2
s_1	0	0	0
s_2	0	0	1
s_3	0	1	1
s_4	0	1	0
s_5	1	1	0
s_6	1	1	1

chronizable. This is our main concern in this paper. To specify the problems we consider, we need some definitions.

2.2 State-assigned FSMs

Throughout this paper, let $M = (S, I, \delta)$ be a FSM and $n = |S|$. Let $y_0, y_1, \ldots y_{m-1}$ be m state variables ($m \geq \lceil \log_2 n \rceil$), and k ($0 \leq k \leq m$) be the number of scannable/resettable state variables. Without loss of generality, we assume that $y_0, y_1, \ldots y_{k-1}$ be scannable/resettable if $k \geq 1$, and that these state variables can be set to only zeros if they are resettable.

A *state assignment* f is an injective function[2] of S into an integer set $\{v \mid 0 \leq v < 2^m\}$: $f(s) = v$ implies that the state s is represented by setting $\langle y_0, y_1, \ldots y_{m-1} \rangle = v$ ($\langle y_0, y_1, \ldots y_{m-1} \rangle$ is the binary representation of the decimal number v). Furthermore, the decimal values of the scannable/resettable state variables and the other state variables are denoted by $f^s(s)$ and $f^n(s)$ respectively. That is, when $y_0, y_1, \ldots y_{k-1}$ are scannable/resettable, $f^s(s) = f(s)$ **div** 2^{m-k} and $f^n(s) = f(s)$ **mod** 2^{m-k}.

When a state assignment f and the number k of scannable/resettable state variables are fixed, the state transition caused by a scan/reset operation is uniquely determined. For example, consider a FSM M_1 (Fig. 1) with one scannable state variable y_0 and state assignment f_1 in Table 1. If M_1 is at state s_3, setting y_0 to 1 causes a state transition to s_6.

To consider formally the state transitions caused by a scan/reset operation, we shall define a *state-assigned FSM* for a given FSM. Intuitively, the state-assigned FSM is obtained by augmenting the given FSM with the input symbols and the state transitions caused by scan/reset operations. However, we need some attention in order that its state transition function may be a total function, since there may exist a state that has no next state for a scan/reset operation in case of $n < 2^m$. For example, consider the above FSM M_1 with the state assignment f_1, and consider the case that the scannable state variable y_0 is set to 1 when M_1 is at s_1. It becomes $(y_0, y_1, y_2) = (1, 0, 0)$, but there is no state s of M_1 such that $f(s) = 4$ ($= \langle 1, 0, 0 \rangle$). When FSM M_1 is implemented as

[2] A function f is injective if $f(s) = f(s')$ implies $s = s'$ for any states $s, s' \in S$.

a sequential circuit, $2^m - n$ *invalid states* are introduced. Among the invalid states, there exists s' that corresponds to $(y_0, y_1, y_2) = (1, 0, 0)$, and M_1 passes to s' in the above case. In synthesis of sequential circuits, state transitions from invalid states must be determined according to some objective (e.g., improvement in performance, area or testability). It may be possible to determine the state transitions for ease-to-synchronize. In this paper, however, the state transitions are not determined for ease-to-synchronize, so that they can be determined for other objective. In other words, we consider problems to make a FSM synchronizable whatever state transitions from invalid states are determined.

To consider the problems independently of the state transitions from invalid states, invalid states are not introduced in the state-assigned FSM. Instead, a fictitious state $s_\perp$ is introduced to make its state transition function total: if a state s has no next state for a scan/reset operation, add a state transition from s to $s_\perp$ on the scan/reset operation. Formally, the state-assigned FSM is defined as follows.

Definition 1 (a state-assigned FSM)
Let $M = (S, I, \delta)$ be an n-state FSM and $y_0, y_1, \ldots y_{m-1}$ be state variables ($m \geq \lceil \log_2 n \rceil$). Let f be a state assignment of M, and $y_0, y_1, \ldots y_{k-1}$ be scannable state variables. The *f-state-assigned FSM with k scannable state variables* $M^{f,k} = (S^{f,k}, I^{f,k}, \delta^{f,k})$ is defined as follows.

1. $S^{f,k} = S \cup \{s_\perp\}$ where $s_\perp \notin S$.
2. $I^{f,k} = I \cup \{\underline{i} \mid 0 \leq i < 2^k\}$ where $\underline{i} \notin I$ for each i $(0 \leq i < 2^k)$. The input symbol $\underline{i}$ $(0 \leq i < 2^k)$ corresponds to the scan operation that sets the scannable state variables so that $\langle y_0, y_1, \ldots y_{k-1} \rangle = i$. The set $\{\underline{i} \mid 0 \leq i < 2^k\}$ is denoted by I_s. We refer input symbols in I and I_s to *normal input symbols* and *scan input symbols* respectively.
3. The state transition function $\delta^{f,k}$ is defined as follows:
   (a) For $s \in S$ and $i \in I$, $\delta^{f,k}(s, i) = \delta(s, i)$.
   (b) For $s \in S$ and $\underline{i} \in I_s$ $(0 \leq i < 2^k)$,
   $$\begin{aligned} \delta^{f,k}(s, \underline{i}) &= s' \quad \text{(if there exists } s' \in S \text{ s.t. } f^s(s') = i \text{ and } f^n(s') = f^n(s)) \\ &= s_\perp \quad \text{(otherwise)} \end{aligned}$$
   (c) For $i \in I^{f,k}$, $\delta^{f,k}(s_\perp, i) = s_\perp$ ■

Figure 2 shows the f_1-state-assigned FSM of M_1 with one scannable state variable y_0.

