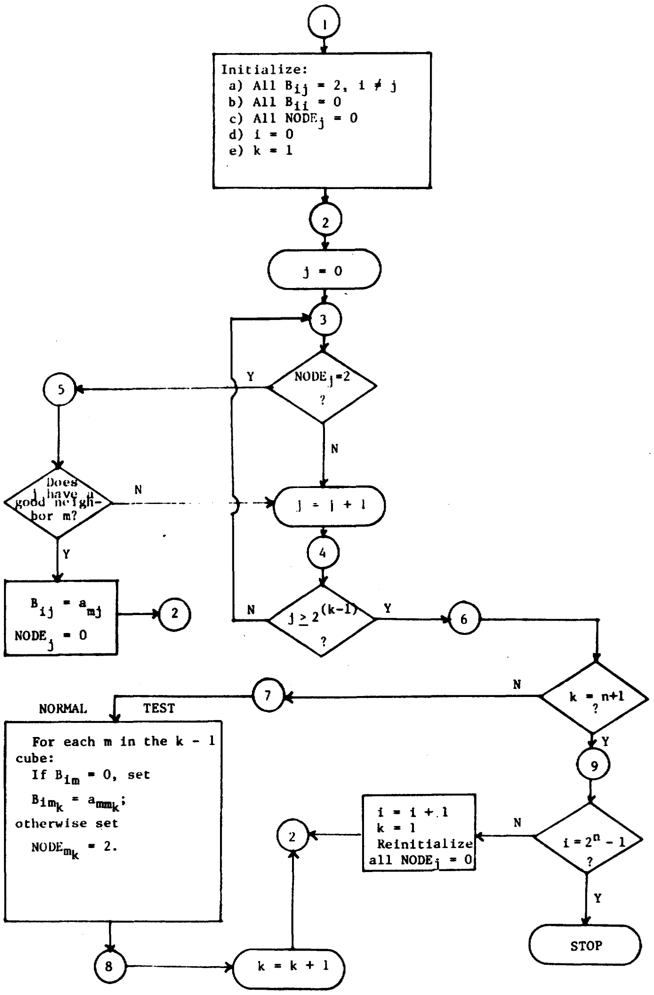

Fig. 4. Algorithm 1 flowchart.

```
         ↙j = 0                                ↙j = 0
i = 0 → 010*0***        0 → 1        i = 0 → 01010010
        00*0*1**        0 → 2                00000110
        0*01**1*        2 → 3                01010010
        *100***0        0 → 4                01000010
        0***001*        2 → 6                01010010
        *1**00*0        4 → 5                01010010
        **0*1*00        5 → 7                01011000
        ***1*010                             01010010
          (a)            (b)                    (c)
```

Fig. 5. Test algorithm operation. (a) A array. (b) Order of determining the elements of B_o. (c) B array.

```
        ↓j = 0
i=0 →  011*1***          01121222
       10*1*1**          10212122
       1*01**1*          12012212
       *110***0          21101000
       1***011*          12220112
       *1**10*0          21101000
       **1*1*00          21101000
       ***0*000          21101000

       a Array           B Array
```

Fig. 6. A array and B array for processor 1, 2, and 4 faulty.

disjoint paths were shown to exist between any two nodes in the cube. In doing so the proof illustrates how the message algorithm described in [1] can be made reconfigurable.

If the test table (B array) is constructed and interpreted by a special processor, this constitutes a system hardcore. In [12] we describe an easily testable logic structure which can perform this construction and interpretation function.

References

[1] H. Sullivan and T. R. Bashkow, "A large scale, homogeneous, fully distributed parallel machine, I," in *Proc. IEEE 1977 Comput. Arch.*, 1977, pp. 105-117.

[2] H. Sullivan, T. R. Bashkow, and D. Klappholz, "A large scale, homogeneous, fully distributed parallel machine, II," in *Proc. IEEE 1977 Comput. Arch.*, 1977, pp. 118-124.

[3] K. J. Thurber, "Interconnection networks—A survey and assessment," in *Proc. Nat. Comput. Conf.*, 1974, pp. 909-919.

[4] F. Harary, *Graph Theory.* Reading, MA: Addison-Wesley, 1969.

[5] K. Menger, "Zur allgemeinen kurventheorie," *Fundamentals Math.*, vol. 10, pp. 96-115, 1927.

[6] H. Whitney, "Congruent graphs and the connectivity of graphs," *Amer. J. Math.*, vol. 54, pp. 150-168, 1932.

[7] L. W. Beinike and F. Harary, "The connectivity function of a graph," *Mathematika*, vol. 14, pp. 197-202, 1967.

[8] F. P. Preparata, G. Metze, and R. T. Chien, "On the connection assignment problem of diagnosable systems," *IEEE Trans. Comput.*, vol. C-16, pp. 848-854, Dec. 1967.

[9] G. G. L. Meyer and G. M. Masson, "An efficient fault diagnosis algorithm for symmetric multiple processor architectures," *IEEE Trans. Comput.*, vol. C-27, pp. 1059-1063, Nov. 1978.

[10] S. L. Hakimi and A. T. Amin, "Characterization of the connection assignment of diagnosable systems," *IEEE Trans. Comput.*, vol. C-23, pp. 86-88, Jan. 1974.

[11] J. R. Armstrong and F. G. Gray, "Fault diagnosis in a Boolean *n* cube array of microprocessors," Dep. Elec. Eng., Virginia Polytech. Inst. and State Univ., Tech. Rep. EE-8103.

[12] ——, "A multivalued, global, majority voter," in *Proc. 10th Int. Symp. Multiple-Valued Logic*, pp. 268-271.

[13] J. G. Kuhl and S. M. Reddy, "Distributed fault tolerance for large multiprocessor systems," in *Proc. 7th Int. Symp. Comput. Arch.*, May 1980, pp. 23-30.

