THE AUTOMATIC DESIGN OF TESTABLE CIRCUITS*

M.A. Breuer

DIGITAL INTEGRATED SYSTEMS CENTER REPORT

DISC/82-6

DEPARTMENT OF ELECTRICAL ENGINEERING-SYSTEMS
UNIVERSITY OF SOUTHERN CALIFORNIA
LOS ANGELES, CALIFORNIA 90089-0781

MARCH 1982

---

*This work was supported in part by the National Science Foundation under Grant MCS-8023485.

ABSTRACT

In this report we extended the previous work on testability measures for digital circuits, as represented by the scoap system. In our work we employ a new way for computing the testability measures (costs), namely, we formulate the problem as an ILP. By so doing we can achieve results never before attained, namely automatic design for testability. That is, the system will automatically add test logic so that the desired testability constraints and goals can be achieved.

1. Introduction

Because of the growing complexity of digital circuits, design for testability is essential [GRA80, WIL73]. Rutman [RUT72] and Stephenson [STE74] introduced techniques for measuring the testability of digital circuits. Rutman's work was further extended by Breuer [BRE77, 78, 79] and the results of this work were further developed and implemented in the scoap program [GOL78, 79, 80]. This program, like other testability programs [BEN80], [DUS78], [GRA79], [KOV79], [LON79], calculates the relative observability and controllability of the various nodes in a circuit using a heuristic procedure. Based upon these values, the designer is supposed to redesign his circuit in order to get better testability values. In general this can be a very complex process.

In this paper we show how to formulate the calculation of the testability measures as an integer linear program (ILP). We also show how to include additional constraints and objective criteria so that the program can automatically resynthesize the circuit. Similar concepts have been discussed in [PAR81, PAR82].

2. Testability Measures

Though our technique can apply to several different testability measures, for simplicity we will employ the one introduced by Rutman.

Definitions:

$C\delta_i$ – the complexity (difficulty or cost) of controlling (bringing) line i to the value δ, $\delta \in \{0,1\}$ by placing values on the primary inputs.

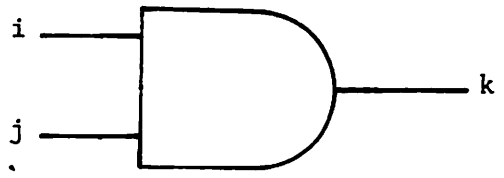
D_i – the complexity of observing (deducing) the value of line i by observing the values on the primary outputs.

The boundary conditions are:

$C\delta_i$ = 0 for all primary input lines i.

D_i = 0 for all primary output lines i.

The values of D_i and $C\delta_i$ for internal nodes are computed as follows. (We will illustrate the concept for only an AND gate.)

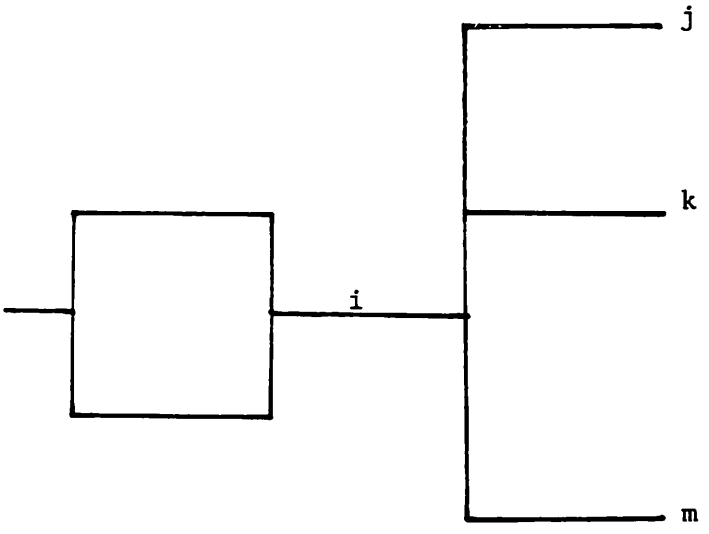


$$C1_k = 1 + C1_i + C1_j \quad (1)$$

$$CO_k = 1 + \min \{CO_i, CO_j\}$$

$$D_j = 1 + D_k + C1_i$$

For a fan-out line as shown below



$$D_i = \min(D_j, D_k, D_m)$$

For sequential circuits the values of $C\delta_i$ and D_i for a feedback line i are initially set to some large (maximum) value M.

The value of all testability factors $C\delta_i$ and D_i in a circuit can be computed in an iterative fashion as described by Rutman [RUT72].

3. ILP Formulation

We will now show how the computation of the testability values in a circuit can be formulated as an ILP. From (1) the computation requires the ability to add as well as find the minimum value of a set of quantities. We need only concern ourselves with the latter problem.

