number of calibration samples from 10 to 91, in the case of a 9 entry LUT, the maximum value of the approximation error referred to the full scale change from 0.37 to 0.29%FS and the corresponding RMS value decreases from 0.14 to 0.10%FS. This performance is better than that obtained using a classical LUT technique. The optimal LUT method has been tested by approximating other functions and, in general, the RMS value of the approximation error decreases considerably, but a similar improvement in the corresponding maximum value is not guaranteed.

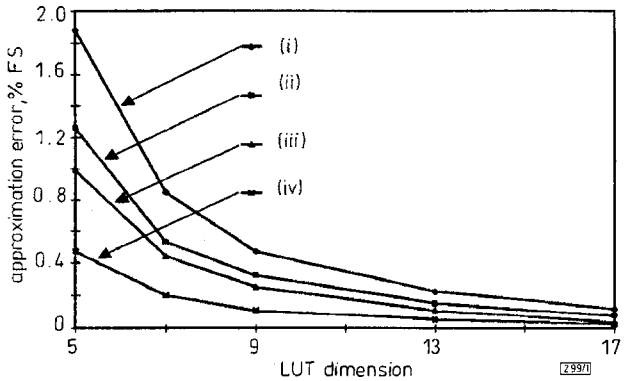

Fig. 1 *Function $y = sin(x)$ ($0° \leq x \leq 90°$): approximation error referred to full scale against LUT dimension*

(i) classical (max.)
(ii) optimal (max.)
(iii) classical (RMS)
(iv) optimal (RMS)

To evaluate the method, the output signals from a nonlinear resistive sensor have been used. An eight bit MC68HC11 microcontroller based system has been programmed to acquire and process the signals. The relationship between the measurand and the output signal has been characterised by means of > 50 points, that were acquired with an eight bit analogue to digital converter of the microcontroller. The end point linearity was 12%FS while the least squares linearity was 9%FS. Four methods were considered for LUTs with 5 and 9 entries:

(i) direct measurement of the LUT values, supported by linear extrapolation techniques if boundary points were not directly measurable
(ii) LUT entries calculated from the 5th degree Lagrange interpolating polynomial [3] obtained using equally spaced samples with respect to the measurand
(iii) LUT entries are calculated from a 5th degree interpolating polynomial, computed by a least mean squares technique using 22 sparse calibration samples
(iv) optimal LUT entries computed using 11, 22 and 42 sparse calibration samples.

Table 1: Measured data from resistive network: approximation error referred to full scale against LUT dimension

Method	Max. error	RMS error
	% FS	% FS
LUT (5 entries) classical (i)	1.915	1.1
LUT (5 entries) Lagrange (ii)	2.255	1.178
LUT (5 entries) fitting (22 samples) (iii)	2.23	1.178
LUT (5 entries) optimal (11 samples) (iv)	1.41	0.674
LUT (5 entries) optimal (22 samples) (iv)	1.325	0.633
LUT (9 entries) classical (i)	0.591	0.299
LUT (9 entries) Lagrange (ii)	0.671	0.296
LUT (9 entries) fitting (22 samples) (iii)	0.785	0.335
LUT (9 entries) optimal (22 samples) (iv)	0.415	0.198
LUT (9 entries) optical (42 samples) (iv)	0.385	0.192

LUT entries were computed using MATLAB. The different methods have resulted in a 3%FS of maximum deviation among corresponding LUT values, normalised to 20000 instead of 255 to minimise the effect of quantisation error. The results are shown in Table 1. Optimal LUTs decrease both the maximum and the RMS values of the approximation error referred to the full scale, and, as in all least mean squares methods, increasing calibration sample number improves the solution accuracy without increasing implementation difficulty. The relative simplicity of the method makes it possible to implement it with a microcontroller which, when reset, calculates the optimal LUT entries and then processes the sensor output signal using a classical and fast LUT with linear interpolation. Such a system allows a fast and inexpensive recalibration that would not be possible if interpolating polynomials were used because of the fixed point arithmetic of an 8 bit microcontroller.

When the sensor's output depends on two variables, an n-dimensional LUT has to be used. The method that has been described can easily be extended to the two variable case where the benefits, in terms of memory space reduction, are even greater. In such a case, the optimal LUT method is preferable with respect to neural techniques [7] as well, since it is a good compromise between performances and simplicity. Neural techniques, in fact, require a longer computation time and also require complex learning procedures that cannot easily be implemented on a microcontroller-based system.

Acknowledgments: The contribution of B. Pinelli to the laboratory work is greatly acknowledged.

