UNIVERSITY OF CALIFORNIA
ENGINEERING · LOS ANGELES

CANONICAL AND MINIMAL FORMS FOR NAND AND NOR LOGIC

M. A. Marin
M. A. Melkanoff

DECEMBER 1965
REPORT NO. 65-57

Code 1

Code 1

Report No. 65-57
December 1965

CANONICAL AND MINIMAL FORMS FOR NAND AND NOR LOGIC

by

M. A. Marin and M. A. Melkanoff

Department of Engineering
University of California
Los Angeles

FOREWORD

The research described in this report, "Canonical and Minimal Forms fcr NAND and NOR Logic," No. 65-57, by M. A. Marin and M. A. Melkanoff, was supported by the Office of Naval Research, Information Systems Branch, Mathematical Sciences Division under contract number Nonr 233(52), and the Atomic Energy Commission, Research Division, Contract Number AT(11-1) Gen10, Project 14. The research was done in the Department of Engineering, University of California, Los Angeles, as part of a continuing program in Digital Technology Research.

ABSTRACT

In this paper we describe a method for deriving 2-level canonical expressions for NAND and for NOR logic using complemented and uncomplemented variables. Both expressions may be obtained directly from the truth table of the given Boolean function. Furthermore we prove that the Quine-McCluskey minimization technique can be extended to minimize the proposed 2-level canonical expressions.

CANONICAL AND MINIMAL FORMS FOR
NAND AND NOR LOGIC*

by

M.A. Marin† and M.A. Melkanoff†

I. INTRODUCTION

The NAND, NOR expressions of Boolean functions and their minimization have become important in the synthesis of switching circuits because of their easy implementation with integrated circuits.

It is well known that if a Boolean function is given in terms of AND-OR logic it is possible to transform this function directly in terms of NOR logic by means of the Shaeffer function, or in terms of NAND logic by the Pierce function [1]. Recently a synthesis technique has been developed [2] which makes it possible to implement a combinational logical expression by means of an all NAND or all NOR structure having a maximum of three logical levels. In this paper we propose to show that if a Boolean function is given in the form of a truth table it is possible to write directly from this truth table a 2-level canonical expression in terms of NAND and a 2-level canonical expression in terms of NOR. Furthermore we prove that the Quine-McCluskey minimization technique can be extended to our canonical forms if we allow complemented and uncomplemented variables and therefore this yields a minimal 2-level NAND or NOR expression.

II. CANONICAL FORM FOR NAND & NOR LOGIC

Shaeffer has proven that five postulates are sufficient to define a Boolean algebra [3]. From these postulates we can conclude that the operation NAND and the complementation operation are sufficient to generate any

Boolean function of the n variables $a_1, a_2, \ldots, a_n$.

Definition 1. We define a NAND elementary term as follows:

NAND$[a_1, a_2, \ldots, a_n]$ is 0 if and only if all $a_i = 1$ $(i = 1, \ldots, n)$,
NAND$[a_1, a_2, \ldots, a_n]$ is 1 if any $a_i = 0$ $(i = 1, \ldots, n)$.

Similarly a NOR elementary term is defined such that

NOR$[a_1, a_2, \ldots, a_n]$ is 1 if and only if all $a_i = 0$ $(i = 1, \ldots, n)$,
NOR$[a_1, a_2, \ldots, a_n]$ is 0 if any $a_i = 1$ $(i = 1, \ldots, n)$.

If we represent by a bar the complementation operation, it follows that

$$\overline{\mathrm{NOR}}[a_1, a_2, \ldots, a_n] = \mathrm{NAND}[\bar{a}_1, \bar{a}_2, \ldots, \bar{a}_n] \tag{1}$$

for indeed the left-hand side of eq. (1) is equal to OR$[a_1, a_2, \ldots, a_n]$ while the right-hand side is equal to the complement of AND$[\bar{a}_1, \bar{a}_2, \ldots, \bar{a}_n]$ which is OR$[a_1, a_2, \ldots, a_n]$.

The properties of the NAND and NOR elementary terms as defined above suggest a procedure for writing the NAND expression or the NOR expression directly from the given truth table.

We will show the procedure by an example. Suppose that the function f of two variables is given by the following truth table:

a	b	f
0	0	0
0	1	1
1	0	1
1	1	0

We can decompose the set of values of the function f into a number of columns such that each column contains only one entry with a value 1.

TABLE I

Function	Decomposition	
f	e	d
0	0	0
1	1	0
1	0	1
0	0	0

According to Definition 1, column *e* of Table I corresponds to the elementary term $\text{NOR}[a,\bar{b}]$, and column *d* to the elementary term $\text{NOR}[\bar{a},b]$. If we consider these two elementary terms as Boolean variables, say $e \equiv \text{NOR}[a,\bar{b}]$, $d \equiv \text{NOR}[\bar{a},b]$ we can construct the function

$$\overline{\text{NOR}}[e,d] = \overline{\text{NOR}}[\text{NOR}[a,\bar{b}],\ \text{NOR}[\bar{a},b]]. \qquad (2)$$