Speed-Efficiency-Complexity Tradeoffs in Universal Diagnosis Algorithms

JON T. BUTLER

***Abstract*—Expressions are derived for the average number of steps required (speed) and the average number of fault-free units replaced (efficiency) when universal diagnosis algorithms are applied to systems of various degrees of interconnection (complexity). Specifically, two algorithms proposed by Smith [4] are considered. It is shown, for example, that there is a clear tradeoff between the two algorithms; one is much faster, while the other is more efficient.**

***Index Terms*—Diagnosis algorithms, fault diagnosis, graph model of system diagnosis, multiple faults, system reliability, test interconnections.**

I. Introduction

Considerable study has been devoted to the diagnosis of faults in the multiprocessor model proposed by Preparata, Metze, and Chien [1]. Variations on this model include a different form of test invalidation [2], replacement of fault-free units [3]-[6], probabilistic considerations [7], [8], and application to other types of digital systems [9].

This correspondence focuses on the tradeoffs between speed and efficiency of two diagnosis algorithms proposed by Smith [4]. Specifically, the average number of fault-free processors replaced (efficiency) and the average number of steps required (time) are used to

Manuscript received January 17, 1980; revised March 23, 1981. This work was supported by the National Science Foundation under Grant MCS-7600326-A01 and a National Research Council Senior Postdoctoral Associateship tenured at the Naval Postgraduate School, Monterey, CA.

The author is with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60201.

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE **MAR 1981**	2. REPORT TYPE	3. DATES COVERED

4. TITLE AND SUBTITLE **Speed-Efficiency-Complexity Tradeoffs in Universal Diagnosis Algorithms**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Postgraduate School,Department of Electrical and Computer Engineering,Monterey,CA,93943**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S) 11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Expressions are derived for the average number of steps.required (speed) and the average number of fault-free units replaced (efficiency) when universal diagnosis algorithms are applied to systems of various degrees of interconnection (complexity). Specifically, two algorithms proposed by Smith [4] are considered. It is shown, for example, that there is a clear tradeoff between the two algorithms; one is much faster, while the other is more efficient.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**7**	

compare the two algorithms. It is shown that a clear tradeoff exists. One algorithm is significantly faster but less efficient, while the opposite is true of the other algorithm. Further, the speed of diagnosis is shown to be significantly less dependent on system complexity than is efficiency.

II. Notation and Introductory Concepts

A *system* consists of a directed graph in which a binary weight is associated with each arc. The nodes, $u_0, u_1, \cdots$, and u_{n-1}, are viewed as *units* or processors and arcs between nodes as *tests* between units. Thus, u_i tests u_j if there exists a directed arc from u_i to u_j. The weight associated with this arc is viewed as a *test outcome* and is 0(1) if u_i finds u_j to be fault-free (faulty). The set of all test outcomes is called a *syndrome*. Test outcomes are generated as follows. If u_i is fault-free, the test outcome is a correct assessment of u_j; that is, the test outcome is 0(1) if u_j is fault-free (faulty). However, if u_i is faulty, the test outcome is invalidated and this can occur in two ways. If *symmetric invalidation* [1] is assumed, the test outcome of a test by a faulty unit is (arbitrarily) 0 or 1. If *asymmetric invalidation* [2] is assumed, then the test outcome is either 0 or 1 if the tested unit is fault-free, but is 1 if it is faulty. Thus, for the case of asymmetric invalidation, a faulty unit fails all its tests.

The object of a diagnosis algorithm is to identify faulty units from a given syndrome. A *step* in the algorithm consists of identifying units to be replaced, replacing them by fault-free units, and generating the new syndrome. The diagnosis is complete when the syndrome consisting of all 0's obtained. It is assumed that:

1) the state of a unit remains unchanged during a diagnosis unless it is faulty and is replaced by a fault-free unit, and

2) unreliable test outcomes remain unchanged as long as the both testing and tested units are unchanged.

The two replacement strategies [4] to be considered are as follows.

Algorithm 1: Replace all units which fail *at least one test* and have not been replaced in a previous step.

Algorithm 2: Replace all units which fail the *maximum number of tests* and have not been replaced in a previous step.

Fig. 1 shows the sequence of syndromes which occurs, given the initial syndrome of Fig. 1(a), when Algorithm 1 is applied and symmetric invalidation is assumed. Note that only one step (one application of the diagnosis strategy) is required and all units are replaced (replaced units are indicated by an asterisk). Fig. 2 shows the use of Algorithm 2 for the same initial syndrome. It is interesting to note that Algorithm 2 requires one more step, but replaces one less fault-free unit. In the next section, we will consider the average number of units replaced and the average number of steps required.

A system of n units is a *design* $D_{\delta t}(n)$ if unit u_i tests u_j iff $j - i = \delta m \bmod n$, where $1 \leq m \leq t$. For example, Fig. 1 shows design $D_{12}(5)$. It has been shown [1] that designs where δ and n are relatively prime are equivalent to $D_{1t}(n)$. It will be convenient to assume $\delta = 1$.

III. Average Number of Fault-Free Units Replaced

Let $F_{\alpha i}(f, I)$ denote the average number of fault-free units replaced in an interconnection I containing f faulty units. It is assumed that Algorithm α, for $\alpha = 1$ or 2 is applied at all steps and either symmetric ($i = S$) or asymmetric ($i = A$) invalidation is in effect. It is assumed further that each of the $\binom{n}{f}$ fault patterns is equally likely for fixed f and that for an arbitrary fault pattern each syndrome in the set of all possible syndromes is equally likely. We have, therefore,

$$F_{\alpha i}(f, I) = \frac{1}{\binom{n}{f}} \sum_{fp} \frac{1}{N(fp)} \sum_{S(fp)} FF(S(fp)) \qquad (1)$$

where $\sum_{fp}$ is a sum over all fault patterns, $N(fp)$ is the number of

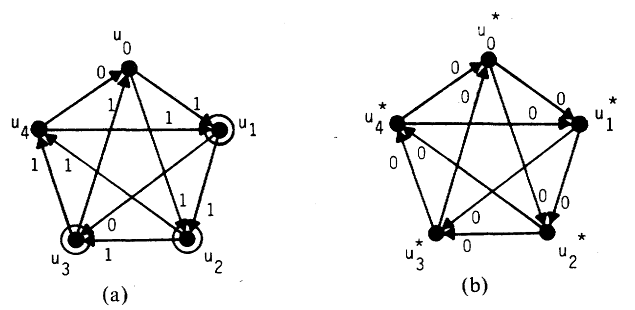

Fig. 1. Example of the use of Algorithm 1 in design $D_{12}(5)$ assuming symmetric invalidation.

