Universal Single Transition Time Asynchronous State Assignments

ARTHUR D. FRIEDMAN, MEMBER, IEEE, R. L. GRAHAM, AND JEFFREY D. ULLMAN, MEMBER, IEEE

***Abstract*—In this paper we consider the problem of deriving upper bounds on the number of state variables required for an n-state universal asynchronous state assignment (i.e., a state assignment which is valid for any n-state asynchronous sequential function). We will consider a special class of state assignments called SST assignments which were first derived by Liu [1] and later extended by Tracey [2]. In these assignments all variables which must change in a given transition are allowed to change simultaneously without critical races. The best universal bound known so far has been developed by Liu and requires $2^{S_0}-1$ state variables, where $S_0=[\log_2 n]$, n being the number of states, and $[x]$ being the least integer $\geq x$. We shall show how this bound can be substantially improved. We further show that, by generalizing the state assignment to allow multiple codings for states, the bounds can be still further improved.**

A mathematical statement of the problem is as follows. Define an (i, j)-separating system (SS) for a finite set S to be a family of subsets $S_1, S_2, \cdots, S_t$ of S such that for any subsets $P, Q \subseteq S$ with $|P|=i$, $|Q|=j$ and $P \cap Q=\phi$, there is a set S_k of the family for which either $P \subseteq S_k$, $Q \cap S_k=\phi$ or $Q \subseteq S_k$, $P \cap S_k=\phi$. Let $m=m(i, j, n)$ denote the *minimum* value that t can assume for an (i, j)-*SS* when $|S|=n$. We show that

$$m \leq \frac{(i+j)\log n}{-\log(1-2^{1-i-j})}.$$

***Index Terms*—Asynchronous state assignments, sequential circuits, single transition time state assignments.**

IN THIS paper we consider the problem of deriving upper bounds on the number of state variables required for an n-state universal asynchronous state assignment (i.e., a state assignment which is valid for any n-state asynchronous sequential function). An asynchronous sequential circuit differs from a synchronous sequential circuit in that it contains no source of clock pulses which regulate the circuit. For the synchronous case, circuit terminal action is examined only when a clock signal appears. Hence, transient conditions during the change of state variables can be completely ignored and several state variables are allowed to change simultaneously. For the asynchronous case, circuit action is examined at all times. Therefore, transient conditions cannot be ignored and several state variables are allowed to change simultaneously, an event referred to as a race, only if the resulting state does not depend on the order of change of these variables, referred to as a noncritical race. The problem of races can be handled by restricting the state assignment in such a manner that there are no state transitions which involve critical races. We will consider a special class of state assignments called STT assignments which were first derived by Liu [1] and later extended by Tracey [2]. In these assignments all variables which must change in a given transition are allowed to change simultaneously without critical races. The best universal bound known so far has been developed by Liu and requires $2^{S_0}-1$ state variables where $S_0=[\log_2 n]$, n being the number of states, and $[x]$ being the least integer $\geq x$. We shall show that this bound can be substantially improved. We further show that by generalizing the state assignment to allow multiple codings for states, the bounds can be still further improved.

It follows from Theorem 1 of [2] that a universal STT assignment with one coding per state must have some variable y, which partitions every pair of states ij from every other pair kl such that $l, k \neq i, j$. We shall say y must satisfy the dichotomy $(i, j: k, l)$. (It is only necessary to consider dichotomies involving four states, since every dichotomy involving only three states is satisfied by the variable satisfying some 4-state dichotomy.) For $n=5$ there are few enough dichotomies to consider the problem in a straightforward manner.

Below we list all the 4-state dichotomies on five states, find the set of maximal compatibles, and then find a minimal covering set as described in [2].

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	0	0	0	0			
2	0	0	0	1	1		1	1		1	1		0	0	0
3	1	1		0	0	0	1		1	1		1	0	1	1
4	1		1	1		1	0	0	0		1	1	1	0	1
5		1	1		1	1		1	1	0	0	0	1	1	0

Max Compatibles 1 2 3, 1 12 15, 2 9 14, 3 6 13, 4 5 6, 4 11 $\overline{14}$, 5 8 $\overline{15}$, 7 8 9, 7 10 $\overline{13}$, 10 11 12.

Minimal covering set 1 12 15, 2 9 14, 3 6 13, 4 5 6, 7 8 9, 10 11 12.