For a FSM with resettable state variables, the *f-state-assigned FSM with k resettable state variables* and the *reset input symbol* are defined similarly. The only difference is the input symbol set: $I^{f,k} = I \cup \{\underline{0}\}$ instead of $I \cup \{\underline{i} \mid 0 \leq i < 2^k\}$.

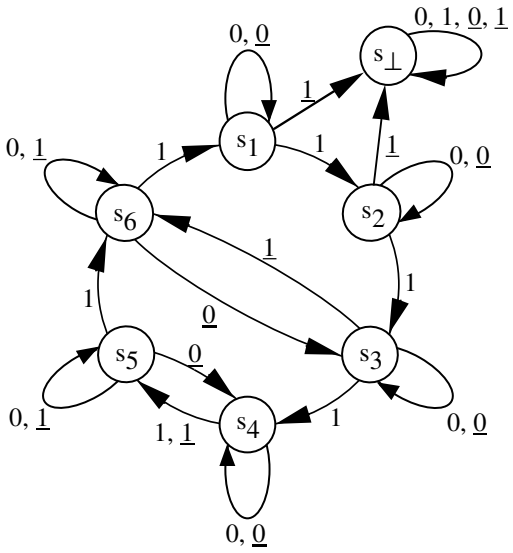

Figure 2: The f_1-state-assigned FSM with one scannable state variable $M_1^{f_1,1}$

In the following, we define an *extended synchronizing sequence* that consists of normal input symbols and scan/reset input symbols, and that takes a FSM to a specified final state regardless of the initial state.

Definition 2 (an extended synchronizing sequence)
Let $M = (S, I, \delta)$ be a FSM. Let f be a state assignment of M and $M^{f,k} = (S^{f,k}, I^{f,k}, \delta^{f,k})$ be the f-state-assigned FSM with k scannable/resettable state variables. An input symbol sequence $\tilde{i} \in I^{f,k^+}$ is called an *extended synchronizing sequence* (abbreviated to ESS) of $M^{f,k}$ if $\delta^{f,k}(S, \tilde{i}) = \{s\}$ for some $s \in S$.[3] The f-state-assigned FSM $M^{f,k}$ is called *extended-synchronizable* if it has an ESS. Furthermore, the FSM M is called *extended-synchronizable with k scannable/resettable state variables*, if there exists some state assignment f such that the f-state-assigned FSM with k scannable/resettable state variables $M^{f,k}$ is extended-synchronizable. ■

For example, the f_1-state-assigned FSM with one scannable state variable $M_1^{f_1,1}$ in Fig. 2 is extended-synchronizable, because it has an ESS $(\underline{0}, 1, 1, \underline{0}, 1, \underline{0})$. Thus, the FSM M_1 is extended-synchronizable with one scannable state variable.

Notice that the above definition of the extended-synchronizability does not consider invalid states that may be introduced when a FSM is implemented as a sequential circuit. Thus, the ESS, say $\tilde{i}$, may not drive the sequential circuit to a specified final state when applied to the sequential circuit at an invalid state. This problem can be simply solved by determining state transitions from invalid states so that $\delta(S_{inv}, \tilde{i}') \subseteq S$ for some input symbol sequence $\tilde{i}'$ where S is the state set of

[3] Note that $s_\perp \notin S$.

the given FSM and S_{inv} is the invalid state set: the input sequence corresponding to $\tilde{i'} \cdot \tilde{i}$ drives the sequential circuit to a specified final state regardless of the initial state (it may be an invalid state). The imposed condition for the state transitions from invalid states is easy, and it is left to designers to determine the state transitions for some objective under the condition.

2.3 Problems

We define two problems considered in this paper.

For any n-state FSM M, it is obvious that M is extended-synchronizable with $\lceil \log_2 n \rceil$ resettable state variables. Thus, it is interesting to find the minimum number of scannable/resettable state variables that make a given FSM extended-synchronizable. This is the first problem we consider.

Definition 3 (the scannable/resettable state variable minimization problem)
INSTANCE: An n-state FSM M and the number m of state variables ($m \geq \lceil \log_2 n \rceil$). (The state assignment of M is not yet determined.)
QUESTION: What is the minimum number k such that the FSM M is extended-synchronizable with k scannable/resettable state variables? ■

Even if the number k of scannable/resettable state variables is limited, it is possible to augment a given FSM with *additional states* (and state transitions from them) so that the resulting FSM is extended-synchronizable with k scannable/resettable state variables. From one of the results Jiang *et al.*[7] show, it follows that $n-2$ additional states are *enough* to make any given n-state FSM extended-synchronizable with one resettable state variable. Thus, it is interesting to find the minimum number of additional states that make a given FSM extended-synchronizable with k scannable/resettable state variables. The second problem is defined as follows.

Definition 4 (the additional state minimization problem)
INSTANCE: An n-state FSM M, the number m of state variables ($m \geq \lceil \log_2 n \rceil$) and the number k of scannable/resettable state variables. (The state assignment of M is not yet determined.)
QUESTION: What is the minimum number of additional states to make the resulting FSM extended-synchronizable with k scannable/resettable state variables. ■

3 FSMs with a single input symbol

We consider the problems defined in the previous section and show the *exact* solutions for a FSM $M = (S, I, \delta)$ with only a single input symbol, i.e., $|I| = 1$. More precisely, we show the necessary and sufficient condition for FSMs with a single input symbol to be extended-synchronizable with

k-resettable/scannable state variables, and to be augmented with l additional states so that the resulting FSM is extended-synchronizable with k-resettable/scannable state variables.