13 May 1997
Electronics Letters Online No: 19970809

A. Flammini, D. Marioli and A. Taroni (*Department of Electronics for Automation, Faculty of Engineering, Brescia University, Via Branze 38, I-25123 Brescia, Italy*)

References

1 BRIGNELL, J.: 'Software techniques for sensor compensation', *Sens. Actuators A*, 1991, **25–27**, pp. 29–35
2 TRANKLER, H.R.: 'Signal processing' *in* GRANKE, T., and KO, W.H. (Eds.): 'Sensors: A Comprehensive Survey, Vol. 1, Fundamentals and General Aspects' (VCH, Weinheim, 1989), Chap. 10
3 PRENTER, P.M.: 'Splines and variational methods' (John Wiley and sons, New York, 1975), Chap. 2
4 MAHANA, P.N., and TROFIMENKOFF, F.N.: 'Transducer output signal processing using an eight-bit microcomputer', *IEEE Trans.*, 1986, **IM-35**, (2), pp. 182–186
5 MALCOVATI, P., AZEREDO LEME, C., O'LEARY, P., MALOBERTI, F., and BALTES, H.: 'Smart sensor interface with A/D conversion and programmable calibration', *IEEE J. Solid-State Circuits*, 1994, **29**, (8), pp. 963–966
6 VAN DER HORN, G., LYAHOU, K., and HUIJSING, J.: 'Calibration and linearisation method for microcontroller-based sensor systems', *Meas. Control*, 1996, **29**, pp. 270–273
7 JOHNSON, M.: 'Neural networks in advanced instrument design', *Meas. Control*, 1996, **29**, pp. 101–105

Best ROBDD variable ordering for functions with disjunctive decompositions

E.V. Dubrova, D.M. Miller and J.C. Muzio

Indexing terms: Reduced order systems, Boolean functions

The problem of finding a variable ordering to minimise the size of an ROBDD is considered for functions possessing disjunctive decompositions. An example is presented showing that the best ordering for a function with a disjunctive decomposition cannot always be directly determined from the best orderings for the component functions.

Introduction: Reduced ordered binary decision diagram (ROBDD) is a graphical data structure for the efficient representation of Boolean functions. Functions are represented by directed, acyclic graphs, which are built for some chosen ordering of the function variables [1]. Although a function requires, in the worst case, a

graph of exponential size in the number of variables, many practical functions have a linear representation for this.

For functions with disjunctive decompositions, storage can be saved [2] by expressing them as a composition of two functions g and h, namely

$$f(X) = g(h(Y), Z)$$

with Y and Z being sets of variables forming a partitioning of the set of variables $X = \{x_1, x_2, ..., x_n\}$ of f, and storing the ROBDDs of g and h. The ROBDD for f can be expanded in a straightforward fashion from the ROBDDs for g and h, by replacing the composition variable in the ROBDD of g with the graph for h, and then reducing the resulting diagram.

Normally, the size of the graph varies for different variable orderings and, for some functions, it is highly sensitive to the ordering. Finding a best ordering that minimises the size of the graph requires, in the worst case, a time exponential in the number of variables [3]. Therefore, computing the best orderings for two functions of variables n_1 and n_2 will usually be much faster than computing a best ordering for one function of $n_1 + n_2$ variables. Thus, a natural question to ask concerning functions with disjunctive decompositions is whether a best ordering for $f = g(h(Y), Z)$ can always be determined from the best orderings for h and g.

This Letter gives a negative answer to this question. We show that the best ordering for a function with a disjunctive decomposition cannot always be directly determined from the best orderings of the component functions.

Best ROBDD variable ordering for functions with disjunctive decompositions: Let $f(x_1, x_2, ..., x_n)$ be a completely specified Boolean function of type f: $B^n \rightarrow B$ on $B = \{0, 1\}$. We denote by X the set of the variables of f, i.e. $X = \{x_1, x_2, ..., x_n\}$

An ordering of the variables in an ROBDD for f is a vector describing the variables in order from top to bottom of the ROBDD. A best ordering is the ordering resulting in the ROBDD with a minimal number of nodes.