It is clear that the function $\overline{\text{NOR}}[e,d]$ will take the value 0 only when both *e* and *d* are 0 and therefore it realizes the given function f. Moreover, using eq. (1) the function f can be written in terms of the NAND operator:

$$\begin{aligned}\overline{\text{NOR}}[\text{NOR}[a,\bar{b}],\ \text{NOR}[\bar{a},b]] &= \text{NAND}[\overline{\text{NOR}}[a,\bar{b}],\ \overline{\text{NOR}}[\bar{a},b]]\\ &= \text{NAND}[\text{NAND}[\bar{a},b],\ \text{NAND}[a,\bar{b}]].\end{aligned}$$

Hence, the NAND canonical form of a function given by its truth table is given directly by the following rules:

1) Take the set of values of the variables for which the function is 1.
2) In this set of values replace the 1's by the corresponding literals and the 0's by the corresponding complemented literals.
3) Join each group with a NAND operator, and join again those NAND elementary terms with a NAND operator.

Example

a	b	c	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Application of the rules stated above yields

$$\mathrm{NAND}[\mathrm{NAND}[\bar{a},\bar{b},\bar{c}],\ \mathrm{NAND}[\bar{a},\bar{b},c],\ \mathrm{NAND}[a,\bar{b},\bar{c}]].$$

Similarly it is possible to write the NOR expression for the function f according to the following rules:

1) Take the set of values of the variables for which the function is 0.
2) In this set of values replace the 0's by the corresponding literals and the 1's by the corresponding complemented literals.
3) Join each group with a NOR operator and join again those NOR elementary terms with a NOR operator.

Thus, in the previous example the NOR canonical form of f is

$$\mathrm{NOR}[\mathrm{NOR}[a,\bar{b},c],\ \mathrm{NOR}[a,\bar{b},\bar{c}],\ \mathrm{NOR}[\bar{a},b,\bar{c}],\ \mathrm{NOR}[\bar{a},\bar{b},c],\ \mathrm{NOR}[\bar{a},\bar{b},\bar{c}]].$$

As there is no difficulty in generalizing the NOR and NAND operators applied to n variables, the above algorithm may be used with any number of variables.

III. GENERALIZATION OF THE CANONICAL FORMS

The generalization of the proposed canonical forms is straightforward using the concept of mapping [4] of a set A^n into B, where B = {0,1} and A^n is the set of the 2^n possible distinct states described by the Boolean variables a_1, $a_2, \ldots,$ a_n. Thus, for example, $A^2 = \{00, 01, 10, 11\}$. We will show that the 2^{2^n} possible mappings of A^n into B admits a canonical expansion in terms of the NAND mapping or the NOR mapping as defined in Definition 1.

Let us first consider the mapping of A^1 into B

a_1	$f(a_1)$	$=0$	$=a_1$	$=\bar{a}_1$	$=1$
0	f(0)	0	0	1	1
1	f(1)	0	1	0	1

The 2^2 possible mappings can be expressed by means of Definition 1 in the form

$$f(a_1) = \mathrm{NAND}[\mathrm{NAND}[f(0),\ \bar{a}_1],\ \mathrm{NAND}[f(1),\ a_1]]. \tag{3}$$

Similarly all possible mappings $f(a_1,a_2)$ of A^2 into B can be represented as follows:

$$f(a_1,a_2) = \mathrm{NAND}[\mathrm{NAND}[f(0,a_2),\bar{a}_1],\ \mathrm{NAND}[f(1,a_2),a_1]] \tag{4}$$

$$f(0,\ a_2) = \mathrm{NAND}[\mathrm{NAND}[f(0,0),\bar{a}_2],\ \mathrm{NAND}[f(0,1),a_2]] \tag{5}$$

$$f(1,\ a_2) = \mathrm{NAND}[\mathrm{NAND}[f(1,0),\bar{a}_2],\ \mathrm{NAND}[f(1,1),a_2]] \tag{6}$$

where we have used successively eq. (3) on the variables a_1 and a_2. Substituting eqs. (5) and (6) into (4) we obtain the 4-level canonical expansion

$$\begin{aligned} f(a_1,a_2) = \mathrm{NAND}[&\mathrm{NAND}[\mathrm{NAND}[\mathrm{NAND}[f(0,0),\bar{a}_2],\ \mathrm{NAND}[f(0,1),a_2]],\ \bar{a}_1], \\ &\mathrm{NAND}[\mathrm{NAND}[\mathrm{NAND}[f(1,0),\bar{a}_2],\ \mathrm{NAND}[f(1,1),a_2]],\ a_1]]. \end{aligned} \tag{7}$$

which can be transformed into the following 2-level canonical expression (cf. Appendix):

$$\begin{aligned} f(a_1,a_2)\ \mathrm{NAND}[&\mathrm{NAND}[f(0,0),\bar{a}_1,\bar{a}_2],\ \mathrm{NAND}[f(0,1),\bar{a}_1,a_2], \\ &\mathrm{NAND}[f(1.0),a_1,\bar{a}_2],\ \mathrm{NAND}[f(1,1),a_1,a_2]]. \end{aligned} \tag{8}$$

We might presume that the 2-level canonical expansion of the mapping A^n into B may be written, in a similar fashion (7), as follows

$$\begin{aligned} f(a_1,a_2,\ldots,a_n) = \mathrm{NAND}[&\mathrm{NAND}[f(0,0,\ldots,0),\bar{a}_1,\bar{a}_2,\ldots,\bar{a}_n],\\ &\mathrm{NAND}[f(1,0,\ldots,0),a_1,\bar{a}_2,\ldots,\bar{a}_n],\\ &\vdots\\ &\mathrm{NAND}[f(1,1,\ldots,1),a_1,a_2,\ldots,a_n]. \end{aligned} \tag{9}$$

We shall now prove eq. (9).