We consider only the extended-synchronizability, so we omit the term "extended" in this section.

3.1 The scannable/resettable state variable minimization problem

Lemma 1 Let $M = (S, I, \delta)$ be an n-state FSM such that $I = \{i\}$, $y_0, y_1, \ldots y_{m-1}$ $(m \geq \lceil \log n \rceil)$ be state variables, and $y_0, y_1, \ldots y_{k-1}$ be resettable state variables. If a state assignment f satisfies the following conditions for every state $s \in S$, the sequence $\underline{0}(i\,\underline{0})^{2^{m-k}-1}$ is an ESS of the f-state-assigned FSM with k-resettable state variables $M^{f,k}$.

1. If $f(s) = 0$, then $f^n(\delta(s, i)) = 0$.
2. If $f^s(s) = 0$ and $f^n(s) \neq 0$, then $f^n(\delta(s, i)) < f^n(s)$.
3. If $f^s(s) \neq 0$, then there exists a state $t \in S$ such that $f^s(t) = 0$ and $f^n(t) = f^n(s)$.

(Proof) We show by induction on h $(0 \leq h \leq 2^{m-k} - 1)$ that $\delta^{f,k}(S, \underline{0}(i\,\underline{0})^h) \subseteq \{u \in S \mid f(u) < 2^{m-k} - h\}$. It follows that $\underline{0}(i\,\underline{0})^{2^{m-k}-1}$ is an ESS of $M^{f,k}$.

1. The basis step $(h = 0)$: Since $y_0, y_1, \ldots, y_{k-1}$ is resettable state variables, it is obvious that $\delta^{f,k}(S, \underline{0}) \subseteq \{u \in S \mid f(u) < 2^{m-k}\}$.
2. The induction hypothesis: Assume that $\delta^{f,k}(S, \underline{0}(i\,\underline{0})^h) \subseteq \{u \in S \mid f(u) < 2^{m-k} - h\}$ holds for some h $(0 \leq h < 2^{m-k} - 1)$.
3. The induction step: Let $Q = \{u \in S \mid f(u) < 2^{m-k} - h\}$. Then,

$$\delta^{f,k}(S, \underline{0}(i\,\underline{0})^{h+1}) = \delta^{f,k}(\delta^{f,k}(S, \underline{0}(i\,\underline{0})^h), i\,\underline{0}) \subseteq \delta^{f,k}(Q, i\,\underline{0}).$$

Notice that $f^s(s) = 0$ for each state $s \in Q$. From the conditions 1 and 2, for each state $s \in Q$, if $f^n(s) = 0$ then $f^n(\delta(s, i)) = 0$ and if $f^n(s) \neq 0$ then $f^n(\delta(s, i)) < f^n(s)$. Thus, $\delta^{f,k}(Q, i) \subseteq \{u \in S \mid f^n(u) < 2^{m-k} - h - 1\}$. The condition 3 guarantees for every state $s \in S$ that there exists a state $t \in S$ such that $\delta^{f,k}(s, \underline{0}) = t$. Therefore,

$$\delta^{f,k}(S, \underline{0}(i\,\underline{0})^{h+1}) \subseteq \delta^{f,k}(Q, i\,\underline{0}) \subseteq \{u \in S \mid f(u) < 2^{m-k} - h - 1\}.$$

■

The *state diagram* of a FSM $M = (S, I, \delta)$ such that $I = \{i\}$ is a digraph $G = (S, A)$ where $A = \{(s, \delta(s, i)) \mid s \in S\}$. We use graph terminologies for the state diagram. A *semipath* is a state

sequence $u_1 u_2 \ldots u_h$ ($h \geq 1$) such that $\delta(u_j, i) = u_{j+1}$ or $\delta(u_{j+1}, i) = u_j$ for each j ($1 \leq j \leq h-1$). A *weakly-connected component* of G is a maximal subgraph of G such that there exists a *semipath* between every two states in the subgraph. Since each state has exactly one outgoing edge, every weakly-connected component C has a *sink state* that is reachable from every state in C[5].

The following theorem shows the necessary and sufficient condition for that a FSM is synchronizable with k scannable/resettable state variables.

Theorem 1 Let $M = (S, I, \delta)$ be an n-state FSM such that $I = \{i\}$, and $w(M)$ be the number of weakly-connected components of the state diagram of M.

1. Case that there exists a state s such that $\delta(s, i) = s$:
   (a) If $w(M) \leq (1 - \frac{1}{2^k})n + 1$, then M is synchronizable with k *resettable* state variables and has an ESS of length $2\lceil \frac{n}{2^k} \rceil - 1$.
   (b) If $w(M) > (1 - \frac{1}{2^k})n + 1$, then M is not synchronizable with k *scannable* state variables.
2. Otherwise (case that there exists no state s such that $\delta(s, i) = s$):
   (a) If $n = 2$ or $w(M) < n/2$, then M is synchronizable with *one resettable* state variable and has an ESS of length $2\lceil \frac{n}{2} \rceil - 1$.
   (b) Otherwise (case that $n \geq 4$ and $w(M) = n/2$):
      i. M is synchronizable with *two resettable* state variables and has an ESS of length $2\lceil \frac{n}{4} \rceil - 1$.
      ii. If $n/2$ is even, M is not synchronizable with *one scannable* state variable.
      iii. If $n/2$ is odd, M is synchronizable with *one scannable* state variable and has an ESS of length $2n - 6$, but is not synchronizable with *one resettable* state variable.