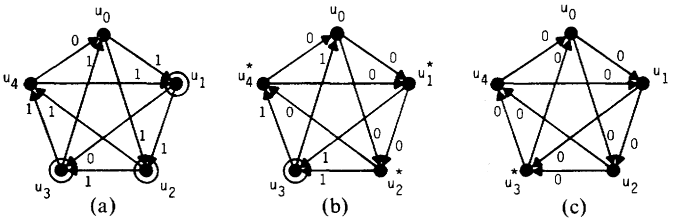

Fig. 2. Example of the use of Algorithm 2 in design $D_{12}(5)$ assuming symmetric invalidation.

syndromes associated with fault pattern fp, $\sum_{S(fp)}$ is the sum over all syndromes of fp, and $FF(S(fp))$ is the number of fault-free units replaced when $S(fp)$ is the original syndrome.

A. Algorithm 1 with Symmetric Invalidation

Consider the diagnosis of a design when symmetric invalidation is assumed and Algorithm 1 is applied. A fault-free unit u will be replaced in the first step if in the initial syndrome there is a test outcome of a test of u by a faulty unit which incorrectly fails u. Similarly, in subsequent steps the only way a fault-free unit can be replaced is if it fails a test by a faulty unit. This, however, implies that the test outcome of a test by a faulty unit has changed from a 0 to a 1 during the course of the diagnosis contradicting an earlier assumption. Fault-free units are, therefore, replaced in the first step only. Thus, $F_{\alpha i}(f, I)$ for $\alpha = 1$ and $i = S$, can be calculated by considering just initial syndromes. When $I = D_{1t}(n)$, the number of syndromes $N(fp)$ for any fault pattern fp is 2^{ft}, since there are f faults each producing t unreliable tests whose outcomes occur in two ways. Thus, for $I = D_{1t}(n)$, (1) becomes

$$F_{\alpha i}(f, D_{1t}(n)) = \frac{1}{\binom{n}{f}_{2}^{\mathrm{ft}}} \sum_{fp} \sum_{S(fp)} FF(S(fp)). \qquad (2)$$

The double sum of (2) for $\alpha = 1$ and i = S can be solved by observing that each unit u in the design contributes a single replaced fault-free unit when

1) u is fault-free,

2) at least one faulty unit tests u, and

3) at least one of the unreliable tests of u produces a "fail" outcome.

There are $\binom{t}{k}\binom{n-t-1}{f-k}$ distributions of faulty units such that k of them test u, since there are $\binom{t}{k}$ ways in which k of the t units testing u are faulty and $\binom{n-t-1}{f-k}$ ways in which the $n - t - 1$ remaining units are faulty. For each of these fault patterns, there are $2^k - 1$ different assignments of test outcomes to unreliable tests of u which indicate u is faulty and 2^{ft-k} different assignment of test outcomes for the remaining unreliable tests. Thus,

$$F_{1S}(f, D_{1t}(n)) = \frac{1}{\binom{n}{f}_{2}^{\mathrm{tf}}} \sum_{k=1}^{\min\{f,t\}} n \binom{t}{k}\binom{n-t-1}{f-k} (2^k - 1)(2^{ft-k}). \quad (3)$$

Since $\binom{n-t-1}{f-k} = 0$ when $k > f$, (3) remains unchanged if the upper limit of the sum is replaced by t. Doing this and rearranging yields

$$F_{1S}(f, D_{1t}(n)) = \frac{n}{\binom{n}{f}} \sum_{k=1}^{t} \binom{t}{k}\binom{n-t-1}{f-k} \frac{2^k-1}{2^k}. \quad (4)$$

For a specific design, (4) reduces to a closed-form expression. For example, in the case of $D_{11}(n)$, a single loop of n units

$$F_{1S}(f, D_{11}(n)) = \frac{f(n-f)}{2(n-1)}. \quad (5)$$

It is interesting to note that $F_{1S}(f, D_{11}(n))$ is symmetric with respect to f, increasing from 0 at $f = 0$ to a maximum at $f = n/2$ and then decreasing to 0 again at $f = n$. The increase in the average number of replaced fault-free units with increasing f is due to the additional unreliable data on which fault-free units are incorrectly replaced. For $f > n/2$, this phenomena still exists; however, as f increases there are fewer fault-free units available for replacement. Thus, $F_{1S}(f, D_{11}(n))$ decreases with increasing f for this case.

For designs $D_{12}(n)$, (4) reduces to

$$F_{1S}(f, D_{12}(n)) = F_{1S}(f, D_{11}(n)) \left[\frac{2n - \frac{f}{2} - \frac{5}{2}}{n-2}\right] \quad (6)$$

which is not symmetric with respect to f. Fig. 3 shows how the average number of fault-free units replaced by Algorithm 1 (shown in solid lines) in design $D_{1t}(n)$ varies with the number of faults f, for $1 \leq t \leq 3$ and $t + 1 \leq n \leq 7$. The bell-shaped curve associated with the plot of $F_{1S}(f, D_{11}(n))$ versus f is also a characteristic of other designs as well.

Note that designs with a higher value of t have a higher average number of replaced fault-free units. It is straightforward to show that

$$F_{1S}(f, D_{1t}(n)) \sim F_{1S}(f, D_{11}(n)) \qquad \text{when } \frac{f}{n} \simeq 0 \quad (7)$$

and

$$F_{1S}(f, D_{1t}(n)) \sim \frac{2^t - 1}{2^{t-1}} F_{1S}(f, D_{11}(n)) \qquad \text{when } f \simeq n \quad (8)$$

where $g(n) \sim h(n)$ means $\lim_{n \to \infty} \frac{g(n)}{f(n)} = 1$. Thus, with respect to the tradeoff between the number of test links (complexity) and the number of fault-free units replaced (efficiency) in designs, there is considerable advantage in the case of Algorithm 1 to using as few links as possible. This is true of general systems as well. Bender [10] has shown that the addition of one test link to an arbitrary system increases the average number of fault-free units replaced.

