concerned with diagnosis and have neglected the detection problem. One
result of interest is that it is shown that "for complete diagnosis the
number of tests which must be applied to the circuit is proportional to
the number of gates and is independent of the number of primary input
terminals." Assumed but not given in their method is an algorithm for the
purpose of computing a test to distinguish between two failures if such
exists.

A gate in a logic circuit is said to be sensitized for some primary input
pattern T if a change in the signal on the gate output causes a change on
the primary output. The difficulty here is that for a given failure on the
gate, say stuck-at-1, it may be sensitized by T but for another failure on the
same
gate, say stuck-at-0, it is not.

A graphical procedure is given, with the above assumption, for
computing a set of tests for diagnosing any single-gate failure. It does not re-
quire the construction of a fault table. The authors say that a program is
being written for this procedure and that they will further investigate the
single-gate failure assumption. In LSI such an assumption is probably not
valid, but just what would be a reasonable working assumption does not
seem to be clear at this time.

J. PAUL ROTH
IBM Corporation
Yorktown Heights, N. Y.

R70-13 Multigate Synthesis of General Boolean Functions by Threshold

This paper is based on the following idea. If the two residues $x \cap f(x)$
and $x \cap f(x)$ are realizable, respectively, with $p$ and $q$ threshold gates,
then $f$ is realizable with at most $p + q$ gates. And conversely, if the residues
require separately at least $r$ gates, then so does $f$. Thus, given a table of
minimal realizations for 4-argument functions (which require at most
three gates), realizations for 5-argument functions can be obtained which
are demonstrably minimal or close to it, by considering the five different
pairs of residues.

Unfortunately, such tables do not exist for larger $n$ (and would be
very large anyway), so application of the idea for 6-argument functions
(and beyond) is difficult. Furthermore, the two bounds on number of gates
needed—the worst residue realization $p$ or $q$ versus the best sum $p + q$—
begin to spread apart for higher $n$, so near-minimality quickly becomes
nonguaranteeable.

The paper presents an interesting idea well. The reader should be fore-
warned that positive interconnection weights are assumed.

R. O. WINDER
RCA Laboratories
Princeton, N. J.

R70-14 Some Results on Cascade Decomposition of Automata—B. G.

The automaton decomposition procedure described by Zeiger does
not necessarily yield a unique cascade decomposition of a given machine
$M$. This paper describes some characteristics of the machines in any such
decomposition of $M$ which may be predicted by examining states of $M$
that are permuted under some input sequence $x$. The principal results fall
into three main categories: cascade consequences of traces which


permu proper subsets of $Q_M$, the state set of $M$; consequences of symbols
which permute $Q_M$; and the decomposition of definite automata.

$S_{\infty}$ denotes the largest subset of $Q_M$ permuted by tape $x$. It is shown
that $S_{\infty}$ occurs as a cover element in any sequence of covers $C_1,
C_2, \ldots, C_n$ associated with a decomposition of $M$. Further, if $S_{\infty}$ is an
element of $C_i$ replaced in obtaining $C_{i+1}$, then each symbol in $x$ causes
a permutation of the states of the corresponding machine of the decomposi-
tion. Finally, the machines in any decomposition of $M$ receive no permuta-
tion inputs if and only if each $S_{\infty}$ is a singleton.

For an input symbol $\sigma$ which permutes $Q_M$, it can be shown that $\sigma$
permutes the elements of each cover $C_i$ in any such sequence of covers
$C_1, \ldots, C_n$. The author establishes the following: If $L_1, \ldots, L_n$ are
machines in the cascade decomposition of $M$ corresponding to $C_1, \ldots, C_n$
respectively, then $\sigma$ induces the identity permutation on $C_i$ if and only if
the cascade input $\sigma$ causes no state change in $L_1, \ldots, L_n(i \leq n)$.

Finally, definite automata are characterized as precisely those whose
cascade decompositions consist of reset machines (ignoring identity per-
mutations resulting from DON'T CARE state transitions).

In some of the proofs, particularly in the early portions of the paper, the
complicated notation common in such work tends to obscure essentially
simple ideas. The reader will often find a state diagram useful. The author's
principal contribution is toward working out some of the consequences
of present structure theory. It is to be hoped that this and similar investiga-
tions will help shape algebraic machine theory into a tool more useful to
the larger community of computer scientists.

W. F. CUTFIP
Central Washington State College
Ellensburg, Wash.

R70-15 Realization of Sequential Machines with Threshold Elements—

This paper deals with the realization of the combinatorial logic network
of sequential machines with threshold logic elements, with a view to one-
level realization. The state assignment problem is already a difficult one,
and here it is further complicated by the restriction that the logical gates
to be used are all threshold logic elements. This problem is quite important
and is worthy of intensive study because, although threshold logic element
as a general gate is well known, its value is not fully appreciated unless its
applicability in various situations is demonstrated.

This paper develops an ingenious method of realization, using binary
partitions of inputs, states, and outputs. This method is complete in the
sense that it starts with a theoretical study and ends with an algorithm
yielding the acceptable assignments, although a part of it, which was said
to be covered in the Ph.D. dissertation of one of the authors, is not included
in this paper. For an assignment to be acceptable, certain conditions must
hold for every state partition and every output partition. These conditions,
which form the backbone of this paper, are given in Theorem 4 and are
rigorously proved.

For convenience and ease of testing, the condition of linear separability
is replaced by a slightly weaker condition of 2-summability (complete
monotonicty), which is necessary but not sufficient. This seems to be
justified, because for switching functions of eight or fewer variables, 2-
summability is known to be a necessary and sufficient condition for a
function to be linearly separable.

There are some minor mistakes. It is stated that the addition referred
to in domain 2-summability is component-wise Boolean addition. How-
ever, to the knowledge of this reviewer, the addition in 2-summability
and 4-summability should be ordinary algebraic addition. For instance,
let $t_1 = 1100, t_2 = 1010, t_3 = 1010, t_4 = 0101$. If Boolean addition is
used, then $t_1 + t_2 = 1101 = t_3 + t_4$. Or the set-pair $[t_1, t_2, t_3, t_4]$ is "equal
sum." However, it is easily seen that $[t_1, t_2]$ is linearly separable from
$[t_3, t_4]$. 