(Proof) Let $C_0, C_1, \ldots, C_{w(M)-1}$ be the weakly-connected components of the state diagram of M. Let t_j be a sink state in C_j and n_j be the number of states in C_j.

1. Case that there exists a state s such that $\delta(s, i) = s$: Notice that s is a sink.
   (a) Case of $w(M) \leq (1 - \frac{1}{2^k})n + 1$: Without loss of generality, we assume that $m = \lceil \log_2 n \rceil$, $\delta(t_0, i) = t_0$ and $n_1 \geq n_2 \geq \ldots \geq n_{w(M)-1}$.

   Figure 3 shows a procedure for determining a state assignment f_a so that the f_a-state-assigned FSM with k resettable state variables $M^{f_a,k}$ is synchronizable. It determines the state assignment f_a in the lexicographic order of $(f_a^n(s), f_a^s(s))$. Figure 4 shows

```
begin
    /* Initially every state is unmarked. */
    for fn := 0 to 2^{m-k} - 1 do begin
        if fn = 0 then begin
            determine f_a(t_0) = 0; mark t_0 end
        else if there exists an unmarked state then begin
            choose any unmarked state s s.t. δ(s,i) is marked; (*)
            determine f_a^s(s) = 0 and f_a^n(s) = fn; mark s end;
        for fs := 1 to 2^k - 1 do
            if t_j is unmarked for some J then begin
                choose the unmarked sink state t_j s.t. j = min{l | t_l is unmarked};
                determine f_a^s(t_j) = fs and f_a^n(t_j) = fn; mark t_j end
            else if there exists an unmarked state then begin
                choose any unmarked state s;
                determine f_a^s(s) = fs and f_a^n(s) = fn; mark s end
    end
end.
```

Figure 3: A procedure to determine the state assignment f_a

an example of the state assignment f_a where $n = 12$, $m = 4$ and $k = 1$. The state assignment is described by a pair $(f_a^s(s), f_a^n(s))$ for every state s in Fig. 4.

It is guaranteed from $w(M) \leq (1 - \frac{1}{2^k})n + 1$ and $n_1 \geq n_2 \geq \ldots \geq n_{w(M)-1}$ that there is an unmarked state s such that $\delta(s, i)$ is marked when the statement with (*) in Fig. 3 is executed. Thus, the procedure in Fig. 3 terminates and determines the state assignment $f_a(s)$ for every state $s \in S$. It is easy to show that the state assignment f_a satisfies the conditions in Lemma 1. Thus, M is synchronizable with k resettable state variables.

Although Lemma 1 gives us an ESS of length $2^{m-k+1} - 1$ (it becomes $\lceil \frac{n}{2^{k-2}} \rceil - 2$ in the worst case), we can obtain a better bound of the length by closely looking at the proof of Lemma 1. Since the procedure in Fig. 3 determines the state assignment in the lexicographic order of $(f_a^n(s), f_a^s(s))$, $f_a^n(s) \leq \lceil \frac{n}{2^k} \rceil - 1$ holds for every state $s \in S$. By a similar discussion to the proof of Lemma 1, we can show that the sequence $\underline{0}(i\,\underline{0})^{\lceil \frac{n}{2^k} \rceil - 1}$ is an ESS of $M^{f_a,k}$.

(b) Case of $w(M) > (1 - \frac{1}{2^k})n + 1$: For contradiction, we assume a f-state-assigned FSM with k scannable state variables $M^{f,k}$ is synchronizable. Construct an undirected graph $G = (S, E)$ from $M^{f,k}$ as follows:

$E = E_1 \cup E_2$ where

$E_1 = \{(s, \delta(s,i)) \mid s \in S\}$, and

$E_2 = \{(s,t) \mid f^s(s) \neq 0, f^s(t) = 0, f^n(s) = f^n(t), s \in S, t \in S\}$.

The graph $G - E_2$ [4] is the underlying graph of the state diagram of M. On the other

[4] $G - E_2$ is the graph obtained from G by deleting all edges in E_2.

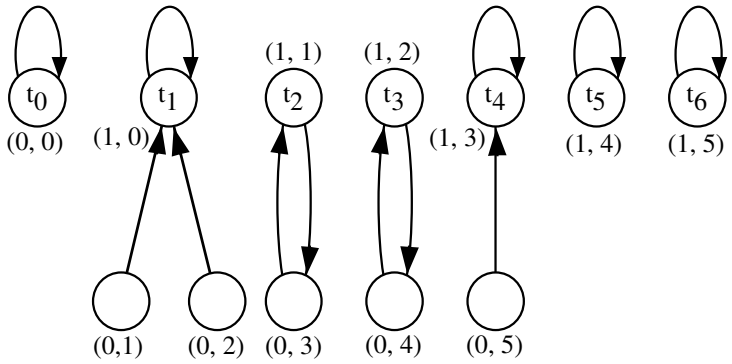

Figure 4: An example of the state assignment f_a

begin
  /* Initially every state is unmarked. */
  determine $f_b(t_0) = 0$; mark t_0;
  determine $f_b^s(\delta(t_0, i)) = 1$ and $f_b^n(\delta(t_0, i)) = 0$; mark $\delta(t_0, i)$;
  for $fn := 1$ **to** $2^{m-1} - 1$ **do begin**
    if there exists an unmarked state **then begin**
      choose any unmarked state s s.t. $\delta(s, i)$ is marked; (*)
      determine $f_b^s(s) = 0$ and $f_b^n(s) = fn$; mark s **end**;
    if t_j is unmarked for some j **then begin**
      choose the unmarked sink state t_j s.t. $j = \min\{l \mid t_l \text{ is unmarked}\}$;
      determine $f_b^s(t_j) = 1$ and $f_b^n(t_j) = fn$; mark t_j **end**
    else if there exists an unmarked state **then begin**
      choose any unmarked state s;
      determine $f_b^s(s) = 1$ and $f_b^n(s) = fn$; mark s **end**
  end
end.