We define the symbol

$$\mathop{\mathbf{N}}_{\ell=1}^{n}[f(\ell),a_\ell^{(i_\ell)}] \equiv \mathrm{NAND}[f(I),a_1^{(i_1)},a_2^{(i_2)},\ldots,a_n^{(i_n)}] \tag{10}$$

and the symbol $\mathop{\mathbf{N}}_{I \in A^n}$ as the NAND'ing of the 2^n possible elementary functions corresponding to each distinct state of A^n thus

$$\begin{aligned} \mathop{\mathbf{N}}_{I \in A^n}[F(I;\ a_1^{(i_1)},\ a_2^{(i_2)},\ldots,a_n^{(i_n)}] = \\ \mathrm{NAND}[&F(0,0,\ldots,0;\ a_1^{(0)},a_2^{(0)},\ldots,a_n^{(0)}),\\ &F(1,0,\ldots,0;\ a_1^{(1)},a_2^{(0)},\ldots,a_n^{(0)}),\\ &\vdots\\ &F(1,1,\ldots,1;\ a_1^{(1)},a_2^{(1)},\ldots,a_n^{(1)})]. \end{aligned}$$

In particular, therefore,

$$\begin{aligned} \mathop{\mathbf{N}}_{I \in A^n}\ \mathop{\mathbf{N}}_{\ell=1}^{n}[f(I),a_\ell^{(i_\ell)}] &= \mathop{\mathbf{N}}_{I \in A^n}[\mathrm{NAND}[f(I),a_1^{(i_1)},a_2^{(i_2)},\ldots,a_n^{(i_n)}]]\\ &= \mathrm{NAND}[\mathrm{NAND}[f(0,0,\ldots,0),a_1^{(0)},a_2^{(0)},\ldots,a_n^{(0)}],\\ &\qquad \mathrm{NAND}[f(1,0,\ldots,0),a_1^{(1)},a_2^{(0)},\ldots,a_n^{(0)}],\\ &\qquad \vdots\\ &\qquad \mathrm{NAND}[f(1,1,\ldots,1),a_1^{(1)},a_2^{(1)},\ldots,a_n^{(0)}]]. \end{aligned} \tag{11}$$

If we add the condition

$$\begin{aligned} a^{(i)} &= a \text{ if } i = 1 \\ a^{(i)} &= \bar{a} \text{ if } i = 0, \end{aligned} \qquad (12)$$

then eq. (11) is identical to eq. (9) and therefore, represents in condensed notation the 2-level NAND canonical form of the 2^n possible mappings of A^n into B.

We prove now expansion (9) or (11) by mathematical induction. Consider the mapping

$$f(a_1,a_2,\ldots,a_n,a_{n+1})$$

of A^{n+1} into B, then according to eq. (4),

$$\begin{aligned} f(a_1,a_2,\ldots,a_n,a_{n+1}) = \text{NAND}[&\text{NAND}[f(a_1,a_2,\ldots,a_n,0),\bar{a}_{n+1}], \\ &\text{NAND}[f(a_1,a_2,\ldots,a_n,1),a_{n+1}]]. \end{aligned}$$

which can be written by means of eq. (11) as

$$\begin{aligned} \text{NAND}[&\text{NAND}[\underset{I\in A^n}{\mathbf{N}}[\overset{n}{\underset{\ell=1}{\mathbf{N}}}[f(I,0),a_\ell^{(i_\ell)}]],\ \bar{a}_{n+1}], \\ &\text{NAND}[\underset{I\in A^n}{\mathbf{N}}[\overset{n}{\underset{\ell=1}{\mathbf{N}}}[f(I,1),a_\ell^{(i_\ell)}]],\ a_{n+1}]] \end{aligned} \qquad (13)$$

Equation (13) represents the 4-level canonical form. It is possible to reduce this expression into a 2-level form by the procedure outlined in the Appendix and therefore expression (13) can be written as

$$f(a_1,a_2,\ldots,a_n,a_{n+1}) = \underset{I\in A^{n+1}}{\mathbf{N}}[\overset{n+1}{\underset{\ell=1}{\mathbf{N}}}[f(I),\ a_\ell^{(i_\ell)}]]. \qquad (14)$$

Equation (14) represents the 2-level NAND canonical expansion of the 2^{n+1} possible mappings of A^{n+1} into B. Thus eq. (11) with condition (12) is valid for any value of n.

The NOR canonical form can be also generalized for any number of variables in a similar fashion.