Let $\delta \in \{0,1\}$. Then min (x,y) can be expressed as an ILP as follows, where M is a large constant.

$$x - y \leq M\delta \tag{2a}$$

$$x \leq \alpha + M(1-\delta) \tag{2b}$$

$$y \leq \beta + M\delta \tag{2c}$$

$$Z(\min) = Z' + \alpha + \beta + \delta \tag{2d}$$

Assume $x > y$. Then $\delta = 1$ (2a) and from (2b) $x \leq \alpha$ and from (2c) $y \leq M$. By minimizing $\alpha + \beta$ we obtain $\alpha = x$ and $\beta = 0$. Similarly, if $x \leq y$, then $\delta = 0$ and $\alpha = 0$ and $\beta = y$. Thus

$$\alpha + \beta = \min(x,y).$$

Hence an equation of the form

$$C = 1 + \min(x,y)$$

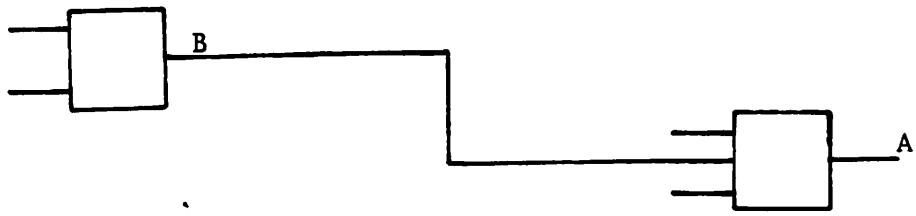
takes the ILP form

$$C = 1 + \alpha + \beta$$

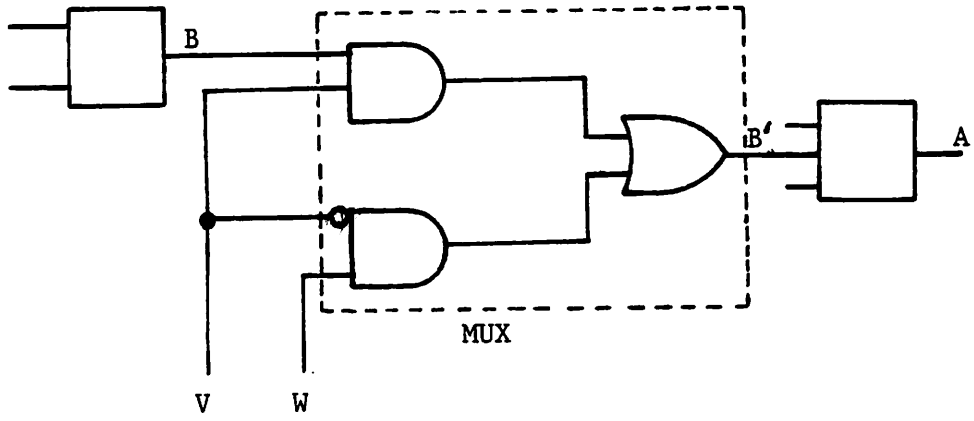
Controlability Circuitry

A node can be made more controllable by adding an AND gate (0-control) or an OR gate (1-control) or both.

Example:



Original Circuit



Modified Circuit

When V = 1 the circuit operates in its original way (B' = B), while if V = 0, B' = W (the injected line value).

The ILP can automatically add such test circuitry when required. The formulation is as follows.

Let $p \in \{0,1\}$ and $\Delta \in \{0,M\}$.

p = 0 implies test circuit not used

p = 1 implies test circuit used

Set $\Delta = (1-p)\ M$

Then p = 0 implies $\Delta = M$
p = 1 implies $\Delta = 0$

Set $CO_{B'} = \min\ (CO_B, \Delta)$

$C1_{B'} = \min\ (C1_B, \Delta)$

We modify the objective function to be of the form

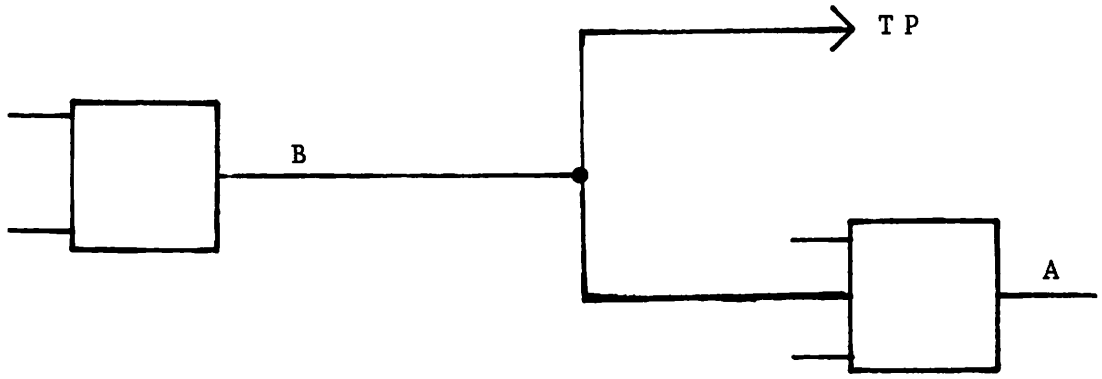
$Z(\min) = . \ . \ . \ + K_1 p$

where K_1 is the relative cost of a test circuit.