B. Algorithm 2 with Symmetric Invalidation

The calculation of the average number of fault-free units replaced when Algorithm 2 is applied and symmetric invalidation is assumed is complicated by the fact that fault-free units can be replaced in steps after the first. For example, in the diagnosis shown in Fig. 4, fault-free unit u_1 is replaced in step 2. Thus, a calculation of $F_{2S}(f, D_{1t}(n))$ must, in general, enumerate fault-free units replaced in fault patterns subsequent to the initial fault pattern. Therefore, (2) is not applicable in the general case.

However, (2) can be used when the number of faulty units is small enough to guarantee replacement at the first step only. In addition, it will be convenient to assume that the number of faulty units is small enough so that when a fault-free unit is replaced, there will be a faulty unit tested exclusively by fault-free units. Thus, the maximum number of tests some unit u_1 fails is t initially and throughout the diagnosis. Since the number of tests a fault-free unit can fail will remain the same or decrease during the diagnosis, a fault-free unit is replaced only when it fails t tests and only in the first step. A firm upper bound B on f, the number of faulty units in which every replacement of a fault-free unit is accompanied by at least one faulty unit which is tested by fault-free units exclusively, is shown in [5] to be

$$B = \frac{n}{t} + t - 3, \qquad \text{for } t > 2 \text{ and } n \geq t + 1, \quad (9)$$

$$B = \frac{n-1}{2} \qquad \text{for } t = 2, \quad (10)$$

and

$$B = n - 1 \qquad \text{for } t = 1. \quad (11)$$

The calculation of $F_{2S}(f, D_{1t}(n))$ for $f \leq B$ can be accomplished by observing that, if t faulty units test some fault-free unit, there are $\binom{n-t-1}{f-t}$ ways to distribute of the remaining $f - t$ faults to the remaining $n - t - 1$ units. For each of these fault patterns, there are $ft - t = t(f - 1)$ unreliable tests of units other than u_1. Since these test outcomes can occur in two ways, there are $\binom{n-t-1}{f-t} 2^{t(f-1)}$ syndromes in fault patterns in $D_{1t}(n)$ which result in the replacement of u_i when it is fault-free. The double sum of (2) for $\alpha = 2$ and $i = S$ is just $n\binom{n-t-1}{f-t} 2^{t(f-1)}$. Thus, the average number of fault-free units replaced is

$$F_{2S}(f, D_{1t}(n)) = \frac{n\binom{n-t-1}{f-t}}{2^t\binom{n}{f}}, \qquad \text{for } f \leq \frac{n}{t} + t - 3 \text{ if } t > 2$$

$$f \leq \frac{n-1}{2} \qquad \text{if } t = 2,$$

and

$$f \leq n - 1 \qquad \text{if } t = 1. \quad (12)$$

For the special cases of $t = 1$ and 2, (12) reduces to the following expressions

$$F_{2S}(f, D_{11}(n)) = \frac{(n-f)f}{2(n-1)}, \qquad \text{for } f \leq n - 1 \quad (13)$$

and

$$F_{2S}(f, D_{12}(n)) = F_{2S}(f, D_{11}(n)) \frac{f-1}{2(n-2)}, \qquad \text{for } f \leq \frac{n-1}{2}. \quad (14)$$

Note that $F_{2S}(f, D_{11}(n)) = F_{1S}(f, D_{11}(n))$, as it should, since Algorithms 1 and 2 are identical in $D_{11}(n)$. It is interesting to observe that contrary to the case for Algorithm 1, there is a reduction in the average number of fault-free units replaced as t increases. Because of the 2^t term in the denominator, this reduction is large for a moderate increase in t.

Fig. 3 shows how the average number of fault-free units replaced in various designs when Algorithm 2 is applied compares to the number replaced when Algorithm 1 is applied. Even for small t, the difference between the two algorithms is substantial. The values of $F_{2S}(f, D_{1t}(n))$ not covered by (12) were solved from a computer enumeration of all fault patterns and syndromes.