Figure 5: A procedure to determine the state assignment f_b

hand, each connected component of $G - E_1$ consists of the states from which $M^{f,k}$ reaches the same state by a scan operation. Obviously, G is a connected graph if $M^{f,k}$ is synchronizable. However, since the number of connected components of $G - E_2$ is $w(M)$ $(> (1 - \frac{1}{2^k})n + 1)$ and $|E_2| \leq (1 - \frac{1}{2^k})n$, G is not connected. This is a contradiction.

2. Case that there exists no state s such that $\delta(s, i) = s$: Without loss of generality, we assume $n_0 \geq n_1 \geq \ldots \geq n_{w(M)-1}$. Since there exists no state s such that $\delta(s, i) = s$, each weakly-connected component consists of two or more states, i.e., $n_{w(M)-1} \geq 2$.

   (a) Case that $n = 2$ or $w(M) < n/2$: It is obvious that M is synchronizable with one resettable variable if $n = 2$. In the following, we consider the case of $w(M) < n/2$. Without loss of generality, we assume that $m = \lceil \log_2 n \rceil$.

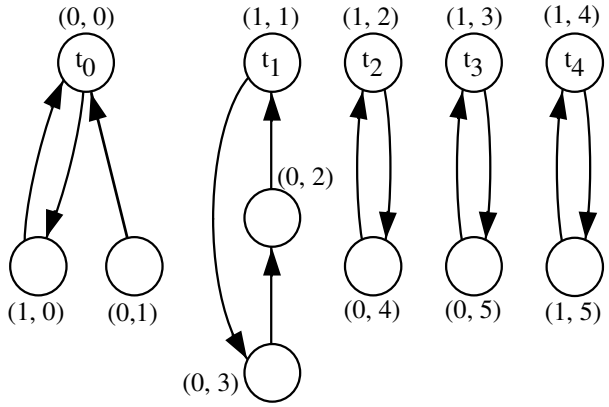

Figure 6: An example of the state assignment f_b

Figure 5 shows a procedure for determining a state assignment f_b so that the f_b-state-assigned FSM with one resettable state variable $M^{f_b,1}$ is synchronizable. It determines the state assignment f_b in the lexicographic order of $(f_b^n(s), f_b^s(s))$. Figure 6 shows an example of the state assignment f_b where $n = 12$, $m = 4$ and $k = 1$. The state assignment is described by a pair $(f_b^s(s), f_b^n(s))$ for every state s in Fig. 6.

It follows from $w(M) < n/2$ and $n_0 \geq n_1 \geq \ldots \geq n_{w(M)-1}$ that $n_0 \geq 3$. From $n_0 \geq 3$ and $n_j \geq 2$ $(1 \leq j \leq w(M) - 1)$, it is guaranteed that there is an unmarked state s such that $\delta(s, i)$ is marked when the statement with (*) in Fig. 5 is executed. Thus, the procedure in Fig. 5 terminates and determines the state assignment $f_b(s)$ for every state $s \in S$. It is easy to show that the state assignment f_b satisfies the conditions in Lemma 1. Thus, M is synchronizable with one resettable state variables. By a similar discussion to that of Case 1(a), we can prove that M has an ESS of length $2\lceil \frac{n}{2} \rceil - 1$.

(b) Case that $n \geq 4$ and $w(M) = n/2$: Clearly, $n_j = 2$ holds for each j $(0 \leq j \leq n/2-1)$. Let u_j^0 and u_j^1 be the two states in weakly-connected component C_j. Notice that $\delta(u_j^0, i) = u_j^1$ and $\delta(u_j^1, i) = u_j^0$ hold.

i. Without loss of generality, we assume $m = \lceil \log_2 n \rceil$. Determine the state assignment f_c as follows:

- For each u_j^0 $(0 \leq j \leq \lceil n/4 \rceil - 1)$, $f_c^s(u_j^0) = 0$ and $f_c^n(u_j^0) = j$.
- For each u_j^1 $(0 \leq j \leq \lceil n/4 \rceil - 1)$, $f_c^s(u_j^1) = (j \bmod 3) + 1$ and $f_c^n(u_j^1) = \lfloor j/3 \rfloor$.
- For the other states s, determine $f_c(s)$ so that f_c should be injective and $f_c^n(s) \leq \lceil n/4 \rceil - 1$.