It is obvious from Definition 1 that there exists a duality between the NAND and the NOR definitions; therefore replacing in eq. (11) the N operators by similar operators defined in terms of NOR

$$\underset{I\in A^n}{\text{И}}[F(I;a_1^{(i_1)},a_2^{(i_2)},\dots,a_n^{(i_n)})] =$$

$$\begin{aligned}\text{NOR}[&F(0,0,\dots,0;a_1^{(0)},a_2^{(0)},\dots,a_n^{(0)}),\\ &F(1,0,\dots,0;a_1^{(1)},a_2^{(0)},\dots,a_n^{(0)}),\\ &\vdots\\ &F(1,1,\dots,1;a_1^{(1)},a_2^{(1)},\dots,a_n^{(1)})],\end{aligned}$$

$$\overset{n}{\underset{\ell=1}{\text{И}}}[f(I),a_\ell^{(i_\ell)}] \equiv \text{NOR}[f(I),a_1^{(i_1)},a_2^{(i_2)},\dots,a_n^{(i_n)}]$$

and replacing condition (12) by its dual

$$\begin{aligned}a^{(i)} &= \bar{a} \quad \text{if} \quad i = 1\\ a^{(i)} &= a \quad \text{if} \quad i = 0\end{aligned} \tag{15}$$

the 2-level NOR canonical expansion of the 2^n possible mappings of A^n into B becomes

$$\begin{aligned}f(a_1,a_2,\dots,a_n) &= \underset{I\in A^n}{\text{И}}\ \overset{n}{\underset{\ell=1}{\text{И}}}[f(I),a_\ell^{(i_\ell)}] =\\ &= \text{NOR}[\text{NOR}[f(0,0,\dots,0),a_1^{(0)},a_2^{(0)},\dots,a_n^{(0)}],\\ &\qquad \text{NOR}[f(1,0,\dots,0),a_1^{(1)},a_2^{(0)},\dots,a_n^{(0)}],\\ &\qquad \vdots\\ &\qquad \text{NOR}[f(1,1,\dots,1),a_1^{(1)},a_2^{(1)},\dots,a_n^{(1)}]].\end{aligned} \tag{16}$$

IV. MINIMIZATION

The problem is now to minimize the two suggested canonical forms. In the following we will prove by extending the concepts of SUBSUME and CONSENSUS introduced by Quine [5] and McCluskey [6], that any minimizer of AND-OR Boolean functions can be used to minimize the NAND or NOR canonical forms.[1]

According to the previous section, any Boolean function may be written in its NAND canonical form as:

$$\begin{aligned}&\mathrm{NAND}[\mathrm{NAND}[a_1,a_2,\ldots,a_p],\ \mathrm{NAND}[a_q,a_{q+1},\ldots,a_{q+k}],\ldots,\\&\qquad\qquad\qquad\mathrm{NAND}[a_s,a_{s+1},\ldots,a_{s+1}]].\end{aligned} \tag{17}$$

<u>Definition 2</u>. Given two NAND elementary terms

$$\phi = \mathrm{NAND}[a_1,a_2,\ldots,a_p]$$
$$\psi = \mathrm{NAND}[a_1,a_2,\ldots,a_p,\ldots,a_n] \quad n \geq p,$$

we say that ψ subsumes ϕ if all the literals of ϕ are contained in ψ.

<u>Theorem 1</u>

If the NAND canonical form of a Boolean function f contains a NAND elementary term subsumed by another NAND elementary term, the subsuming NAND elementary term can be removed without altering the function f.

<u>Proof</u>

$$\begin{aligned}f = {}&\mathrm{NAND}[\mathrm{NAND}[a_1,a_2,\ldots,a_p],\ \mathrm{NAND}[a_1,\ldots,a_p,\ldots,a_n],\\&\quad\mathrm{NAND}[a_s,\ldots,a_{s+k}],\ldots,\ \mathrm{NAND}[a_r,\ldots,a_{r+1}]];\end{aligned} \tag{18}$$

let

$$\begin{aligned}b_1 &= \mathrm{NAND}[a_1,\ldots,a_p]\\ b_2 &= \mathrm{NAND}[a_1,\ldots,a_p,\ldots,a_n]\\ b_3 &= \mathrm{NAND}[a_s,\ldots,a_{s+k}]\\ &\ \vdots\\ b_t &= \mathrm{NAND}[a_r,\ldots,a_{r+1}]\end{aligned}$$

The function (18) can be written as

$$f = \mathrm{NAND}[b_1,b_2,b_3,\ldots,b_t]. \tag{19}$$

Depending upon the values of b_i $(i = 1,\ldots,t)$ two cases may occur:

1) <u>$f = 0$</u>

If $f = 0$, then according to Definition 1, all $b_i = 1$ $(i = 1,\dots,t)$. Thus

$$b_1 = \mathrm{NAND}[a_1,\dots,a_p] = 1$$

$$b_2 = \mathrm{NAND}[a_1,\dots,a_p,\dots,a_n] = 1$$

But according to Definition 1,

$$b_1 = 1 \text{ if any } a_i = 0 \ (i = 1,\dots,p)$$

$$b_2 = 1 \text{ if any } a_j = 0 \ (j = 1,\dots,p,\dots,n)$$

Since the set i is a subset of set j, the function f will still be equal to 0 if we suppress b_2 from expression (19).