The minimal STT assignment is as follows.

	y_1	y_2	y_3	y_4	y_5	y_6
1	0	0	0	0	0	0
2	0	0	0	1	1	1
3	1	1	0	0	1	1
4	1	0	1	1	0	1
5	0	1	1	1	1	0

For $n=6$ the problem becomes too large to be handled in this straightforward manner. However, the following theorem enables us to find a universal 6-state assignment with the minimum number of state variables.

Theorem 1: There is no universal 6-state STT assignment with one coding per state with 6-state variables.

Proof: For a universal 6-state assignment there are

Manuscript received May 1, 1968; revised January 16, 1969.
The authors are with Bell Telephone Laboratories, Inc., Murray Hill, N. J. 07974.

$$45 = \frac{1}{2}\cdot\binom{6}{2}\cdot\binom{4}{2}$$

2-pair dichotomies which must be covered. Suppose there is such an assignment with 6-state variables.

A variable with three 0's and three 1's covers nine 2-pair dichotomies. A variable with four 0's and two 1's covers six such dichotomies. Hence, there must be at least three variables with three 0's and three 1's in order to cover 45 dichotomies with six variables. Without loss of generality, assume that one of them has 0's in states 1, 2, and 3. Then without loss of generality, the second variable is as shown below.

	Case 1		Case 2	
1	0	0	0	0
2	0	0	0	1
3	0	1	0	1
4	1	0	1	0
5	1	1	1	0
6	1	1	1	1

In either case there is one dichotomy covered by both variables. In addition, the third variable will cover one dichotomy covered by each of the first two (but not by both of them). Hence, the three variables only cover at most 24 distinct dichotomies and any other variable with 3–3 distribution will cover at most seven new dichotomies. Thus, every variable must have a 3–3 distribution if we are only to use six variables.

Since every pair of these variables cover a common distinct dichotomy, six variables with a 3–3 distribution cover at most

$$9\cdot 6 - \binom{6}{2} = 39 < 45$$

dichotomies, and hence six variables are not sufficient.

We have thus shown that the Liu bound of $2^{S_0}-1$ is exact for $n=6$, 7, or 8, since a universal assignment for seven or eight states requires at least as many variables as a universal assignment for six states. However, the Liu bound is not exact for larger values of n, even if $n=2^{S_0}$. In order to derive a constructive class of codes which substantially improve upon this bound, we will need the following preliminary results.

Definition 1: An (i, j)-*separating system* (SS) on a set of states S is a family of subsets $S_1, \cdots, S_t$ of S such that for any subset P with i or fewer members of S and any subset Q, disjoint from P, with j or fewer members of S, there is a k such that either $P \subseteq S_k$, $Q \cap S_k = \phi$ or $Q \subseteq S_k$, $P \cap S_k = \phi$ (where ϕ denotes the empty set). Let $M_{i,j}(S_0)$ denote the minimum number of subsets required for such a separating system if S has 2^{S_0} elements. We will denote the elements of S by a binary string of length t. The kth component is 1 if and only if that element is in S_k.

It is apparent that a valid universal STT assignment is actually a (2, 2)-separating system. We wish to find bounds on $M_{2,2}(S_0)$. To this end we need the following results.

Lemma 1:

$$M_{1,1}(S_0) = S_0.$$

Proof: Assign a unique coding to every state of S. This requires S_0 binary variables if S has 2^{S_0} elements. This defines a (1, 1)-SS since for any two states i, j of S there is some variable y in which i has the value 0 and j the value 1, or vice versa (since i and j do not have the same coding). If fewer than S_0 variables are used, two states i, j must have the same coding and, therefore, are not separated.

Lemma 2:

$$M_{2,1}(S_0) \le \frac{S_0(S_0+1)}{2}.$$

Proof: For any three states i, j, k of S we wish to have some variable y in which i and j are 0 and k is 1, or vice versa. Let the elements of S be arbitrarily labeled $1, 2, \cdots, 2^{S_0}$ and let $n = 2^{S_0-1}$.

The coding R shown below is a (2, 1)-SS on 2^{S_0} states

$$R = \begin{array}{r} 1 \\ \\ 2^{S_0-1} \\ \\ 2^{S_0} \end{array} \left[\begin{array}{c|c|c} & & 0 \\ T & S & 0 \\ & & 0 \\ \hline & & 1 \\ T & \bar{S} & 1 \\ & & 1 \end{array}\right]$$

where T is a (2, 1)-SS on 2^{S_0-1} states and S is a (1, 1)-SS on 2^{S_0-1} states. To show this let i^* be i mod n, j^* be j mod n, and k^* be k mod n.