Figure 7 shows an example of the state assignment f_c. It is easy to show that the state assignment f_c satisfies the conditions in Lemma 1. Thus, M is synchronizable

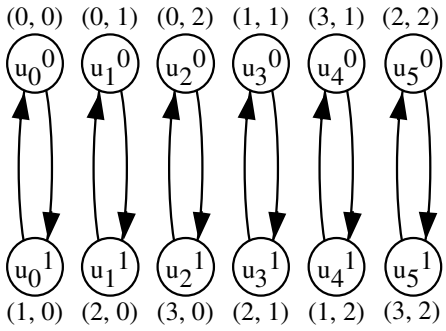

Figure 7: An example of the state assignment f_c

with two resettable state variables, and has an ESS of length $2\lceil \frac{n}{4} \rceil - 1$.

ii. For contradiction, we assume an f-state-assigned FSM with one scannable state variable $M^{f,1}$ is synchronizable. Consider an undirected graph $G = (S, E)$ similar to the Case 1(b). Since $M^{f,1}$ is synchronizable, G is connected. Thus, without loss of generality, we can assume that $f^n(u_j^0) = f^n(u_{j+1}^1)$ for each j $(0 \leq j < n/2-1)$ and $f^n(u_{n/2-1}^0) = f^n(u_0^1)$. For any input symbol sequence $\tilde{i}$ of $M^{f,1}$, let $\delta^{f,1}(u_0^0, \tilde{i}) = u_a^b$ $(0 \leq a \leq n/2-1,\ 0 \leq b \leq 1)$ and $\delta^{f,1}(u_0^1, \tilde{i}) = u_c^d$ $(0 \leq c \leq n/2-1,\ 0 \leq d \leq 1)$. By mathematical induction on the length of the input symbol sequence, we can prove that $b = d$ holds if and only if $|a - c|$ is odd. It follows that $\delta^{f,1}(u_0^0, \tilde{i}) \neq \delta^{f,1}(u_0^1, \tilde{i})$ for any input symbol sequence $\tilde{i}$. This is a contradiction.

iii. First, we prove that M is synchronizable with one scannable state variable. Without loss of generality, we assume $m = \lceil \log_2 n \rceil$. Consider the f_d-state-assigned FSM with one scannable state variable $M^{f_d,1}$ where the state assignment f_d is as follows:

$$f_d^s(u_j^0) = 0 \text{ and } f_d^n(u_j^0) = j \text{ for each } j\ (0 \leq j < n/2-1)$$
$$f_d^s(u_{n/2-1}^0) = 1 \text{ and } f_d^n(u_{n/2-1}^0) = n/2-1$$
$$f_d^s(u_j^1) = 1 \text{ and } f_d^n(u_j^1) = j-1 \text{ for each } j\ (0 < j \leq n/2-1)$$
$$f_d^s(u_0^1) = 0 \text{ and } f_d^n(u_0^1) = n/2-1$$

It is easy to show that $\delta^{f_d,1}(S, \underline{0}(i\,\underline{0})^{n/2-2}\underline{1}(i\,\underline{1})^{n/2-2}) = u_{n/2-1}^1$. Therefore, M is synchronizable with one scannable state variable, and has an ESS of length $2n-6$. Now, we prove that M is not synchronizable with one resettable state variable. For contradiction, we assume that an f-state-assigned FSM with one resettable state variable $M^{f,1}$ is synchronizable. If there exists j $(0 \leq j \leq n/2-1)$ such that $f^s(u_j^0) = f^s(u_j^1) = 0$, then it is clear that there is no input symbol sequence $\tilde{i}$ satisfying $\delta^{f,1}(u_j^0, \tilde{i}) = \delta^{f,1}(u_j^1, \tilde{i})$. Thus, without loss of generality, we assume that

$$f^s(u_j^0) = 0 \text{ and } f^n(u_j^0) = j \text{ for each } j\ (0 \leq j \leq n/2-1)$$

$f^s(u_j^1) = 1$ and $f^n(u_j^1) = j - 1$ for each j $(0 < j \leq n/2 - 1)$

$f^s(u_0^1) = 1$ and $f^n(u_0^1) = n/2 - 1$

It is easy to show that there is no input symbol sequence $\tilde{i}$ satisfying $\delta^{f,1}(u_j^0, \tilde{i}) = \delta^{f,1}(u_j^1, \tilde{i})$. Therefore, M is not synchronizable with one resettable state variable. ■

Let $\mathcal{M}_n$ be the set of all n-state FSMs with a single input symbol, and $\mathcal{S}_n^k \subseteq \mathcal{M}_n$ (resp. $\mathcal{R}_n^k \subseteq \mathcal{M}_n$) be the set of all n-state FSMs that are synchronizable with k scannable (resp. resettable) state variables. The following relations obviously hold.

$$\mathcal{R}_n^0 = \mathcal{S}_n^0$$

$$\mathcal{R}_n^k \subseteq \mathcal{S}_n^k \ (1 \leq k < \lceil \log_2 n \rceil)$$

$$\mathcal{R}_n^{\lceil \log_2 n \rceil} = \mathcal{S}_n^{\lceil \log_2 n \rceil} = \mathcal{M}_n$$

$$\mathcal{S}_n^0 \subseteq \mathcal{S}_n^1 \subseteq \ldots \subseteq \mathcal{S}_n^{\lceil \log_2 n \rceil} = \mathcal{M}_n$$

$$\mathcal{R}_n^0 \subseteq \mathcal{R}_n^1 \subseteq \ldots \subseteq \mathcal{R}_n^{\lceil \log_2 n \rceil} = \mathcal{M}_n$$

From Theorem 1, the following hierarchy is obtained.