2) <u>$f = 1$</u>

If $f = 1$, then according to Definition 1, one or more of the b_i's must be 0 $(i = 1,\dots,t)$.

A) Suppose that all the terms b_i are equal to 1 except b_2. But if $b_2 = 0$, then $b_1 = 0$.

<u>Proof</u>

$$b_2 = \mathrm{NAND}[a_1,\dots,a_p,\dots,a_n] = 0 \text{ if and only if all } a_j = 1, \quad (j = 1,\dots,p,\dots,n)$$

$$b_1 = \mathrm{NAND}[a_1,\dots,a_p] = 0 \text{ if and only if all } a_i = 1, \quad (i = 1,\dots,p).$$

Since the set i is a subset of j the condition $b_2 = 0$ is a sufficient condition for b_1 to be equal to zero. Hence, f will remain equal to 1 if we suppress b_2.

B) Suppose that all the terms $b_i = 1$ except b_1, then the suppression of b_2 will still leave $f = 1$, because the sufficient condition for f to be 1 is that at least one of the terms b_i is equal to zero (Definition 1).

Definition 3. If two NAND terms α and β

$$\alpha = \mathrm{NAND}[a_1,a_2,\ldots,a_p]$$

$$\beta = \mathrm{NAND}[\bar{a}_1,a_2,\ldots,a_p,\ldots,a_n],$$

where a_i is the truth variable and $\bar{a}_i$ is the complemented variable, contain one and only one variable, which is complemented in one term and not in the other, then, the consensus of these two NAND terms is the NAND term formed with the variables of both NAND terms α, β, excluding the variable that is opposed. Thus the consensus of α, β, which we will denote by $C(\alpha,\beta)$ is given by

$$C(\alpha,\beta) = \mathrm{NAND}[a_2,a_3,\ldots,a_p,\ldots,a_n].$$

Theorem 2

Given a function in its NAND canonical form, this function will not change by inserting the consensus of two of its elementary NAND terms inside the main parenthesis.

Proof

We have to prove that the following two forms of f have the same truth table

$$f_1 = \mathrm{NAND}[\mathrm{NAND}[a_1,a_2,\ldots,a_p],\ \mathrm{NAND}[\bar{a}_1,a_2,\ldots,a_p,\ldots,a_n],\ \mathrm{NAND}[a_s,\ldots,a_{s+k}],\ldots,\mathrm{NAND}[a_r,\ldots,a_{r+1}]], \tag{20}$$

$$f_2 = \mathrm{NAND}[\mathrm{NAND}[a_2,\ldots,a_p,\ldots,a_n],\ \mathrm{NAND}[a_1,a_2,\ldots,a_p],\ \mathrm{NAND}[\bar{a}_1,a_2,\ldots,a_p,\ldots,a_n],\ \mathrm{NAND}[a_s,\ldots,a_{a+k}],\ldots,\ \mathrm{NAND}[a_r,\ldots,a_{r+1}]]. \tag{21}$$

For simplicity we define

$$b_1 = \text{NAND}[a_1,\ldots,a_p]$$

$$b_2 = \text{NAND}[\bar{a}_1,\ldots,a_p,\ldots,a_n]$$

$$b_{12} = \text{NAND}[a_2,\ldots,a_p,\ldots,a_n]$$

$$\vdots$$

$$b_t = \text{NAND}[a_r,\ldots,a_{r+1}].$$

Equations (20) and (21) can now can be written as follows

$$f_1 = \text{NAND}[b_1,b_2,\ldots,b_t] \tag{20'}$$

$$f_2 = \text{NAND}[b_{12},b_1,b_2,\ldots,b_t]. \tag{21'}$$

1) If $f_1 = \text{NAND}[b_1,b_2,\ldots,b_t] = 1$ then

$f_2 = \text{NAND}[b_{12},b_1,b_2,\ldots,b_t] = 1.$

<u>Proof</u>

By Definition 1

$$f_1 = 1 \text{ if any } b_i = 0 \quad (i = 1,\ldots,t).$$

But by Definition 1

$$f_2 = 1 \text{ if any } b_j = 0 \text{ or } b_{12} = 0 \text{ for } (j = 1,\ldots,t).$$

As the set j is a subset of set i we conclude that if

$$f_1 = 1 \text{ then } f_2 = 1.$$

2) If $f_1 = \text{NAND}[b_1,b_2,\ldots,b_t] = 0$ then

$f_2 = \text{NAND}[b_{12},b_1,b_2,\ldots,b_t] = 0.$

<u>Proof</u>

By Definition 1

$$f_1 = 0 \text{ if and only if all } b_i = 1 \quad (i = 1,\ldots,t),$$

then in f_2 all $b_j = 1$ $(j = 1,\ldots,t)$ because set i is equal to set j, thus the value of f_2 will be decided by the value of b_{12}; but b_{12} cannot be equal to zero for

$b_{12} = \mathrm{NAND}[a_2,a_3,\ldots,a_p,\ldots,a_n] = 0$ if and only if all $a_i = 1$ for $i = 2,\ldots,p,\ldots,n$, and therefore all $a_i = 1$ for $i = 2,\ldots,p,\ldots,n$. Thus either b_1 or b_2 are 0 but not both, which is in contradiction with the hypothesis $f_1 = 0$. Hence, we conclude

$$f_1 \equiv f_2.$$

The properties of the SUBSUME and CONSENSUS operations as discussed above, suggest a method to reduce the set of NAND elementary terms. This consists in forming the <u>iterated consensus</u>[2] NAND terms, inserting them inside the main parenthesis of the given NAND expression and removing all subsuming NAND terms. The resulting expression contains less number of NAND terms than the original canonical form.