Let Y denote a proper subset of X, and let $Z = X - Y$. The operation functional substitution of a function h into a variable of another function g is defined if h: $B^{|Y|} \rightarrow B$ and g: $B \times B^{|Z|} \rightarrow B$, resulting in the function f: $B^{|Y|} \times B^{|Z|} \rightarrow B$ given by

$$f(X) = g(h(Y), Z)$$

Similarly, the operation ordering substitution of an ordering $<Y>$ into a variable h of another ordering $<Z_1, h, Z_2>$ is defined if $Z_1 \cup Z_2 = Z$, $h \notin Z$ and $Z_1 \cap Z_2 = \emptyset$, resulting in the ordering $<X>$ given by

$$\langle X \rangle = \langle Z_1, \langle Y \rangle, Z_2 \rangle$$

Note that without any confusion we are using h to also denote the substituted variable of g.

We are interested in whether the set of best orderings for a function with a disjunctive decomposition $f(X) = g(h(Y), Z)$ can always be composed from the best orderings of g and h, i.e. if it can be calculated by performing an ordering substitution on the sets of best orderings for h and g. In more formal terms, this questioii can be expressed as follows.

Let S_1 be the set of all non-degenerate (i.e. depending on all their input variables) functions of n variables or less. Let S_2 be the set of all sets which are best orderings of the functions from S_1. Let α: $S_1 \rightarrow S_2$, be defined as the mapping which assigns to any function $f \in S_1$ the set of best orderings for f from S_2. If $\circ$ denotes functional substitution, and $\bullet$ denotes ordering substitution, then we want to check whether the following equality holds:

$$\alpha(g \circ h) = \alpha(g) \bullet \alpha(h) \qquad (1)$$

for all g, $h \in S_1$ for which the operation $\circ$ is defined. Here, $\alpha(g \circ h)$ is the set of all best orderings for $f(X) = g(h(Y), Z)$, and $\alpha(g) \bullet \alpha(h)$ is the set obtained after performing ordering substitution on the sets of best orderings for h and g. Recall that if eqn. 1 holds, then α is a homomorphism between $(S_1, \circ)$ and $(S_2, \bullet)$ [4].

The following theorem shows, however, that this is not the case for $n \geq 5$.

Theorem: Let α: $S_1 \rightarrow S_2$ be the mapping which assigns to any function $f \in S_1$ the set of best orderings for f from S_2. Then, for $n \geq 5$, α is not a homomorphism between $(S_1, \circ)$ and $(S_2, \bullet)$.

Proof: By example; consider the following function of 5 variables:

$$f(x_1, ..., x_5) = x_1(x_4 \oplus x_5)' + x_2'(x_4 \oplus x_5) + x_1x_3 + x_1'x_2x_3'$$

This can be decomposed as $f = g(h(x_4, x_5), x_1, x_2, x_3)$, where

$$g = x_1h' + x_2'h + x_1x_3 + x_1'x_2x_3'$$

and

$$h = x_4 \oplus x_5$$

The ordering $<x_2, x_3, h, x_1>$ is the only best ordering for g, resulting in an ROBDD with 8 nodes. Therefore $\alpha(g) = \{<x_2, x_3, h, x_1>\}$. Since h is totally symmetric, all its orderings give ROBDDs with the same number of nodes, and therefore $\alpha(h) = \{<x_4, x_5>, <x_5, x_4>\}$. So $\alpha(g) \bullet \alpha(h) = \{<x_2, x_3, x_4, x_5, x_1>, <x_2, x_3, x_5, x_4, x_1>\}$. Both of these two orderings result in ROBDDs for f with 12 nodes. For example, the ROBDD for the ordering $<x_2, x_3, x_4, x_5, x_1>$ is shown in Fig. 1*a*.