Corollary 1 The following relations hold.

$$\mathcal{R}_n^0 = \mathcal{S}_n^0 \subset \mathcal{R}_n^1 \subset \mathcal{S}_n^1$$

$$\mathcal{S}_n^{k-1} \subset \mathcal{R}_n^k = \mathcal{S}_n^k \ (2 \leq k \leq \lceil \log_2 n \rceil)$$

$$\mathcal{R}_n^{\lceil \log_2 n \rceil} = \mathcal{S}_n^{\lceil \log_2 n \rceil} = \mathcal{M}_n$$

■

3.2 The additional state minimization problem

Jiang et al. [7] showed any FSM $M = (S, I, \delta)$ can be augmented with $n-2h$ additional states so that the resulting FSM is synchronizable with one resettable state variable, where $h = \max_{s \in S, i \in I} |\{t \in S \mid \delta(t, i) = s\}|$. It is noteworthy that the length of the ESS is *two* (i.e. $\underline{0}\, i$ for some $i \in I$) and is obviously worst-case optimal.

In the additional state minimization problem, however, our main concern is the minimum number of the additional states rather than the minimum length of the ESS. Since Theorem 1 shows the necessary and sufficient condition for synchronizable FSMs, we can find the minimum number of the additional states when given a FSM with a single input symbol and the number of scannable/resettable state variables: we only consider how the given FSM can be augmented with additional states so that it should satisfy the condition in Theorem 1.

Theorem 2 Let $M = (S, I, \delta)$ be an n-state FSM such that $I = \{i\}$ and $w(M)$ be the number of weakly-connected components of the state diagram of M.

1. Case that there exists a state s such that $\delta(s, i) = s$ and that $w(M) > (1 - \frac{1}{2^k})n + 1$ (Notice that M is not synchronizable with k *scannable* state variables from Theorem 1):

   - M cannot be augmented with $\lceil \frac{2^k}{2^k-1}(w(M) - 1) \rceil - n - 1$ additional states so that the resulting FSM is synchronizable with k *scannable* state variables .
   - M can be augmented with $\lceil \frac{2^k}{2^k-1}(w(M) - 1) \rceil - n$ additional states so that the resulting FSM is synchronizable with k *resettable* state variables.

2. Case that there exists no state s such that $\delta(s, i) = s$ and that $n \geq 4$ and $w(M) = n/2$ (Notice that M is not synchronizable with one resettable state variable from Theorem 1): M can be augmented with one additional state so that the resulting FSM is synchronizable with one *resettable* state variable.

(Proof)

1. We cannot decrease the number of weakly-connected components by adding states and related state transitions. However, we can increase the number of states with keeping the number of weakly-connected components unchanged. For example, this can be achieved by adding state s and determining $\delta(s, i) = t$ for some $t \in S$. Thus, from Theorem 1 the minimum number of the additional states is the minimum l satisfying the following inequality.

$$w(M) \leq (1 - \frac{1}{2^k})(n + l) + 1$$

2. If we add a state s and determine $\delta(s, i) = t$ for some $t \in S$, the resulting FSM is synchronizable with one resettable state variable from Theorem 1. ■

It is sometimes required that each of the additional states should be equivalent[8] to some state of the given FSM. If we choose, in the above proof, the state $t \in S$ satisfying $\delta(u, i) = t$ for some $u \in S$, then we can make the additional state s equivalent to u by determining adequately the output function.

4 FSMs with multiple input symbols

In the previous section, we have treated only FSMs with a single input symbol. In this section, we apply the results in the previous section to FSMs with multiple input symbols.

For a FSM $M = (S, I, \delta)$ and an input symbol $i \in I$, consider the *i-restricted FSM* $M' = (S, I' = \{i\}, \delta')$ that is obtained from M by restricting the input symbol set to $\{i\}$: $\delta'(s, i) = \delta(s, i)$ for every state $s \in S$. If M' is extended-synchronizable with k scannable/resettable state variables, then M is clearly extended-synchronizable with k scannable/resettable state variables. Thus Theorems 1 and 2 give us sufficient conditions for FSMs with multiple input symbols to be extended-synchronizable. Especially, we can obtain the following result from Theorem 1.

Corollary 2 Let $M = (S, I, \delta)$ be any n-state FSM. The FSM M is extended-synchronizable with *one resettable* state variable, if there exists an input symbol $i \in I$ satisfying the following condition (where M_i denotes the i-restricted FSM of M).

- $w(M_i) \leq n/2 + 1$, and
- if there is no state $s \in S$ such that $\delta(s, i) = s$, then $n < 4$ or $w(M_i) < n/2$ ■

Corollary 2 gives us a sufficient condition for FSMs to be extended-synchronizable with one resettable state variable. Since the condition is easily satisfied, practical FSMs seem to be extended-synchronizable with one resettable state variable. Actually, we confirm that all MCNC'91 benchmark FSMs[10] satisfy the condition and are extended-synchronizable with one resettable state variable. We may, therefore, reasonably conclude that *practical FSMs are extended-synchronizable with one resettable state variable.*

More precisely, the following holds for a FSM M in most cases: M has an ESS where only a single normal input symbol i and the reset input symbol $\underline{0}$ appear (repeatedly). From a practical point of view, this is also desirable: because the values fed to the normal input terminals of the sequential circuit need not change during the synchronizing process, the ESS can be completely specified by only *one* input pattern and a bit sequence that specifies positions of the normal input symbol in the ESS. Thus,the storage to record the ESS is small. On the other hand, for the case of general ESSs, we need much more storage since the ESS is specified by a sequence of input patterns.