For example, the function described by Table I,

$$f = \mathrm{NAND}[\mathrm{NAND}[\bar{a},\bar{b},\bar{c}],\ \mathrm{NAND}[\bar{a},\bar{b},c],\ \mathrm{NAND}[a,\bar{b},\bar{c}]],$$

can be written as a simpler expression by iterative application of the consensus and subsume operations in the following way:

The consensus of $\mathrm{NAND}[\bar{a},\bar{b},\bar{c}]$, $\mathrm{NAND}[\bar{a},\bar{b},c]$ is $\mathrm{NAND}[\bar{a},\bar{b}]$. According to Theorem 2,

$$f = \mathrm{NAND}[\mathrm{NAND}[\bar{a},\bar{b}],\ \mathrm{NAND}[\bar{a},\bar{b},\bar{c}],\ \mathrm{NAND}[\bar{a},\bar{b},c],\ \mathrm{NAND}[a,\bar{b},\bar{c}]].$$

Since $\psi_1 \equiv \mathrm{NAND}[\bar{a},\bar{b},\bar{c}]$ and $\psi_2 \equiv \mathrm{NAND}[\bar{a},\bar{b},c]$ subsumes $\phi \equiv \mathrm{NAND}[\bar{a},\bar{b}]$, both ψ_1 and ψ_2 can be suppressed. Hence

$$f = \mathrm{NAND}[\mathrm{NAND}[\bar{a},\bar{b}],\ \mathrm{NAND}[a,\bar{b},\bar{c}]].$$

This completes the first iteration of the consensus.

The consensus of $\mathrm{NAND}[\bar{a},\bar{b}]$, $\mathrm{NAND}[a,\bar{b},\bar{c}]$ is $\mathrm{NAND}[\bar{b},\bar{c}]$. According to Theorem 2,

$$f = \mathrm{NAND}[\mathrm{NAND}[\bar{b},\bar{c}], \mathrm{NAND}[\bar{a},\bar{b}], \mathrm{NAND}[a,\bar{b},\bar{c}]].$$

Since $\psi \equiv \mathrm{NAND}[a,\bar{b},\bar{c}]$ subsumes $\phi \equiv \mathrm{NAND}[\bar{b},\bar{c}]$ the term ψ may be removed, and

$$f = \mathrm{NAND}[\mathrm{NAND}[\bar{b},\bar{c}], \mathrm{NAND}[\bar{a},\bar{b}]]. \quad (22)$$

Since no other consensus term can be formed, the simplified expression of f is given by eq. (22).

With the application of iterative consensus and subsume operations it is possible to determine a set of simplified NAND-terms which we will call here NAND prime implicants using the terminology introduced by Quine[5] and McCluskey [6].

The simplified NAND expression of the given function obtained by the above method does not necessarily yield the minimal[3] NAND form because, in some cases, it is still possible to suppress one or more NAND prime implicants without changing the truth table of the given function. This is a consequence of Definition 1. Effectively, the necessary condition for f to be 1 is that all $a_i = 1$, $i = 1,\dots,n$; while the condition for f to be zero is that at least one $a_i = 0$ which is only a sufficient condition. Therefore it is clear that a NAND term b_i of the set of NAND prime implicants may be removed without altering the function f.

In order to find the minimal NAND expression of f it is necessary to determine the minimal set of NAND prime implicants.

Definition 4. A NAND irredundant expression f* for a given function f is an expression such that

1) every NAND elementary term is a NAND prime implicant
2) no NAND elementary term can be eliminated from f* without changing the values of the function f.

Theorem 3

The minimal NAND expression f* of a given function f is a NAND irredundant expression.

Proof

Suppose that the minimal NAND expression is not irredundant, then by Definition 4 two cases occur: 1) not every NAND elementary term of f* is a NAND prime implicant. In this case there exists at least a NAND elementary term which subsumes another one, and using Theorem 1 a shorter expression can be obtained. 2) a NAND elementary term can be eliminated from f* without changing the value of the function f. In this case f* is also not minimal.

Theorem 4

If f* is a NAND irredundant expression for f then each term of the canonical NAND expression for f subsumes at least one NAND elementary term of f*.

Proof

If f* is a NAND irredundant expression, then by Definition 4 every NAND elementary term is a NAND prime implicant; and every NAND prime implicant has been derived from the canonical NAND expression either 1) by application of Theorem 1, and this means that the obtained NAND prime implicant subsumes a NAND term of the canonical form, or 2) it is identical to a certain term of the canonical NAND expression. Since a NAND term subsumes itself, the theorem holds for this second possibility.