If there are numerous potential test circuits which can be added, then the number of circuits added can be constrained via the inequality

$$\sum_i p_i \leq K_2 \text{ (upper bound)}.$$

To aid in observability, a test point can be added, as shown below.



Now $D_{B'} = \min\ (D_B, \Delta)$

The testability values can be forced to be arbitrarily small, at the expense of adding controlability and observability logic, by adding constraints of the form

$$CO_i \leq K_3$$

$$C1_i \leq K_4$$

$$D_i \leq K_5$$

The objective function is of the form.

$$Z(min) = K_6 \sum_i (CO_i + C1_i + D_i) \quad \text{testability measures}$$

$$+ K_7 \sum_i (\alpha_i + \beta + \delta_i) \quad \text{min function}$$

$$+ K_1 \sum_i p_i \quad \text{testability logic}$$

The selection of the values of K_6, K_7 and K_1 control the solution obtained. For example, if K_1 is very large compared to K_6, no testability logic will be added since the reduction in the CO, C1 and D terms may not warrant the cost of the added circuitry.

Numerous other constraints can be added to this formulation to control various aspects of the design.

A system for implementing this technique is currently under development.

ACKNOWLEDGEMENT

The author would like to acknowledge several stimulating discussions with Professor A.C. Parker on the subject of testability and mathematical programming.

REFERENCES

[BEN80] R.G. Bennetts et al., "CAMELOT: A computer-aided measure for logic testability," PROC. ICCC80, New York, 1980, pp. 1162-1165.

[BRE77] "TEST/80 - An Advanced ATG System for Digital Circuits," Breuer and Assoc. Report 1-77, 1977,(available through the Defense Documentation Center).

[BRE78] M.A. Breuer, "New Concepts in automated testing of digital circuits," in Computer Aided Design of Digital Electronic Circuits, Brussels, G. Musgrave (editor), North Holland, 1978, pp. 57-80.

[BRE79] M.A. Breuer and A.D. Friedman, "TEST/80 - A proposal for an advanced automatic test generation system," Proc. Int'l. Automatic Test Conference, Minneapolis, Minnesota, September 19-21, 1979, pp. 305-312.

[CON80] W.M. Consolla and F.G. Danner, "An objective printed circuit board testability design guide and rating system," RADC technical report, RADC-TR-79-327, January 1980.

[DUS78] J. Dussault, "A testability measure," Proc. Semiconductor Test Conference, Cherry Hill, New Jersey, November 1978, pp. 113-116.

[GOL78] L.H. Goldstein, "Controllability observability analysis of digital circuits," Sandia Laboratories Report SAND 78-1895, 1978.

[GOL79] L.H. Goldstein, "Controllability observability analysis of digital circuits," IEEE Trans. on Circuits and Systems, vol. CAS-26, September 1979, pp. 685-693.

[GOL80] L.H. Goldstein and E.L. Thigpen, "SCOAP: Sandia controllability observability analysis program," Proc. 17th Design Automation Conference, June 1980, pp 190-196.

[GRA79] J. Grason, "TMEAS, a Testability measurement program," Proc. 16th Design Automation Conference, San Diego, June 25-27, 1979, pp. 156-161.

[GRA80] J. Grason and A.W. Nagle, "Digital test generation and design for testability," Proc. 17th Design Automation Conference, June 1980, pp. 175-189.

[KEI77] W. Keiner and R. West, "Testability measures," Proc. Int'l. Automatic Test Conference, 1977, pp. 49-55.

[KOV79] P.G. Kovijanic, "Testability analysis," Proc. Semiconductor Test Conference, Cherry Hill, New Jersey, 1979, pp. 310-316.

[LON79] B.A. Longendorfer, "A testability measure for hybrid circuits," Proc. Semiconductor Test Conference, Cherry Hill, New Jersey, 1979.

[PAR81] A.C. Parker, "Toward a Methodology of Solution of Testability or Register-Transfer Circuits," Technical Memo S-3, University of Southern California, March 1981.

[PAR83] A.C. Parker and L. Hafer, "Automating the design of testable hardware," in VLSI 81, John P. Gray editor, Academic Press, pp. 357-363, 1982.

[RUT72] R.A. Rutman, "Fault detection test generation for sequential logic by heuristic tree search," IEEE Computer Group Repository, R72-187, 1972.

[STE76] J.E. Stephenson, "A testability measure for register transfer level digital circuits," Ph.D. Dissertation, Carnegie-Mellon University, Department of Electrical Engineering 1974.

[STE76] J.E. Stephenson and J. Grason, "A testability measure for register level digital circuits," Proc. 6th Fault Tolerant Computing Conference, June 1976, pp. 101-107.

[WEB78] M.B. Weber, "Testability - the key to economical and operationally effective avionic test software," Proc. Int'l Automatic Test Generation Conference, 1978.

[WIL73] M.J.Y. Williams and J.B. Angell, "Enhancing testability for large scale integrated circuits via test points and additional logic," IEEE Trans. on Computers, vol. C-22, January 1973, pp. 46-60.