The shortest ESS of an i-restricted FSM may be longer than that of its original FSM. While our main concern is the minimum number of the scannable/resettable state variables rather than the minimum length of the ESS, we made experiments using the MCNC'91 benchmark FSMs to investigate the difference in length of the shortest ESSs. Some FSMs in the benchmark set have reset inputs explicitly, i.e., they are synchronizable by an input symbol sequence of length one. Hence, we made the experiments on the FSMs that have no reset inputs. As stated above, all the benchmark FSMs are extended-synchronizable with one resettable state variable. Thus, we made the experiments for the case that the numbers of state variables and scannable/resettable state variables are $\lceil \log_2 n \rceil$ (n is the number of states) and one, respectively.

Table 2 shows the experimental results. The first three columns, FSM, ST and PI, denote the FSM name, the number of states and the bit size of a normal input symbol (i.e, the number of normal input symbols is 2^{PI}), respectively. The fourth column, Single, denotes the length of the shortest ESS under the constraint that only a single normal input symbol i and the reset input symbol $\underline{0}$ appear in the ESS. To find the shortest ESS, we applied a heuristic algorithm[6] to the i-restricted FSM for every normal input symbol i. The fifth column, Min, denotes the length of the shortest ESS without the constraint, i.e., all normal input symbols and all scan input symbols ($\underline{0}$ and $\underline{1}$) can appear in the ESS. We estimated the length of the shortest ESS by the exhaustive search. But for some FSMs (marked with † in Table 2), the exhaustive search failed to find the shortest ESS because of lack of memory or time. For those FSMs, we estimated the length of the shortest ESS by the heuristic algorithm[6].

We can see that the ESS in the column Single is as short as that in the column Min except for two FSMs, modulo12 and sand. Thus, we may conclude that the ESS consisting of one kind of normal input symbols and the reset input symbol $\underline{0}$ is sufficiently effective even for FSMs with multiple input symbols.

5 Conclusions

In this paper, we formulated an extended synchronizing sequence that consists of normal input symbols and scan/reset input symbols, and that takes a FSM to a specified final state regardless of the initial state. We newly introduced two problems concerning the minimum number of the scannable/resettable state variables that make a FSM extended-synchronizable. We *completely* solved the problems for FSMs with a single input symbol. We also considered the problems for FSMs with multiple input symbols and concluded that practical FSMs are extended-synchronizable with only one resettable state variable. Furthermore, we considered the length of the extended synchronizing sequence. From the experimental results for MCNC'91 benchmark FSMs, we also concluded that our approach is effective to find a short extended synchronizing sequence of a FSM with multiple input symbols.

Acknowledgment

It is a pleasure to thank Professor Kewal K. Saluja (Univ. of Wisconsin-Madison) for his helpful comments and discussions on this work.

References

[1] V. D. Agrawal and K. T. Cheng. A complete solution to the partial scan problem. In *Proc. Int. Test Conf.*, pages 44–51, 1987.

[2] K. T. Cheng and V. D. Agrawal. Initializability consideration in sequential machine synthesis. *IEEE Trans. Comput.*, 41(3):374–379, 1992.

[3] V. Chickermane and J. H. Patel. An optimization based approach to the partial scan design problem. In *Proc. Int. Test Conf.*, pages 377–386, 1990.

[4] H. Fujiwara. *Logic Testing and Design for Testability.* The MIT Press, 1985.

[5] F. Harary. *Graph Theory.* Addison-Wesley, 1969.

[6] T. Inoue, T. Masuzawa, H. Youra, and H. Fujiwara. An approach to the synthesis of synchronizable finite state machines with partial scan. In *Proc. ATS '96 (to appear)*, 1996.

[7] N. Jiang, R. M. Chou, and K. K. Saluja. Synthesizing finite state machines for minimum length synchronizing sequence using partial scan. In *Digest of 25th Int. Sympo. Fault-Tolerant Computing Systems*, pages 41–49, 1995.

[8] Z. Kohavi. *Switching and Finite Automata Theory.* McGraw-Hill, 1978.

[9] I. Pomeranz and S. M. Reddy. On the role of hardware reset in synchronous sequential circuit test generation. *IEEE Trans. Comput.*, 43(3):1100–1105, 1994.

[10] S. Yang. Logic synthesis and optimization benchmarks user guide version 3.0. Technical report, Microelectronics Center of North Carolina, Research Triangle Park, NC, 1991.

Table 2: Experimental results: ESS length

FSM	ST	PI	ESS Length	
			Single	Min
bbara	10	4	2	2
bbsse	16	7	2	2
bbtas	6	2	2	2
beecount	7	3	2	2
dk14	7	3	2	2
dk16	27	2	3	3 †
dk17	8	2	2	2
dk27	7	1	2	2
dk512	15	1	3	3 †
donfile	24	2	3	3 †
ex1	20	9	2	2
ex2	19	2	2	2
ex3	10	2	2	2
ex4	14	6	4	4 †
ex5	9	2	2	2
ex7	10	2	2	2
keyb	19	7	2	2
lion	4	2	2	2
lion9	9	2	2	2
mc	4	3	2	2
modulo12	12	1	7	5 †
planet	48	7	4	4 †
planet1	48	7	4	4 †
s1	20	8	2	2
s1a	20	8	2	2
s8	5	4	2	2
sand	32	11	6	4 †
shiftreg	8	1	3	3
sse	16	7	2	2
styr	30	9	2	2
tav	4	4	3	3
train11	11	2	2	2
train4	4	2	2	2

†: the length is estimated using a heuristic algorithm in [6].