If i^*, j^*, and k^* are all distinct, there is some variable of T which covers $(i^*, j^*; k^*)$ and this same variable covers $(i, j; k)$.

The only other categories we must consider are shown below.

1) $(i, (i+n); j)$ $\quad ,i, j \le n$
2) $(i, (i+n); j+n)$
3) $(i; i+n, j)$
4) $(i, j; i+n)$
5) $(i; i+n, j+n)$
6) $(i, j+n; i+n)$.

There is some variable of T which separates i from j and this variable of R satisfies cases 1) and 2) since $i = i+n$, $j = j+n$ in these variables. Similarly, there is some variable of S which separates i from j and this same variable satisfies cases 3) and 6), since in this variable $i+n = \bar{i}$ and $j+n = \bar{j}$. Cases 4) and 5) are satisfied by the last variable of R.

We thus obtain the recursive relation

$$M_{2,1}(S_0) \leq M_{2,1}(S_0 - 1) + M_{1,1}(S_0 - 1) + 1$$
$$\leq M_{2,1}(S_0 - 1) + S_0.$$

Iterating S_0 times we obtain

$$M_{2,1}(S_0) \leq \sum_{i=1}^{S_0} i = \frac{S_0(S_0 + 1)}{2}.$$

Theorem 2:

$$M_{2,2}(S_0) \leq \frac{S_0^3 + 5S_0}{6}.$$

Proof: For any four states i, j, k, and l of S we wish to have some variable y in which i and j are 0 and k and l are 1, or vice versa. Let the members of S be arbitrarily labeled $1, 2, \cdots, 2^{S_0}$ and let $n = 2^{S_0-1}$.

The coding U shown below is a (2, 2)-SS on 2^{S_0} states,

$$U = \begin{array}{r} 1 \\ \\ 2^{S_0-1} \\ \\ 2^{S_0} \end{array} \left[\begin{array}{c|c|c} V & T & \begin{matrix}0\\0\\0\end{matrix} \\ \hline V & \overline{T} & \begin{matrix}1\\1\\1\end{matrix} \end{array}\right]$$

where V is a (2, 2)-SS on 2^{S_0-1} states and T is a (2, 1)-SS on 2^{S_0-1} states.

The proof is similar to that of Lemma 2 and the various cases are summarized in the table below.

Dichotomy Class	*Covering Column*
1) $(i, j; k, l)$ [all distinct modulo n]	column of V which covers $(i^*, j^*; k^*, l^*)$
2) $(i, (i+n); j, (j+n))$	column of V which covers $(i; j)$
3) $(i, j+n; i+n, j)$	column of T which covers $(i; j)$
4) $(i, j; i+n, j+n)$	last column of U
5) $(i, i+n; j, k)$	column of V which covers $(i; j, k)$
6) $(i, i+n; j+n, k)$	column of V which covers $(i; j, k)$
7) $(i, i+n; j+n, k+n)$	column of V which covers $(i; j, k)$
8) $(i, j; i+n, k)$	column of T which covers $(i, j; k)$
9) $(i, j; i+n, k+n)$	last column of U
10) $(i, j+n; i+n, k)$	column of T which covers $(i; k, j)$
11) $(i, j+n; i+n, k+n)$	column of T which covers $(i, k; j)$

We therefore obtain the relation

$$M_{2,2}(S_0) \leq M_{2,2}(S_0 - 1) + M_{2,1}(S_0 - 1) + 1$$
$$\leq M_{2,2}(S_0 - 1) + \frac{S_0(S_0 - 1)}{2} + 1.$$

Iterating S_0 times we obtain

$$M_{2,2}(S_0) \leq S_0 + \sum_{i=1}^{S_0} \frac{i(i-1)}{2} = \frac{S_0^3 + 5S_0}{6}.$$

A different class of constructive codes for 2–1 and 2–2 separating systems can be obtained by the constructive procedures described in Theorems 3 and 4.

Let M be an $m \times n$ matrix. Define θM to be an $m \times n$ matrix whose first row is the mth row of M, and whose ith row, $2 \leq i \leq m$, is the $(i-1)$st row of M. That is, the θ operator cycles the rows of M, bringing the bottom row to the top. Let $\theta^0 M = M$ and, for $i \geq 1$, let $\theta^i M = \theta(\theta^{i-1} M)$. Note that if M is an (r, s)-separating system, then $\theta^i M$ is also, for any i.