It is interesting to note that the value of $F_{2S}(f, D_{1t}(n))$ as expressed

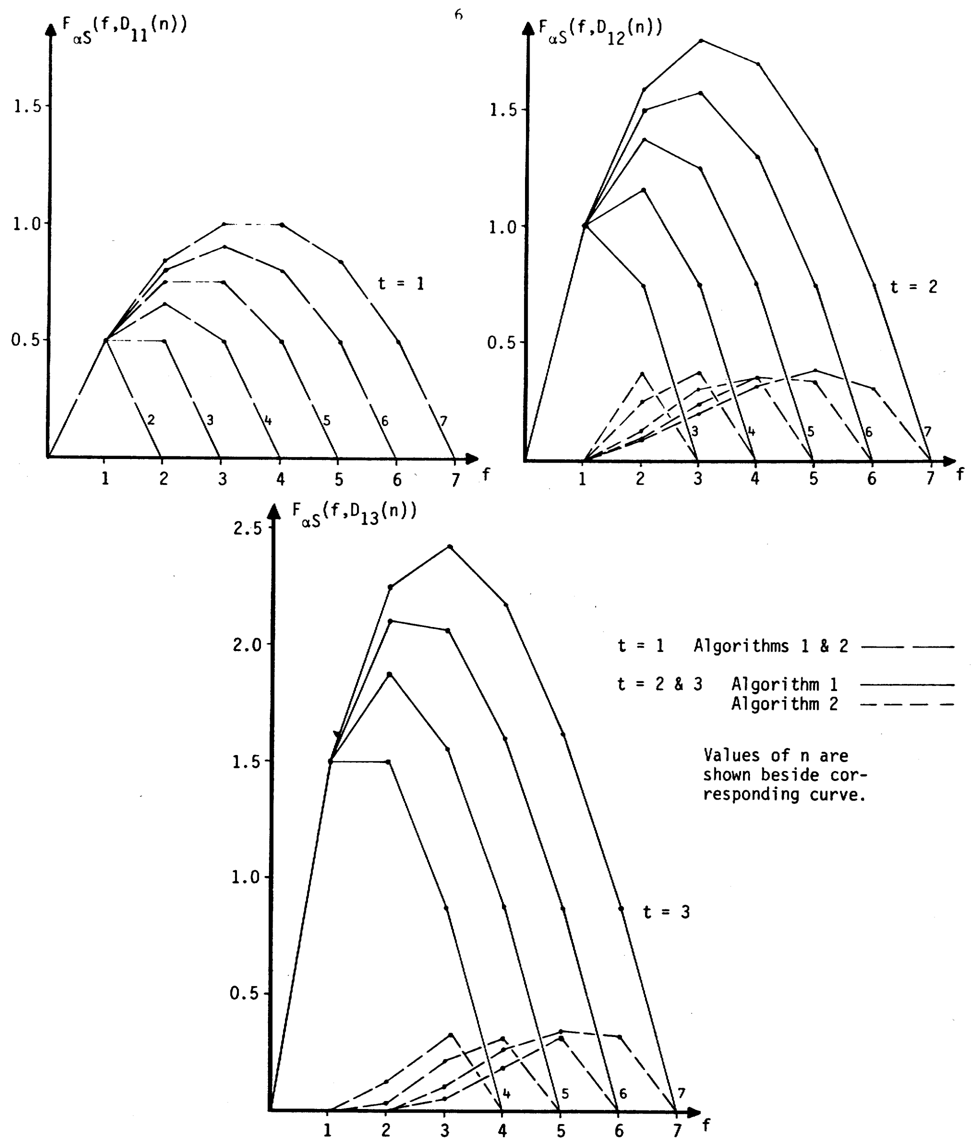

Fig. 3. $F_{\alpha S}(f, D_{1t}(n))$, the average number of fault-free units replaced in design $D_{1t}(n)$ versus f, the number of faulty units, for Algorithms 1 and 2 and symmetric invalidation.

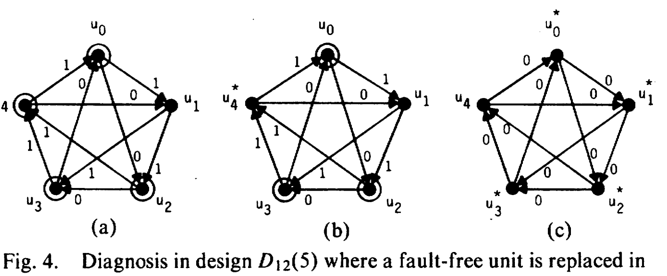

Fig. 4. Diagnosis in design $D_{12}(5)$ where a fault-free unit is replaced in step 2 using Algorithm 2 and assuming symmetric invalidation.

in (12) gives correct values when $B < f \le t$. That is, when $f < t$, (12) yields $F_{2S}(f, D_{1t}(n)) = 0$, which is correct since no fault-free unit is ever replaced when there are fewer faults than tests on a unit. Note that $B < t$ iff $n < 3t$. Thus, (12) is correct whenever $f \le \max \left\{\frac{n}{t} + t - 3, t\right\}$.

C. Algorithm 1 with Asymmetric Invalidation

Recall that when asymmetric invalidation is assumed, a faulty unit will fail all of its tests regardless of the number of faulty units which test it. No faulty units can be "masked" by other faulty units, as in the case of symmetric invalidation. Thus, one step only is required to completely diagnose a system.

Under Algorithm 1 a unit is replaced if it fails at least one test. Thus, when asymmetric invalidation is assumed, it is possible that a unit will be replaced which is certainly not faulty, specifically units which pass at least one test. Therefore, Algorithm 1 is not appropriate when asymmetric invalidation is assumed. However, for completeness, the average number of fault-free units replaced in a design $D_{1t}(n)$ with f faults is

$$F_{1A}(f, D_{1t}(n)) = F_{1S}(f, D_{1t}(n)) \tag{15}$$

since symmetric and asymmetric invalidation are identical with respect to fault-free units. The values of $F_{1A}(f, D_{1t}(n))$ for various t, f, and n are plotted in Fig. 5 in solid lines.

D. Algorithm 2 with Asymmetric Invalidation

On the other hand, Algorithm 2 is appropriate when asymmetric invalidation is assumed. $F_{2A}(f, D_{1t}(n))$ can be calculated from (2) by observing that a fault-free unit u_i will be replaced only when it is tested exclusively by faulty units. Thus, of the total $\binom{n}{f}$ fault patterns only $\binom{n-t-1}{f-t}$ involve replacement of u_i. Let fp be a fault pattern

