JKAU: Eng. Sci., **Vol 17** No. 1, pp: 25 - 32 (2006A.D. /1427 A.H.)

Efficient Design of a Compact Two-Level Multiple-Output Logic Network

Ali M. Rushdi and Omar M. Ba-Rukab*

Department of Electrical and Computer Engineering, King Abdulaziz University, and

**Dept. of Computer Technology, College of Telecom. & Electronics, Jeddah, Saudi Arabia*

Abstract. This note proposes a minor modification of a recently-developed method that achieves two-level multiple-output logic minimization via the constrained minimization of a single function. The modified method is simpler and more efficient than the original one, but unlike the original method, it does not guarantee exact minimality except for small-size circuits.

1. Introduction

A new method for obtaining a two-level collective minimal cover for a set of switching functions $S = \{ f_1 , f_2 , \ldots , f_n \}$ has been recently proposed by the authors[1]. This method relies on the introduction of an auxiliary function F whose subfunctions (restrictions) with respect to (n-1) additional auxiliary variables $y_1 , y_2 , \ldots , y_{(n-1)}$ are certain (possibly repeated) members of S. A particularly constrained minimal cover for F is shown[1] to contain only labeled versions of some paramount prime implicants (PPIs) of S and can be used to construct an exactly minimal multiple-output cover of S. Our method[1] adds an additional (n-1)-valued dimension for the multiple output, in contrast to earlier auxiliary-function methods[2,3] which add an n valued dimension. Hence, the spatial complexity of our method[1] is one half that of these methods. Likewise, this complexity is almost one half that of PPI-based methods[4].

This note proposes a minor modification of the aforementioned method, in which exact minimality is traded off with a much desirable reduction in complexity and dimensionality. In this modification, the number of additional variables is further reduced from (n-1) to $r = \lceil log_2\, n \rceil$. Moreover, no repeated copies of the f_i's appear as subfunctions of the auxiliary function F. Instead, whenever, $2^r > n$, a number of ($2^r - n$) subfunctions of F (w.r.t. the auxiliary variables) are taken as don't cares. Finally, no constrained minimization is needed, and only a usual single-function minimization is adopted.

In the following two sections, the original method and its proposed modification are briefly outlined for n =3. A small illustrative example (in which exact minimization is achieved) follows in section 4 and conclusions are given in section 5.

2. The Original Method

For n= 3, the auxiliary function F is defined by

$$F = \bar{y}_1 \bar{y}_2 f_1 \vee y_1 \bar{y}_2 f_2 \vee y_2 f_3 , \tag{1}$$

which is visualized on the variable-entered Kanrnaugh map (VEKM)[5,6] of map variables y_1 and y_2 shown in Fig.1. Figure 1 actually shows all the conjunctive eliminants of F w.r.t. y_1 and y_2[7], including F itself expressed in VEKM form. It has been shown[1] (as can be easily deduced from Fig. 1 via complete-sum-derivation techniques[8]) that the complete sum of F is:

$$\begin{aligned} CS(F) = {} & \bar{y}_1 \bar{y}_2 CS(f_1) \vee y_1 \bar{y}_2 CS(f_2) \vee \bar{y}_2 CS(f_1 f_2) \vee y_2 CS(f_3) \vee \bar{y}_1 CS(f_1 f_3) \\ & \vee y_1 CS(f_2 f_3) \vee CS(f_1 f_2 f_3) , \end{aligned} \tag{2}$$

which means that CS(F) contains all the prime implicants of f_1, f_2, f_3, $f_1 f_2$, $f_1 f_3$, $f_2 f_3$ and $f_1 f_2 f_3$ (which constitute all the PPI's of the multiple-output function), each labeled with an appropriate tag of auxiliary-variable products. *Exact* multiple-output minimization is achieved when the function F is minimized with identical subfunctions of F being handled similarly[1].