Theorem 3: Let $m = m_1 m_2$, with $m_2 \geq m_1$. Suppose there is a (2, 1)-separating system M_1 with m_1 rows and n_1 columns and a (2, 1)-separating system M_2 with m_2 rows and n_2 columns. Then there is a (2, 1)-separating system M with m rows and $n_1 + 2n_2$ columns.

Proof: M is shown schematically below. The rows of M are partitioned into m_1 blocks of m_2 rows each. The matrix M_{1i}, $1 \leq i \leq m_1$, is an $m_2 \times n_1$ matrix, each of whose rows is the ith row of M_1. In columns $n_1 + 1$ through $n_1 + n_2$, the $m_2 \times n_2$ matrix M_2 appears in each block. In columns $n_1 + n_2 + 1$ through $n_1 + 2n_2$, the matrix M_2 appears in each block, cycled $i - 1$ times in the ith block.

	Columns		
	$1 \quad\quad n_1$	$n_1+1 \quad n_1+n_2$	$n_1+n_2+1 \quad n_1+2n_2$
Block 1	$\uparrow m_2 \downarrow \quad M_{11}$	M_2	M_2
Block 2	$\uparrow m_2 \downarrow \quad M_{12}$	M_2	θM_2
⋮	⋮	⋮	⋮
Block m_1	$\uparrow m_2 \downarrow \quad M_{1m_1}$	M_2	$\theta^{m_1-1} M_2$

Let $(i, j; k)$ be a dichotomy. We must show that some column of M dichotomizes $(i, j; k)$.

Case 1—i, j, and k are in different blocks: Let these blocks be b_1, b_2, and b_3, respectively. There is some column of M_1, say column g, which dichotomizes $(b_1, b_2; b_3)$. Then column g of M dichotomizes $(i, j; k)$.

Case 2—i and j are in the same block, k is in a different block: Let i and j be in block b_1 and k in block b_2. As a general rule, an (r, s)-separating system is an (r', s')-separating system if $r' \leq r$ and $s' \leq s$. Thus there is some column of M_1 that dichotomizes $(b_1; b_2)$. This column dichotomizes $(i, j; k)$.

Case 3—i, j, and k are in the same block: Let $i^* = i$ mod m_2, $j^* = j$ mod m_2 and $k^* = k$ mod m_2. There is some column, say column g, of M_2 which dichotomizes $(i^*, j^*; k^*)$. Then column $g + n_1$ dichotomizes $(i, j; k)$.

Case 4—i and k are in the same block, j is in a different block: Since i and j are interchangeable, this case exhausts the possibilities. Let i and k be in block b_1, j in block b_2. What we shall do is "project" j from block b_2 to block b_1. One way to project j is to find a row j' in block b_2 which has the same values in columns $n_1 + 1$ through $n_1 + n_2$ as j has. The second way to project j is to find a row j'' which has the same values in columns $n_1 + n_2 + 1$ through $n_1 + 2n_2$ as j has. If we can find a column which dichotomizes $(i, j'; k)$ or $(i, j''; k)$, then we know that column also dichotomizes $(i, j; k)$.

Let j_1 and j_2 be the unique integers between 1 and M_2 such that $j_1 \equiv j \bmod M_2$ and $j_2 \equiv (j+b_1-b_2) \bmod M_2$. Since $m_2 \geq m_1$, we know that $j_1 \neq j_2$. Define $j' = m_2(b_1-1)+j_1$ and $j'' = m_2(b_1-1)+j_2$. Then $j' \neq j''$, and both lie in block b_1. Also, rows j and j' have the same entries in columns n_1+1 through n_1+n_2; j and j'' have the same entries in columns n_1+n_2+1 through n_1+2n_2.

Since $j' \neq j''$, they cannot both be k. Let $j' \neq k$. There is some column between n_1+1 and n_1+n_2 which dichotomizes $(i, j'; k)$, since M_2 is a (2, 1)-separating system. This column also dichotomizes $(i, j; k)$. If $j' = k$, then $j'' \neq k$. There is a column between n_1+n_2+1 and n_1+2n_2 which dichotomizes $(i, j''; k)$, since $\theta^p M_2$ is a (2, 1)-separating system for any p. This column also dichotomizes $(i, j; k)$.