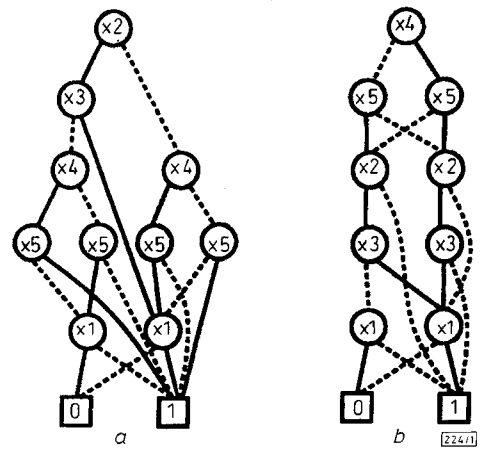

Fig. 1 *ROBDDs for f for different orderings*

a $<x_2, x_3, x_4, x_5, x_1>$
b $<x_4, x_5, x_2, x_3, x_1>$

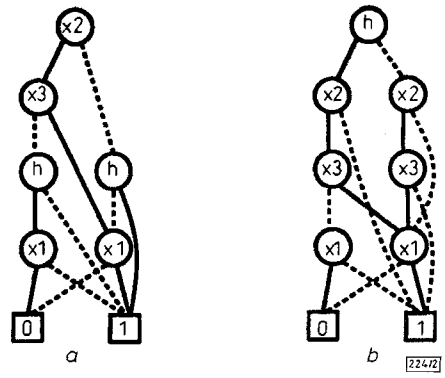

Fig. 2 *ROBDDs for g for different orderings*

a $<x_2, x_3, h, x_1>$
b $<h, x_2, x_3, x_1>$

However, there exist orderings for f which yield an ROBDD with 11 nodes, e.g. the ordering $<x_4, x_5, x_2, x_3, x_1>$ (Fig. 1*b*), and therefore $\alpha(g \circ h) \neq a(g) \bullet \alpha(h)$ for $n = 5$.

The phenomenon demonstrated by the above example holds for any h as long as h is a function of two or more variables, since in the ROBDD for g, for the ordering $<x_2, x_3, h, x_1>$, the variable h is represented by two nodes, while in the ROBDD for the ordering $<h, x_2, x_3, x_1>$ the variable h is represented by just one node (see Fig. 2*a* and *b*). Thus the theorem holds for $n \geq 5$.

The above function is also applicable for the case where negated edges [5] are allowed in the ROBDD.

Conclusion: This Letter gives a negative response to the question of whether a best ordering for a function of five or more variables with a disjunctive decomposition $f(X) = g(h(Y), Z)$ can always be determined from the best orderings of h and g. We believe that for $n < 5$ the answer is positive.

Our example shows that for $n \geq 5$, sometimes an ordering generated from the best orderings of g and h is not a best one for f. Furthermore, it demonstrates that it is possible that none of the

orderings generated this way are the best for f. Such examples, however, are quite rare, and are hard to find. We are presently working on determining a set of conditions for identifying such cases. It would also be interesting to see how the percentage of these cases changes as n increases.

7 May 1997
Electronics Letters Online No: 19970826

E.V. Dubrova, D.M. Miller and J.C. Muzio (*VLSI Design and Test Group, Department of Computer Science, University of Victoria, PO Box 3055, Victoria, V8W 3P6, Canada*)

E-mail: elena@csr.uvic.ca

References

1 BRYANT, R.E.: 'Graph-based algorithm for Boolean function manipulation', *IEEE Trans.*, 1986, **C-35**, (8), pp. 677–691
2 AKERS, S.B.: 'Binary decision diagrams', *IEEE Trans.*, 1978, **C-27**, (6), pp. 509–516
3 FRIEDMAN, S.J., and SUPOWIT, K.J.: 'Finding the optimal variable ordering for binary decision diagrams'. Proc. 24th ACM/IEEE Design Automation Conf., 1987, pp. 348–355
4 BIRKHOFF, G., and MacLANE, S.: 'Brief survey of modern algebra' (Macmillan, New York, 1977), 4th edn.
5 MINATO, S., ISHIURA, N., and YAJIMA, S.: 'Shared binary decision diagrams with attributed edges for efficient Boolean function manipulation'. Proc. 27th ACM/IEEE Design Automation Conf., 1990, pp. 52–57

Forward body-bias MOS (FBMOS) dual rail logic using an adiabatic charging technique with sub –0.6V operation

K. Kioi, H. Kotaki, S. Kakimoto, T. Fukushima and Y. Sato

Indexing terms: MOS logic circuits, Logic circuits

A novel logic family for low-voltage adiabatic logic, called forward body-bias MOS (FBMOS) dual rail logic, has been proposed. This technique uses forward body-bias effects to enable non-floating output levels during the entire data valid time without increased transistor count.

Introduction: Recently, several researchers independently proposed adiabatic circuits for low-energy logic [1 – 4]. The theoretical limit on energy dissipation per logical step is $kT\ln 2$ for logically irreversible operations [5], where k is Boltzmann's constant and T is the absolute temperature. Logically, reversible operations can conquer the limit but are currently not essential because the switching energy in CMOS logic circuits is much larger than the limit. Hence, the previous designs of adiabatic circuits using logically irreversible operations [1, 2] are practical from a size viewpoint. However, the previous adiabatic circuits have floating output nodes [1] which may be eliminated by the addition of extra transistors [2]. The proposed circuits adopt a novel circuit technique which avoids the floating output nodes without any additional transistor count. FBMOS have non-floating output levels over the entire data valid time. An additional advantage of the proposed circuits is that a voltage drop caused by the capacitive coupling effect is drastically reduced. The power supply voltage of the proposed circuits is < 0.6V [6, 7].