Theorem 4 suggested a method similar to that used by Quine [5]-McCluskey [6], to reduce the set of NAND prime implicants. Effectively, the minimal NAND expression will be formed with those NAND prime implicants which satisfy Theorem 4.

It is possible to prove in the same way as we have done for the NAND canonical expression that the same Theorems (1,2,3,4) hold for NOR canonical expressions and therefore the same minimization technique can be applied.

Let us consider for example the following function

a	b	c	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Considering the 0's of the function and applying Rule 2, we find that the NOR canonical form is

$$\mathrm{NOR}[\mathrm{NOR}[a,b,c],\ \mathrm{NOR}[\bar{a},b,c],\ \mathrm{NOR}[\bar{a},b,\bar{c}],\ \mathrm{NOR}[\bar{a},\bar{b},c],\ \mathrm{NOR}[\bar{a},\bar{b},\bar{c}]].$$

Applying the consensus operation and eliminating the subsumed terms we obtain

$$\mathrm{NOR}[\mathrm{NOR}[b,c],\ \mathrm{NOR}[\bar{a},\bar{c}],\ \mathrm{NOR}[\bar{a},\bar{b}]].$$

The consensus $C(\mathrm{NOR}[b,c],\ \mathrm{NOR}[\bar{a},\bar{c}]) = \mathrm{NOR}[b,\bar{a}]$ therefore

$$f = \mathrm{NOR}[\mathrm{NOR}[b,c],\ \mathrm{NOR}[\bar{a},\bar{c}],\ \mathrm{NOR}[\bar{a},\bar{b}],\ \mathrm{NOR}[b,\bar{a}]],$$

but $C(\mathrm{NOR}[\bar{a},\bar{b}],\ \mathrm{NOR}[b,\bar{a}]) = \mathrm{NOR}[\bar{a}]$, therefore

$$f = \mathrm{NOR}[\mathrm{NOR}[b,c],\ \mathrm{NOR}[\bar{a},\bar{c}],\ \mathrm{NOR}[\bar{a},\bar{b}],\ \mathrm{NOR}[b,\bar{a}],\ \mathrm{NOR}[\bar{a}]].$$

Since $\psi_1 \equiv \mathrm{NOR}[\bar{a},\bar{c}]$, $\psi_2 \equiv \mathrm{NOR}[\bar{a},\bar{b}]$ and $\psi_3 \equiv \mathrm{NOR}[b,\bar{a}]$ subsume $\mathrm{NOR}[\bar{a}]$, ψ_1, ψ_2, and ψ_3 can be eliminated.

The final NOR function is[4]

$$f = \mathrm{NOR}[\mathrm{NOR}[b,c],\ \mathrm{NOR}[\bar{a}]] = \mathrm{NOR}[\mathrm{NOR}[b,c],\ a]$$

And the minimal form becomes

$$f = \mathrm{NOR}[\mathrm{NOR}[b,c],\ a].$$

V. CONCLUSION

From the above presentation we can conclude the following:

1) Given the truth table of a Boolean function we can directly write two canonical forms, one corresponding to NAND logic and the other to NOR logic in terms of the truth variables and their complements.
2) The suggested technique for writing the functions in terms of the operators NOR or NAND leads directly to the actual circuit which realizes the function.
3) The Quine-McCluskey minimization process can be applied to the canonical forms to obtain the minimal NOR and NAND implementation of the function.
4) The two minimal forms obtained by the method, i.e. the NOR and NAND minimal forms of the given function, can be compared and a decision can be made as to which implementation needs less elements.

The present algorithm is being extended to those cases where fan-in/fan-out limitations appear as constraints. Furthermore a computer program is being prepared which yields the minimal forms as defined in this paper for any number of input variables less than 26.

APPENDIX

In this Appendix we prove that eq. (7) and (8) are identical.

Using the following equations

$$\text{NAND}[a_1,a_2,\ldots,a_n] = \text{OR}[\bar{a}_1,\bar{a}_2,\ldots,\bar{a}_n] \qquad (1A)$$

$$\overline{\text{NAND}}[a_1,a_2,\ldots,a_n] = \text{AND}[a_1,a_2,\ldots,a_n] \qquad (2A)$$

successively in (7) we obtain

$$\begin{aligned} f(a_1,a_2) = \text{OR}[&\overline{\text{NAND}}[\text{NAND}[\text{NAND}[f(0,0),\bar{a}_2],\ \text{NAND}[f(0,1),a_2]],\ \bar{a}_1],\\ &\overline{\text{NAND}}[\text{NAND}[\text{NAND}[f(1,0),\bar{a}_2],\ \text{NAND}[f(1,1),a_2]],\ a_1]]. \end{aligned}$$

$$\begin{aligned} f(a_1,a_2) = \text{OR}[&\text{AND}[\text{NAND}[\text{NAND}[f(0,0),\bar{a}_2],\ \text{NAND}[f(0,1),a_2]],\ \bar{a}_1],\\ &\text{AND}[\text{NAND}[\text{NAND}[f(1,0),\bar{a}_2],\ \text{NAND}[f(1,1),a_2]],\ a_1]]. \end{aligned}$$