Example: There is a (2, 1)-separating system [which is also a (2, 2)-separating system] with four rows and three columns, namely,

$$M = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

Using M as both M_1 and M_2 in Theorem 1, we can construct M', a (2, 1)-separated system with 16 rows and 9 columns. M' is exhibited below.

M'									
block 1	0	0	0	0	0	0	0	0	0
	0	0	0	0	1	1	0	1	1
	0	0	0	1	0	1	1	0	1
	0	0	0	1	1	0	1	1	0
block 2	0	1	1	0	0	0	1	1	0
	0	1	1	0	1	1	0	0	0
	0	1	1	1	0	1	0	1	1
	0	1	1	1	1	0	1	0	1
block 3	1	0	1	0	0	0	1	0	1
	1	0	1	0	1	1	1	1	0
	1	0	1	1	0	1	0	0	0
	1	0	1	1	1	0	0	1	1
block 4	1	1	0	0	0	0	0	1	1
	1	1	0	0	1	1	1	0	1
	1	1	0	1	0	1	1	1	0
	1	1	0	1	1	0	0	0	0

In general, let $m = 2^{2^p}$ for some integer p. Starting with the matrix M above, and using Theorem 1, we can construct a (2, 1)-separating system with m rows and $n = 3^p$ columns. The relation between m and n is

$$n = (\log_2 m)^{\log_2 3} = (\log_2 m)^{1.59}.$$

We can generalize the result of Theorem 3 to (2–2)-separating systems, with one restriction. The number of rows in a block must be odd.

Theorem 4: Let $m = m_1 m_2$, with $m_2 \geq m_1$ and m_2 odd. Suppose there is a (2, 2)-separating system M_1 with m_1 rows and n_1 columns and a (2, 2)-separating system M_2 with m_2 rows and n_2 columns. Then there is a (2, 2)-separating system M with m rows and n_1+3n_2 columns.

Proof: The plan of M is shown below. M again has its rows partitioned into m_1 blocks of m_2 rows each. The matrices M_{1i}, $1 \leq i \leq m_1$, are again $m_2 \times n_1$ matrices, each of whose rows is the ith row of M_1.

	Columns			
	1 … n_1	n_1+1 … n_1+n_2	n_1+n_2+1 through n_1+2n_2	n_1+2n_2+1 through n_1+3n_2
block 1	↑ m_2 ↓ M_{11}	M_2	M_2	M_2
block 2	↑ m_2 ↓ M_{12}	M_2	θM_2	$\theta^2 M_2$
block 3	↑ m_2 ↓ M_{13}	M_2	$\theta^2 M_2$	$\theta^4 M_2$
	⋮	⋮	⋮	⋮
block m_1	↑ m_2 ↓ M_{1m_1}	M_2	$\theta^{m_1-1} M_2$	$\theta^{2(m_1-1)} M_2$

Note that any column of M that dichotomizes $(i, j; k, l)$ also dichotomizes $(j, i; k, l)$, $(i, j; l, k)$. $(k, l; i, j)$, and four other dichotomies which are essentially the same as $(i, j; k, l)$. Let $(i, j; k, l)$ be an arbitrary dichotomy. By the above remark, there are only seven essentially different cases to consider.

Case 1—i, j, k, and l are all in different blocks: Some column among the first n_1 columns dichotomizes in this case.

Case 2—Only i and j are in the same block: Again some column among the first n_1 serves.

Case 3—i and j are in the same block, k and l are in another block: Again some column among the first n_1 serves.

Case 4—All are in the same block: Some column from n_1+1 to n_1+n_2 dichotomizes.

The remaining cases are more complicated, and a few definitions will make things easier. Again we must project rows from one block to another. There are three useful ways to do so, either by preserving the values in columns n_1+1 through n_1+n_2, in columns n_1+n_2+1 through n_1+2n_2, or in columns n_1+2n_2+1 through n_1+3n_2. Three "projection functions" P_1, P_2, and P_3 will be defined. Let b_1 and b_2 be block numbers, $b_1 \neq b_2$, and g be a row in block b_2.

Let $1 \leq g_1 \leq M_2$, $g_1 \equiv g \bmod M_2$, and define $P_1(g, b_1, b_2) = m_2(b_1-1)+g_1$. Note that $P_1(g, b_1, b_2)$ is the row in block b_1 which has the same entries in columns n_1+1 through n_1+n_2 as g has.