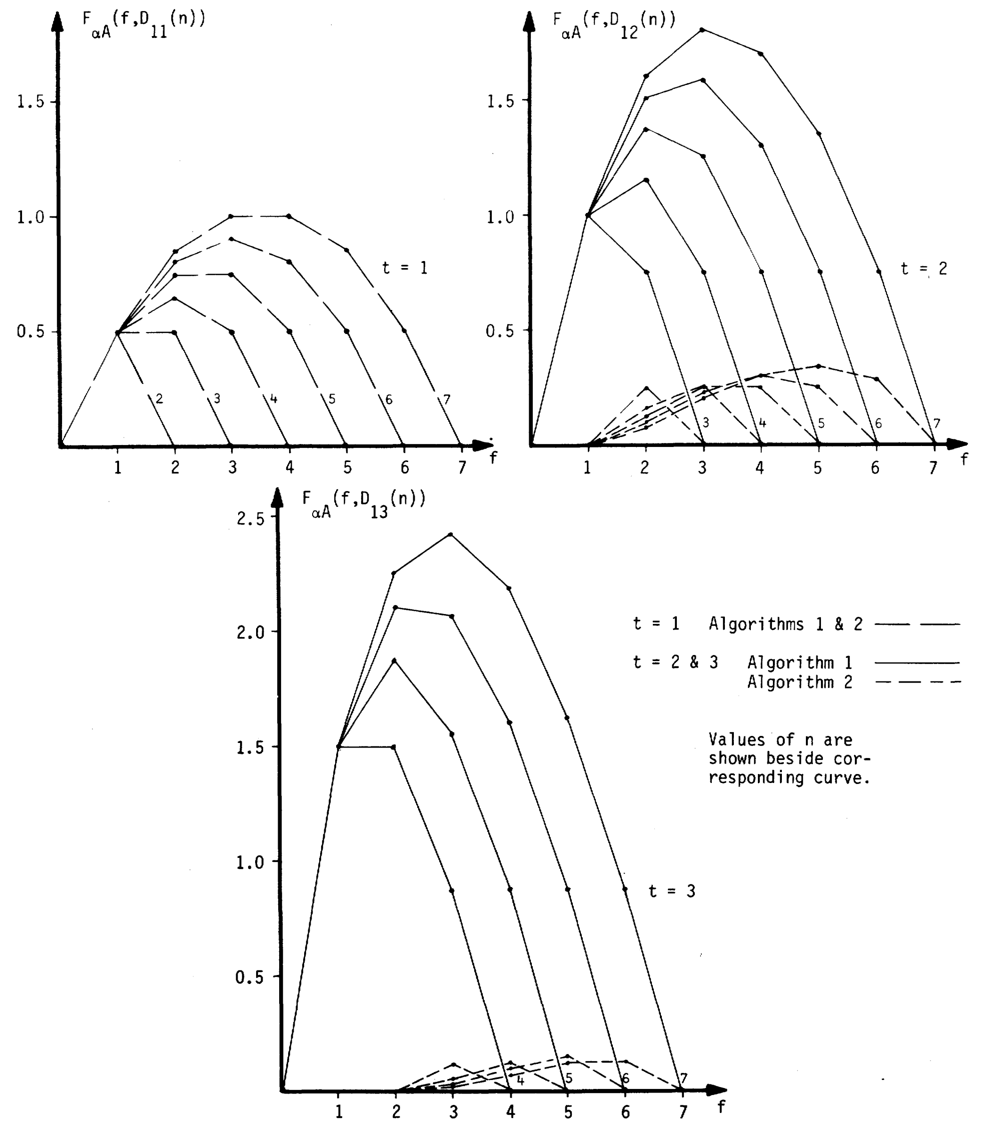

Fig. 5. $F_{\alpha S}(f, D_{1t}(n))$, the average number of fault-free units replaced in design $D_{1t}(n)$ versus f, the number of faulty units, for Algorithms 1 and 2 and symmetric invalidation.

in which u_i is replaced and $K(fp)$ the number of test links in fp in which a faulty unit tests a fault-free unit. Because asymmetric invalidation is assumed, these are the only links whose test outcomes are uncertain. Thus, the number of syndromes $N(fp)$ associated with fp is given as

$$N(fp) = 2^{K(fp)}.$$

The number of syndromes in which u_i is replaced is $2^{K(fp)-t}$ since t of the $K(fp)$ test outcomes, those associated with tests of u_i, must be 1. Therefore, the contribution of u_i to

$$\frac{1}{N(fp)} \sum_{S(fp)} FF(S(fp))$$

of (2) is just $1/2^t$ and it follows that

$$F_{2A}(f, D_{1t}(n)) = \frac{n}{\binom{n}{f}} \frac{1}{2^t} \binom{n-t-1}{f-t}. \tag{16}$$

Note that $F_{2A}(f, D_1t(n))$ is identical to $F_{2S}(f, D_{1t}(n))$ when $f \leq \max \left\{\frac{n}{t} + t - 3, t\right\}$. Fig. 5 shows how $F_{2A}(f, D_{1t}(n))$ varies with f for various values of n and t (shown as dotted lines). It is interesting to note that when f is small, there is no advantage of asymmetric over symmetric invalidation, with respect to the average number of fault-free units replaced. This is a somewhat surprising result.

IV. Average Number of Steps Required

Of interest in this section is $T_{\alpha i}(f, I)$, the average number of steps required to replace all f faulty units in a system I, where I is design $D_{1t}(n)$. It is assumed that Algorithm α for $\alpha = 1$ or 2 is applied and either symmetric ($i = S$) or asymmetric ($i = A$) invalidation is in effect. Again, it is assumed that each fault pattern is equally likely and that for any arbitrary fault pattern, each syndrome of the set of all possible syndromes is equally likely.

A. Algorithm 1 with Symmetric Invalidation

Since the derivation of $T_{1S}(f, D_{1t}(n))$ is long, it is omitted. A complete derivation is shown in [5]. The plot of $T_{1S}(f, D_{1t}(n))$ versus f is shown in Fig. 6 for various values of n and t.[1]

The graphs show that the average number of steps, for a fixed de-

[1] $T_{1S}(f, D_{1t}(n))$ for $n = t$ is not covered by the expression derived in [5]. The values for this case were obtained by a computer enumeration. Note that when $n = f$ the syndrome consisting of 0's exclusively does not lead to any replaced faulty units. This syndrome, it is assumed, contributes one step to $T_{1S}(f, D_{1t}(n))$ and $T_{2S}(f, D_{1t}(n))$.