And applying again eqs. (1A) and (2A) we obtain

$$\begin{aligned} f(a_1,a_2) = \text{OR}[&\text{AND}[\text{OR}[\text{AND}[f(0,0),\ \bar{a}_2],\ \text{AND}[f(0,1),\ a_2]],\ \bar{a}_1],\\ &\text{AND}[\text{OR}[\text{AND}[f(1,0),\ \bar{a}_2],\ \text{AND}[f(1,1),\ a_2]],\ a_1]]. \end{aligned} \qquad (3A)$$

Using the distributive law for AND/OR operators inside the main parenthesis of eq. (3A) we obtain

$$\begin{aligned} f(a_1,a_2) = \text{OR}[&\text{OR}[\text{AND}[f(0,0),\bar{a}_2,\bar{a}_1],\ \text{AND}[f(0,1),a_2,\bar{a}_1]],\\ &\text{OR}[\text{AND}[f(1,0),\bar{a}_2,a_1],\ \text{AND}[f(1,1),a_2,a_1]]]. \end{aligned} \qquad (4A)$$

The associative law of the OR operator applied to eq. (4A) yields finally

$$\begin{aligned} f(a_1,a_2) = \text{OR}[&\text{AND}[f(0,0),\bar{a}_2,\bar{a}_1],\ \text{AND}[f(0,1),a_2,\bar{a}_1],\\ &\text{AND}[f(1,0),\bar{a}_2,a_1],\ \text{AND}[f(1,1),a_2,a_1]]. \end{aligned}$$

If we use now eqs. (1A) and (2A) substituting NAND for OR according to eq. (1A) and NAND for $\overline{\text{AND}}$ according to eq. (2A) we obtain:

$$\begin{aligned} f(a_1,a_2) = \text{NAND}[&\text{NAND}[f(0,0),\bar{a}_2,\bar{a}_1],\ \text{NAND}[f(0,1),a_2,\bar{a}_1],\\ &\text{NAND}[f(1,0),\bar{a}_2,a_1],\ \text{NAND}[f(1,1),a_2,a_1]]. \end{aligned}$$

Thus the transformation of eq. (7) into (8) is valid.

*Received ____________________. This work was partly sponsored by the Atomic Energy Commission and the Office of Naval Research. One of the authors (M.A. Marin) also gratefully acknowledges a fellowship from the Del Amo Foundation.

†Department of Engineering, University of California, Los Angeles, California.

FOOTNOTES

1. Reference [2] states that "A rigorous proof has been made for the equivalence of any two-level NAND structure to the corresponding AND-OR structure. Minimizing techniques used in series-parallel synthesis can therefore be used on the two-level NAND structure." However we have been unable to obtain the internal memorandum referenced in [2].
2. The iterative consensus is used here in the way described by Samson [7] and Quine [5].
3. We understand here by minimal NAND expression that expression which contains the least number of NAND elements.
4. We have used here the equality NAND[a] = NOR[a] = $\bar{a}$ which follows from Definition 1.

REFERENCES

[1] Bartee, T.C., I.L. Lebow and I.S. Reed: Theory and Design of Digital Machines, N.Y.:McGraw-Hill, 1962, pp 37-38.

[2] Ellis, D.T.: "A Synthesis of Combinational Logic with NAND or NOR Elements," IEEE Trans. on Electronic Computers, vol. EC-14, pp 701-705, October 1965.

[3] Shaeffer, M.M.: "A Set of Five Postulates for Boolean Algebras, with Application to Logic Constants," Trans. Amer. Math. Soc., vol. 14, 1913, pp 481-488.

[4] See ref. [1], pp 103-107.

[5] Quine, W.V.: "On Cores and Prime Implicants of Truth Functions," Amer. Math. Monthly, vol. 66, Nov. 1959, pp 755-760.

[6] McCluskey, E.J., Jr.: "Minimization of Boolean Functions, Bell System Tech. J., vol. 35, Nov. 1956, pp 1417-1444.

[7] Samson, E.W. and B.E. Mills: Circuit Minimization: Algebra and Algorithm for New Boolean Canonical Expressions, AFCRC-TR-54-21, Cambridge, Mass., 1954.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Department of Engineering University of California Los Angeles	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

CANONICAL AND MINIMAL FORMS FOR NAND AND NOR LOGIC

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(Last name, first name, initial)*

Marin, Miguel A. and Melkanoff, Michel A.

6. REPORT DATE December 1965	7a. TOTAL NO. OF PAGES 23	7b. NO. OF REFS 7

8a. CONTRACT OR GRANT NO. b. PROJECT NO. Nonr 233(52) and AEC AT(11-1) GEN10, Project 14 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) 65-57 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. AVAILABILITY/LIMITATION NOTICES

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Navy Office of Naval Research Atomic Energy Commission

13. ABSTRACT

In this paper we describe a method for deriving 2-level canonical expressions for NAND and for NOR logic using complemented and uncomplemented variables. Both expressions may be obtained directly from the truth table of the given Boolean function. Furthermore, we prove that the Quine-McCluskey minimization technique can be extended to minimize the proposed 2-level canonical expressions.

DD FORM 1 JAN 64 1473 0101-807-6800

Unclassified
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Canonical expressions NAND, NOR logic						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
______________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
______________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
______________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.