Let $1 \leq g_2 \leq m_2$, $g_2 \equiv (g+b_1-b_2) \bmod M_2$, and define $P_2(g, b_1, b_2) = m_2(b_1-1)+g_2$. Then $P_2(g, b_1, b_2)$ is that row in block b_1 having the same entries in columns n_1+n_2+1 through n_1+2n_2 as g has.

Let $1 \leq g_3 \leq M_2$, $g_3 \equiv (g+2b_1-2b_2) \bmod M_2$, and define $P_3(g, b_1, b_2) = m_2(b_1-1)+g_3$. $P_3(g, b_1, b_2)$ is that row in block b_1 having the same entries in columns n_1+2n_2+1 through n_1+3n_2 as g has.

Since $m_2 \geq m_1$, it should be clear that $P_1(g, b_1, b_2) \neq P_2(g, b_1, b_2)$ and $P_2(g, b_1, b_2) \neq P_3(g, b_1, b_2)$. It is also true that $P_1(g, b_1, b_2) \neq P_3(g, b_1, b_2)$, for if not, then we would have $0 = 2(b_1-b_2) \bmod m_2$. But we require that m_2 be odd, so the above relation is not possible. With these inequalities in mind, we are ready to continue with the proof.

Case 5—i, j, and k are in one block, l in another: Let i, j, and k be in block b_1 and l be in block b_2. Let $l_1 = P_1(l, b_1, b_2)$, $l_2 = P_2(l, b_1, b_2)$, and $l_3 = P_3(l, b_1, b_2)$. Since l_1, l_2, and l_3 are all different, one of these must be neither i nor j. If $l_1 \neq i$ and $l_1 \neq j$, then one of column n_1+1 through n_1+n_2 must dichotomize $(i, j; k, l_1)$. (The case $k = l_1$ is not ruled out.) This column also dichotomizes $(i, j; k, l)$. Similarly, if l_2 or l_3 is neither i nor j, we can find a column among n_1+n_2+1 through n_1+2n_2 or among n_1+2n_2+1 through n_1+3n_2, respectively, dichotomizing $(i, j; k, l)$.

Case 6—i and k are in one block, j and l in different blocks: Let i and k be in block b_1, j in b_2, and l in b_3. Define $j_q = P_q(j, b_1, b_2)$ and $l_q = P_q(l, b_1, b_3)$, for $q = 1$, 2, and 3. As mentioned, j_1, j_2, and j_3 must be distinct, and l_1, l_2, and l_3 must be distinct. Therefore, for some $q = 1$, 2 or 3, we must have both $j_q \neq k$ and $l_q \neq i$. Then, one of columns $n_1+(q-1)n_2+1$ through n_1+qn_2 must dichotomize $(i, j_q; k, l_q)$. This column also dichotomizes $(i, j; k, l)$.

Case 7—i and k are in one block, j and l in another block: Let i and k be in block b_1, j and l in b_2. Define $j_q = P_q(j, b_1, b_2)$ and $l_q = P_q(l, b_1, b_2)$, $q = 1$, 2, and 3. The argument proceeds as in Case 6.

The case where m_2 is even will be of most use to us. A trivial corollary to Theorem 4 is as follows.

Corollary: If m_2 is even, $m_1 < m_2$, and there are m_1-row, n_1-column and m_2-row, n_2-column (2, 2)-separating systems, then there is an $m_1(m_2-1)$-row, (n_1+3n_2)-column (2, 2)-separating system.

Proof: If there is an m_2-row, n_2-column (2, 2)-separating system, there is surely one with m_2-1 rows and n_2 columns.

Construction of Specific (2, 2)-Separating Systems

To construct good (2, 2)-separating systems using Theorem 4, we need good starting matrices with small numbers of rows. These can be constructed by the following arguments:

a) There exist 4-row, 3-column and 8-row, 7-column (2, 2)-separating systems.
b) There is a 10-row, 11-column (2, 2)-separating system. (This was discovered by Roth [6] with the aid of a computer.)
c) There is a 16-row, 14-column (2, 2)-separating system [from a) and Theorem 2].

TABLE I

2-2 SEPARATING SYSTEMS KNOWN OR CONSTRUCTED BY THEOREMS 1, 2, AND 3

Rows	Columns
4	3
8	7
10	11
16	14
20	20
32	24
64	39

TABLE II

2-2 SEPARATING SYSTEMS CONSTRUCTED BY THEOREM 4

Rows	Factors of Rows	Columns
49	7×7	28
72	8×9	40
81	9×9	44
120	8×15	49
150	10×15	53
225	15×15	56
304	16×19	74
361	19×19	80
496	16×31	86
620	20×31	92
961	31×31	96
980	20×49	104
1568	32×49	108
2403	49×49	112
3087	49×63	145
3479	49×71	148
3969	63×63	156
4544	64×71	159

d) There is a 10-row, 8-column (2, 1)-separating system [6].