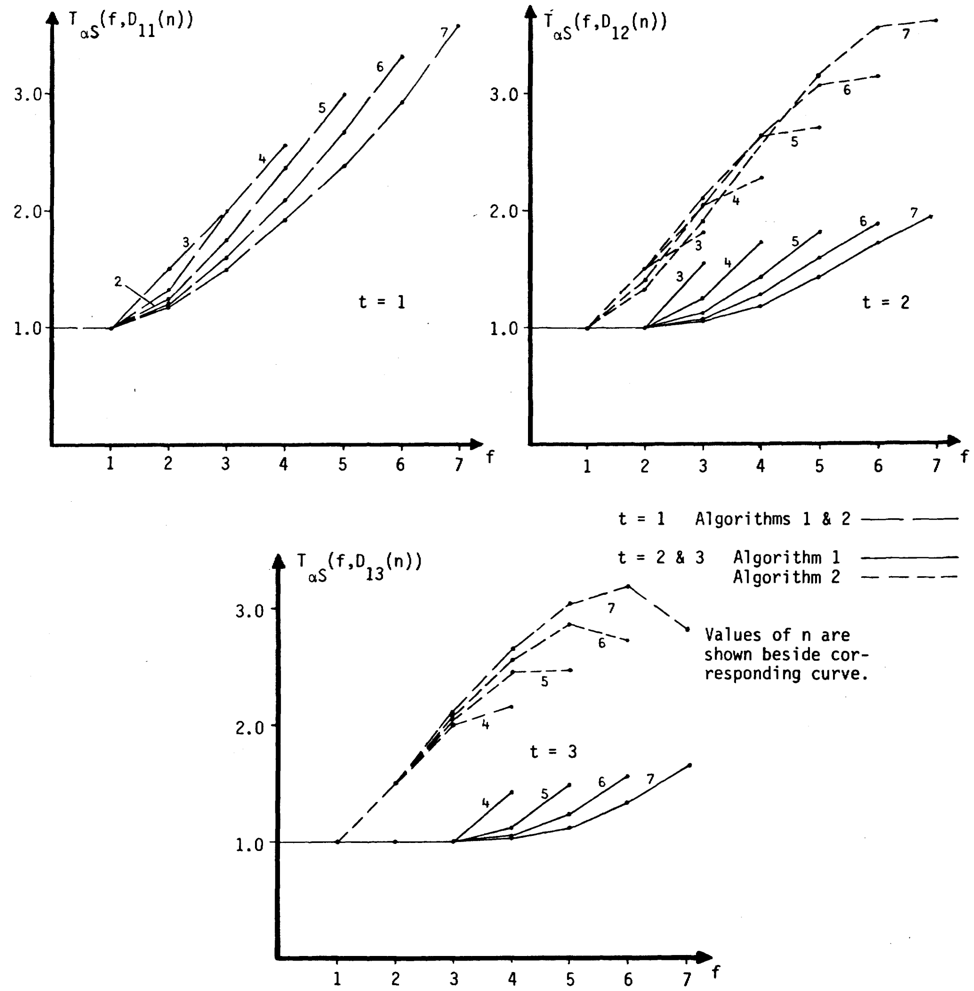

Fig. 6. $T_{\alpha S}(f, D_{1t}(n))$, the average number of steps required to diagnose design $D_{1t}(n)$ versus f, the number of faulty units, for Algorithms 1 and 2 and symmetric invalidation.

sign, increases approximately linearly with an increase in f, the number of faults. The rate of increase is low. For example, in design $D_{12}(6)$, if the number of faults is increased four-fold form 1 to 4, the average number of steps required to diagnose the system increases from 1.0 to 1.275, about a 28 percent increase. The data also shows that there is a significant decrease in diagnosis time with an increase in t, the number of tests per unit, for small t. For example, in a system $D_{1t}(6)$, the average number of steps required to diagnose four faults decreases by 18 percent from 1.275 to 1.050 when t increases from 2 to 3.

B. Algorithm 2 with Symmetric Invalidation

The complexity in characterizing the number of steps required to diagnose a set of f faults seems to preclude a concise analysis. However, to serve as a basis of comparison, a computer program was written to enumerate the syndromes, the steps required for each syndrome, and the average of the number of steps required.

Fig. 6 shows that the average number of steps required when Algorithm 2 is applied increases with increasing faulty units f. Since Algorithm 2 is identical to Algorithm 1 when $t = 1$, the plot of the number of steps is the same for this case. When $t > 1$, Algorithm 2 exhibits a larger number of steps required, on the average, as well as a higher rate of increase than Algorithm 1. It is interesting to note that when the number of faults is large, the rate of increase of $T_{2S}(f, D_{1t}(n))$ with f is smaller than for small values of f. Further, when $t = 3$, $n = 6$, and $f = 5$, an increase in f to 6 results in a (unexplained) decrease in $T_{2S}(f, D_{13}(6))$! A similar situation occurs when $t = 3$, $n = 7$, and $f = 6$.

Unlike Algorithm 1, where an increase in t results in a decrease in the average number of steps in a diagnosis, Algorithm 2 displays varying dependencies on t. For example, in $D_{1t}(6)$, if $f = 2$ or 3, there is a monotonic increase in $T_{2S}(f, D_{1t}(6))$ with increasing t. However, if $f = 4$ or 5, the average number of steps required increases when t is changed from 1 to 2 and decreases when t is changed from 2 to 3. If $f = 6$ on the other hand, there is a monotonic decrease in $T_{2S}(f, D_{1t}(6))$ with t.

C. Algorithms 1 and 2 with Asymmetric Invalidation

Since a faulty unit will fail all of its tests when asymmetric invalidation is assumed, both Algorithms 1 and 2 always result in one-step diagnosis for this case. Because the number of steps is 1 regardless of the number of faults f their plots are omitted.

V. CONCLUDING REMARKS

Because: 1) Algorithms 1 and 2 both yield one-step diagnosis when asymmetric invalidation is assumed, and 2) Algorithm 2 results in significantly fewer replaced fault-free units, Algorithm 2 is the obvious choice for this case. On the other hand, when symmetric invalidation is assumed, the choice of algorithm depends on whether it is more important to maximize efficiency or to minimize time. The tradeoffs with respect to these criteria are as follows.

Efficiency:

1) Algorithm 2 replaces significantly fewer fault-free units than Algorithm 1 for $t > 1$.

2) when the number of faults f and units n are fixed, an increase in t, the number of tests per unit, results in an increase in the number of fault-free units replaced when Algorithm 1 is applied. However, the converse is true of Algorithm 2.

Time:

1) Algorithm 1 requires significantly fewer steps to diagnose a system than Algorithm 2 for $t > 1$,

2) Algorithm 1 demonstrates a significantly lower rate of increase in the number of steps required versus f, than does Algorithm 2.