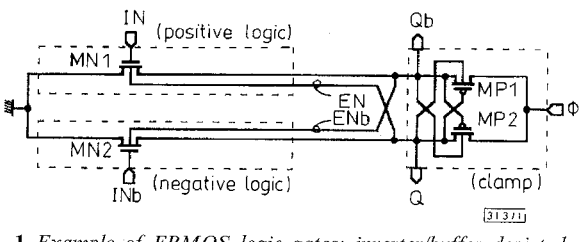

Fig. 1 *Example of FBMOS logic gates: inverter/buffer depicted with pMOS clamp*

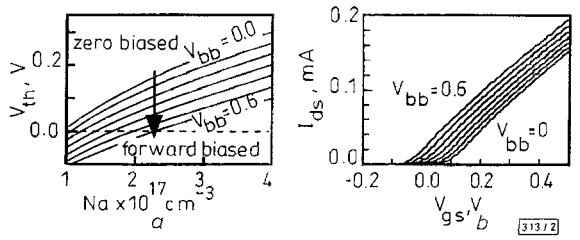

Fig. 2 *Characteristics of threshold voltage against net impurity concentration and simulation results of $I_{ds} - V_{gs}$ characteristics of nMOS transistor for various forward body-bias voltages*

a Threshold voltage against net impurity concentration
b Simulation results
$Na = 1.5 \times 10^{17}$

Circuit configuration of FBMOS logic: An example of the proposed FBMOS gates is shown in Fig. 1 This logic family has the same circuit configuration as ECRL [1] except for the cross-coupled forward body-bias nodes, EN and ENb of the nMOS transistors, and the cross-coupled body nodes of the pMOS transistors. In many other MOS circuits, the silicon substrate is connected to a fixed bias voltage. However, in the case of the twin double-well structure [6], the body terminal of each transistor is individually connected to another terminal as an SOI structure. The threshold voltage of a MOS transistor varies slightly against forward body-bias voltage. If the body-bias voltage, V_{bb}, equals zero, then the threshold voltage, V_{th}, is at its maximum value. As V_{bb} is increased, V_{th} decreases slightly. The net impurity concentration of nMOS transistors is adjusted so that when the body bias $V_{bb} = V_{dd}$, they are in a depletion mode of operation. When the body bias $V_{bb} = 0$, they are in an enhancement mode of operation. For net impurity concentrations within the feasible range, when V_{bb} is raised from 0 to 0.6V, the threshold voltage drops by ~0.1–0.2V, as shown in Fig. 2*a*.

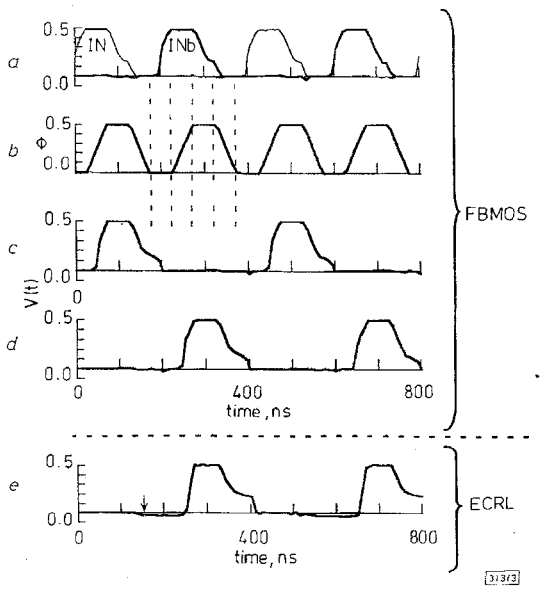

Fig. 3 *Simulation waveforms of FBMOS INV/BUF gate a–d and ECRL INV/BUF gate e*

nMOS $Na = 1.5 \times 10^{17}$cm^{-3}, pMOS $Na = 4.0 \times 10^{17}$cm^{-3}
a Inputs IN, INb
b Clock Power
c Output Q
d Output Qb
e Output Qb

Basic operation: The timing and logical operation of the FBMOS gate uses the same 4-phase clocking scheme as in ECRL and 2N-2N2P [2]. Consider the INV/BUF gate as an example of the basic operation of the FBMOS family. The waveforms are shown in Fig. 3 It is assumed that the valid level of input IN is low and that of INb is high. At the beginning of the WAIT (first) phase, the