We have exhibited a (2, 1)-separating system with 16 rows and 9 columns. Thus, using c) and Theorem 2, there is a 32-row, 24-column (2, 2)-separating system. Also using Theorem 1, there is a 32-row, 14-column (2, 1)-separating system. Thus, using Theorem 2, we can construct a 64-row, 39-column (2, 2)-separating system. Using b), d), and Theorem 2, there is a 20-row, 20-column (2, 2)-separating system. These results are summarized in Table I.

Starting with these values, we can apply Theorem 4 or its Corollary in various ways. Some of these results are given in Table II.

In general, suppose i is an odd integer and that there is an i-row, $f(i)$-column (2, 2)-separating system. Let $m = i^{2^p}$ for some integer p. Then by Theorem 2, there is an m-row (2, 2)-separating system with $n = f(i)4^p$ columns. The relationship between m and n can be expressed as

$$n = \frac{f(i)}{(\log_2 i)^2} (\log_2 m)^2.$$

Thus, we can find an infinity of (2, 2)-separating systems whose number of columns is proportional to the square of the logarithm of the number of rows.

TABLE III
VALUES OF THE PROPORTIONALITY CONSTANT

i	$f(i)$	$f(i)/(\log_2 i)^2$
3	3	1.19
7	7	0.88
15	14	0.91
31	24	0.98
63	39	1.09

The constant of proportionality, $f(i)/(\log_2 i)^2$, is usually about 1. Some actual values are shown in Table III.

We have thus shown how universal STT assignments can be derived which are relatively efficient when compared with previously known bounds. However, these assignments are far from optimal as the following theorem will show.

Theorem 5: There exists a universal n-state STT assignment with m variables, where m is any integer which satisfies

$$m > \frac{\log_2 3\binom{n}{4}}{\log_2 \frac{16}{14}} \geq 21 \log_2 n = 21S_0.$$

Proof:

	1 ···	···	··· m
1	···	0 ··· 1	···
2	···	0 ··· 1	···
3	···	1 ··· 0	···
4	···	1 ··· 0	···
·	···	···	···
·	···	···	···
·	···	···	···
n	···	···	···

Consider the class of $n \times m$ binary arrays. There are 2^{nm} such arrays. Now consider the dichotomy (1, 2; 3, 4). An array will cover this dichotomy if and only if it has one of the columns shown above.

There are thus $(2^4-2)^m 2^{(n-4)m}$ arrays which do not cover this dichotomy. For n states there are

$$3\binom{n}{4}$$

dichotomies. There are thus at most

$$\left(3\binom{n}{4}\right) 14^m 2^{(n-4)m}$$

arrays which do not cover some dichotomy, and hence at least

$$2^{nm} - \left(3\binom{n}{4}\right) 14^m 2^{(n-4)m}$$

arrays which cover all dichotomies. We wish to find the smallest value of n such that this number is positive, i.e., such that

$$2^{nm} > 3\binom{n}{4} 14^m 2^{(n-4)m}$$

$$1 > 3\binom{n}{4}\left(\frac{14}{16}\right)^m$$

$$\left(\frac{16}{14}\right)^m > 3\binom{n}{4}.$$

Thus, for

$$m > \frac{\log_2 3\binom{n}{4}}{\log_2 \frac{16}{14}}$$

there is at least one array which covers all dichotomies and the theorem is proved.

The proof of Theorem 5 can easily be extended to (r, s)-separating systems yielding bounds of

$$m \geq \frac{(r+s)\log n}{\log\left(\frac{1}{1-2^{1-r-s}}\right)}.$$

These bounds are nonconstructive, and probably play a role similar to the Gilbert bound [5] of coding theory. Although Gilbert and others showed in 1952, by a nonconstructive argument, that error-correcting codes with very reasonable redundancy rates exist, constructive procedures for realizing such codes in all cases have yet to be discovered.

The bound itself can probably be substantially improved. For instance, for (1, 1)-separating systems the derived bound is $n \geq 2S_0$. Since this is not exact (as shown by Lemma 1), it seems quite likely that the constant of Theorem 5 can also be improved.

