the Boolean identities. However, \( \overline{\overline{f}} \) and \( \overline{\overline{f}} \) do not have to be in the minimal form.

G5) For each \( x_i \) in \( F' \) generate all the \( x_i \) which are one variable less specified. If there are multiple copies of the same \( x_i \) only one need be retained in the total list.

G4) For all \( x_i \) in the list compute

\[
\hat{X}_i = \overline{\overline{f}}X_i
\]

and if \( \hat{X}_i \not= \phi \) leave the \( x_i \) in the \( x_i \) list.

G5) If \( \hat{X}_i = \phi \) in (16), replace \( x_i \) with all one variable less specified terms from \( x_i \) and add to the \( x_i \) list and go to G4.

G6) When the \( x_i \) list is complete, any (preferably minimal) set of minterms from \( \overline{\overline{f}} \) that cover all the \( x_i \) is the abnormal true tests.

G7) For each \( x_n \) in \( F' \) go through G3–G5 forming a partial \( x_n \) list and the \( \hat{X}_n \) by

\[
\hat{X}_n = \overline{\overline{f}}X_n
\]

G8) When the \( x_n \) list is complete, any (minimal) covering set of minterms from \( \overline{\overline{f}} \) is the abnormal false tests.

The commenters’ conclusion 3 applies to any near-minimal solution. As to the question arising from their conclusion 1), one does not have to perform the NR procedure on both \( F \) and \( \overline{\overline{f}} \) sides. The better side to work on would be the side that has fewer terms in the expression. Statistically speaking, one would expect a lesser number of test patterns from this side as well as smaller computation efforts.

REFERENCES


Comments on “Parallel Processing Algorithms for the Optimal Control of Nonlinear Dynamic Systems”

DANIEL TABAK

In a recent publication1 the authors have presented, in detail parallel processing algorithms, solving several classes of optimal control problems. Considerable attention is given to dynamic programming. On p. 777 the authors state: “The purpose of this paper is to describe a continuing effort in the development of algorithms for solving optimal control problems for nonlinear dynamic systems on parallel processors [4]-[11].” The author’s references [4]-[11] range through the period of 1971-1973. In fact, the idea of solving dynamic programming problems on parallel processors has been brought up and documented in a much earlier correspondence [1]. The solution of other types of optimization problems by parallel processors has been discussed by the author of this correspondence at the Asilomar Conference on Circuits and Systems in 1967, and documented in its Proceedings. At that time, the ideas were discussed in general form and no detailed formulations of the algorithms were given. It is certainly pleasant to see that the work in this area has been carried on to a much more advanced extent and some of the optimal control algorithms worked out in detail. The authors should be congratulated on their valuable contribution. It will be even more interesting to see the algorithms practically implemented on parallel processors.

REFERENCES


Manuscript received August 23, 1974.

The author is with the Department of Electrical Engineering, University of the Negev, Beer-Sheva, Israel, on leave from RPI, Hartford, Conn.

P. E. BLANKENSHIP

The two’s complement multiplication algorithm discussed by Baugh and Wooley1 is indeed a viable one, and we have used it several times in the past to great advantage in applications where modestly high performance signed multiplies were required, but the design was constrained to commercially available, standard components. This approach to sign compensation, which was first described to me in 1969 by C. M. Rader, does give rise to a uniform partial product array which can be efficiently summed by any of several techniques.

A method was devised here that used a Wallace tree summation approach and combined effective partial product formation with the first level additions by applying multiplier bit pairs as direct control inputs to programmable ALU-type devices. The programmable units are clearly necessary only on the first level of combination, subsequent levels requiring simple full adders. With standard 10 000 series ECL functions in a point-to-point wire wrap interconnect environment, 12 x 12 and 16 x 8 bit configurations were constructed, yielding measured worst case settling times of 45 ns.

As a point of further interest, it might be noted that the \( p_{m+1} \) and \( p_{m+1} \) columns of Fig. 3 of the above paper1 can be modified in several different ways to reduce the number of terms to be summed. The lower left-hand corner of the Baugh-Wooley array is

\[ x_{a-1} y_{m-1} \]

\[ x_{a-2} \]

\[ \cdots \]

\[ 1 \]

\[ \frac{y_{m-1}}{p_{m+1} p_{m+1} \cdots} \]

S. D. Pezaris suggests the following alteration, which has the advantage of employing the same side functions as the rest of the array:

\[ x_{a-1} y_{m-1} \]

\[ x_{a-2} \]

\[ \cdots \]

\[ 1 \]

\[ \frac{y_{m-1}}{p_{m+1} p_{m+1} \cdots} \]

An alternate approach which we found useful for our programmed ALU/Wallace tree structure requires only the inclusive-or of the sign bits in each column:

\[ x_{a-1} y_{m-1} x_{a-2} \]

\[ \cdots \]

\[ p_{m+1} p_{m+1} \]

All three approaches are equivalent, as can be easily verified by direct evaluation of the logical expressions for a binary full adder. Also, in many applications the \( p_{m+1} \) bit constitutes redundant information and its formation can be omitted entirely. Certainly \( p_{m+1} \) agrees with \( p_{m+1} \) in all but the special case wherein the product of the greatest negative operands is desirable.

Manuscript received February 12, 1974. This work was supported by the Department of the Air Force.

The author is with the M.I.T. Lincoln Laboratory, Lexington, Mass. 02173.


Comments on “A Two’s Complement Parallel Array Multiplication Algorithm”

DAVID KROFT

Abstract—This correspondence presents a simpler proof for Baugh and Wooley’s two’s complement parallel array multiplication algorithm, as demonstrated in a recent paper. The above algorithm

Manuscript received March 29, 1974; revised June 24, 1974. This work was supported by Control Data Canada Ltd., Mississauga, Ont., Canada.

The author is with the Canadian Development Division, Control Data Canada Ltd., Mississauga, Ont., Canada.