Acknowledgment

The author would like to thank the following people for conversations which inspired topics in this paper: M. Liron, Bell Laboratories, Naperville, IL, D. T. Lee and G. Gursel, Northwestern University, Evanston, IL, and E. Bender of the University of California—San Diego. E. Bender noticed an error in (15) of an earlier version of this paper and valuable comments were contributed by the referees.

References

[1] F. P. Preparata, G. Metze, and R. T. Chien, "On the connection assignment problem of diagnosable systems," *IEEE Trans. Comput.*, vol. C-16, pp. 848-854, Dec. 1967.

[2] F. Barsi, F. Grandoni, and P. Maestrini, "A theory of diagnosability of digital systems," *IEEE Trans. Comput.*, vol. C-25, pp. 585-593, June 1976.

[3] S. Karunanithi and A. D. Friedman, "Analysis of digital systems using a new measure of system diagnosis," *IEEE Trans. Comput.*, vol. C-28, pp. 121-133, Feb. 1979.

[4] J. E. Smith, "Universal system diagnosis algorithms," *IEEE Trans. Comput.*, vol. C-27, pp. 374-378, May 1979.

[5] J. T. Butler, "An analysis of universal diagnosis algorithms," Dep. Elec. Eng. Comput. Sci., Northwestern Univ., Evanston, IL, Tech. Rep., Sept. 1980.

[6] ——, "On the number of steps required to diagnose a sequence of faulty processors," preprint.

[7] S. N. Maheshwari and S. L. Hakimi, "On models for diagnosable systems and probabilistic fault diagnosis," *IEEE Trans. Comput.*, vol. C-25, pp. 228-236, Mar. 1976.

[8] H. Fujiwara and K. Kinoshita, "Some existence theorems for probabilisticly diagnosable systems," *IEEE Trans. Comput.*, vol. C-27, pp. 379-384, Apr. 1978.

[9] J. D. Russell and C. R. Kime, "System fault diagnosis: Closure and diagnosability with repair," *IEEE Trans. Comput.*, vol. C-24, pp. 1155-1161, Dec. 1975.

[10] E. A. Bender, Dep. Math., Univ. of California, San Diego, personal communications.

[11] M. Hall, Jr., *Combinatorial Theory*. New York: Wiley, 1967.

Stored State Asynchronous Sequential Circuits

ALAN B. HAYES

***Abstract*—A method is described for realizing asynchronous sequential circuits in a manner analogous to the stored state method for synchronous sequential circuits. The method simplifies the process of constructing asynchronous sequential circuits, allows utilization of existing MSI parts, and avoids the necessity for concern with races or hazards.**

***Index Terms*—Asynchronous sequential circuits, Muller circuits, self-synchronization, self-timed systems, speed independent circuits, stored state sequential circuits.**

Manuscript received May 12, 1980; revised September 22, 1980, January 26, 1981, and March 13, 1981.

The author is with the Department of Computer Science, University of Utah, Salt Lake City, UT 84112.

I. Introduction

Although the classical method [1]-[3] for realizing asynchronous sequential circuits is well developed, it requires the expenditure of a substantial amount of effort to find state assignments that produce hazard-free and race-free realizations. It also yields realizations that use large quantities of random gate logic, resulting in high part count with SSI circuits and little if any regularity. The circuit complexity of conventionally realized asynchronous state machines increases very rapidly as the number of states and inputs to the state machine goes up, making the direct realization of reliable machines with more then a relatively small number of states and inputs an intractable problem. While it is possible to decompose a single machine into a collection of smaller and more easily realizable machines [3], [4], this approach will typically increase the part count and may substantially slow the operation of the machine.

The introduction of field programmable ROM's simplified the process of realizing synchronous sequential circuits by allowing the development of stored state techniques [5], [6]. In this correspondence a method is described for applying stored state techniques and self-synchronization [7]-[10] to asynchronous sequential circuits in an environment where all elements are Muller circuits or self-timed systems [11], [12]. In a self-timed environment all system elements incorporate initiation and completion signals, and a signaling protocol is used that insures the validity of associated data buses over specified intervals which are bounded by transitions of the initiation and completion signals. This greatly simplifies the self-synchronization process in that the number of signals relevant to the process is small, and in that the machine operates in single-input-change mode with respect to these signals.

In what follows the terminology and conventions used for self-timed systems are compatible with those used in [12], to which the reader is referred for background and additional detail.

II. Signaling Protocol

Because an asynchronous system lacks a clock that can be used to delimit data transmissions, the signaling protocol used must somehow incorporate a means to indicate the limits of a transmission. To assure that the data transmitted from one element of the system to another are actually received, some form of closed-loop or "handshaking" protocol is required.

A pulse, regardless of its duration, can be too fast for some element in a system; therefore transitions must be used in asynchronous signaling conventions. The minimum number of transitions required is two: one to mark the initiation of the operation (generally called REQUEST), and another to mark the termination of the operation (generally called ACKNOWLEDGE). Protocols that use this minimum number of transitions per operation are called 2-cycle or nonreturn-to-zero (NRZ) signaling schemes.

An alternative to 2-cycle signaling is 4-cycle or return-to-zero (RZ) signaling, which uses two transitions each on the REQUEST and ACKNOWLEDGE lines per operation. 4-cycle signaling was first invented by Muller [11] and is used in many of his examples of speed independent circuits. Both 2-cycle and 4-cycle signaling can be realized in single-rail (1 wire per bit) form, which is sensitive to the relative delays of the data paths and the associated REQUEST or ACKNOWLEDGE path; or in a delay-insensitive, double-rail (2 wires per bit) form where the REQUEST or ACKNOWLEDGE signal is embedded in each bit of the associated data bus [12].

The single rail form of 4-cycle signaling shown in Fig. 1 is used in the circuits described here. This figure and the following description of the signaling protocol are intended to show how data may be transferred between requester and acknowledger. It is not intended to imply that data may *only* be transferred between the requester and acknowledger. While it may seem that the state associated with the negative going transition of the input request or acknowledge signal is redundant, it is our experience that this is seldom the case. Where

0018-9340/81/0800-0596$00.75 © 1981 IEEE