Unger has shown [4] that any flow table without essential hazards can be realized without delay elements using a special STT assignment. It can be shown that a (3, 2)-separating system is sufficient to satisfy the additional constraints required by Unger.

The results obtained so far in this paper have all been under the assumption that each state had only one coding assigned to it. However, it is possible to have STT assignments in which several codings are assigned to the states but all variables which must change in any transition are allowed to change simultaneously without introducing critical races. We now show that this generalization of STT assignments enables a further improvement in the bounds of universal assignments for at least some values of n.

Consider the state assignment shown below.

	y_1	y_2	y_3	y_4	y_5
1	0	0	0	0	0
1′	1	1	1	1	1
2	0	1	1	0	0
2′	1	0	0	1	1
3	1	1	0	0	1
3′	0	0	1	1	0
4	0	0	0	1	1
4′	1	1	1	0	0
5	1	0	1	1	0
5′	0	1	0	0	1
6	0	1	1	1	1
6′	1	0	0	0	0

This is an assignment for a 6-state table which contains two codings for each state. The two codings assigned to any state are complements of each other. (Such assignments will be referred to as complement codes.) We will now show that this assignment is a universal 6-state STT assignment. This is done in the following manner. Associate with each pair of states i, j two subcubes consisting of those variables which stay fixed in transitions from i to j and i' to j' if $D(i,j)<3$ [where $D(i, j)$ is the Hamming distance from i to j], or from i to j' and i' to j if $D(i, j)>3$. Since i, i' and j, j' are complements, at least three variables stay fixed in each transition. The subcubes obtained in this manner are shown below.

12	0--00	23′	0-1-0	34′	11-0-	45′	0-0-1
1′2′	1--11	2′3	1-0-1	3′4	00-1-	4′5	1-1-0
13′	00--0	24′	-1100	35′	-1001	46	0--11
1′3	11--1	2′4	-0011	3′5	-0110	4′6′	1--00
14	000--	25′	01-0-	36′	1-00-	56′	10--0
1′4′	111--	2′5	10-1-	3′6	0-11-	5′6	01--1
15′	0-00-	26	011--				
1′5	1-11-	2′6′	100--				
16′	-0000						
1′6	-1111						

The reader may verify that the subcube associated with either ij transition is disjoint from the subcube associated with either kl transition if both k and l are distinct from i and j. Hence, there are no critical races and the assignment is a valid universal 6-state assignment requiring only 5-state variables. It was previously shown that the universal 6-state assignment using only one coding per state required 7-state variables. We have also derived a universal 14-state assignment using 11-state variables.

The assignment is

$$T^* = \begin{bmatrix} T & T/2 \\ T & \overline{T}/2 \end{bmatrix}$$

where T is the equidistant code

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

and $T/2$ is the first four columns of T.

It seems quite likely that constructive codes superior to those of Theorems 2 and 4 can be derived for the class of complement STT assignments. It is also possible that more general assignments in which more than two codings are assigned to states might yield still better bounds but no systematic procedures for making such codings have been developed at the present time. The Hamming assignment [3] may be viewed as an extreme case of this kind of assignment in which $2^{m-1}/m$ codings are assigned to each of the m states. However, this assignment yields inferior bounds.

In this paper we have considered the problems of deriving bounds on STT assignments which are valid for any m-state flow table. We have presented two classes of constructive codes and have proved that the actual bound is proportional to S_0. We have also shown that the use of STT codes with more than one coding per state will allow some improvement in the number of variables required for the class of constructive codes.

References

[1] C. N. Liu, "A state variable assignment method for asynchronous sequential switching circuits," *J.ACM*, vol. 10, pp. 209–216, April 1963.

[2] J. H. Tracey, "Internal state assignment for asynchronous sequential machines," *IEEE Trans. Electronic Computers*, vol. EC-15, pp. 551–560, August 1966.

[3] S. H. Caldwell, *Switching Circuits and Logical Design*. New York: Wiley, 1958.

[4] S. H. Unger, "A row assignment procedure for delay-free realizations of flow tables without essential hazards," *1966 Proc. 7th Ann. Symp. on Switching and Automata Theory*, pp. 154–159.

[5] E. N. Gilbert, "A comparison of signalling alphabets," *Bell Sys. Tech. J.*, vol. 31, 1952.

[6] R. H. Roth (private communication).

Reprinted from IEEE TRANSACTIONS ON *COMPUTERS*

Volume C-18, Number 6, June, 1969

pp. 541-547