3. The Modified Method

The modified auxiliary function G is now introduced as the following minterm expression with respect to $r = \lceil log_2 n \rceil$ auxiliary variables $y_1, y_2, ..., y_r$

$$G = \bigvee_{i=1}^{n} (\bigwedge_{j=1}^{r} y_j^{a_{ij}}) f_i \vee \bigvee_{i=n+1}^{2^r} d(\bigwedge_{j=1}^{r} y_j^{a_{ij}}), \tag{3}$$

where $(a_{ir} .... a_{i2} \; a_{i1})$ is the r-bit binary representation of the number $(i-1)$, and

$$y_j^{a_{ij}} = \begin{cases} \bar{y}_j & if \quad a_{ij} = 0 \\ y_j & if \quad a_{ij} = 1. \end{cases} \tag{4}$$

For $n = 3$, G reduces to:

$$G = \bar{y}_1 \bar{y}_2 f_1 \vee y_1 \bar{y}_2 f_2 \vee \bar{y}_1 y_2 f_3 \vee d(y_1 y_2), \tag{5}$$

which is visualized together with its other conjunctive eliminants w.r.t. y_1 and y_2 in VEKM form in Fig. 2. Note that while the cell $y_1 y_2 = 11$ in the VEKM of F in Fig. 1 contains a repeated copy of f_3, the corresponding cell for G in Fig. 2 is filled with a don't care. The complete sum of G is now deduced from Figure 2 as

$$\begin{aligned} CS(G) = {} & \bar{y}_1 \bar{y}_2 CS(f_1) \vee (y_1 \bar{y}_2 \vee d(y_1)) CS(f_2) \vee (\bar{y}_1 y_2 \vee d(y_2)) CS(f_3) \vee d(y_1 y_2) \\ & \vee \bar{y}_2 CS(f_1 f_2) \vee \bar{y}_1 CS(f_1 f_3) \vee CS(d(f_1 f_2 f_3)). \end{aligned} \tag{6}$$

Note that all PPIs are present in (6) except those arising from the product $f_2 f_3$. Therefore, exact minimality is not guaranteed when G is minimized instead of F. With appropriate choices of the d's, CS(f_2) can be tagged by y_1 and CS(f_3) is to be tagged by y_2 .

4. Illustrative Example

We illustrate our new approach by using it to solve the problem[1] of designing a two-level multiple-output network for three 4-variable incompletely specified functions given in decimal notation as follows:

$$f_1(X_1, X_2, X_3, X_4) = \sum(0, 1, 2, 9, 11, 12) + d(4, 10), \tag{7}$$

$$f_2(X_1, X_2, X_3, X_4) = \sum(0, 5, 11, 13, 14) + d(2, 10, 15), \tag{8}$$

$$f_3(X_1, X_2, X_3, X_4) = \sum(1, 4, 9, 10, 11, 13, 14) + d(5, 15). \tag{9}$$

Now, we define the auxiliary function G using 2 additional variables y_1 and y_2 according to the scheme of Fig. 2, which is shown enlarged in Fig. 3 wherein cells with 0 entries are left blank. Now, we construct a minimal cover for G in two stages. The first stage is algorithmic and produces all essential PIs of G and is shown in Fig. 3 while the second stage is the standard trial-and-error K-map procedure to add non-essential PIs and is shown in Fig. 4. In going from Fig. 3 to Fig. 4, all 1 entries covered by loops in Fig. 3 are switched into d entries to facilitate the selection of the non-essential PI loops. In general, the modified method is not guaranteed to achieve exact minimality. However, for the case of n= 3, it is possible to compensate for the missing PPIs that belong to the product $f_2 f_3$. Any PI loop with tag $y_1 (y_2)$ is checked to see if it can be used with the alternative tag $y_2 (y_1)$. Therefore, after constructing the PI loops $P_2 = X_1 X_3 y_1$ and $P_3 = X_2 \overline{X}_3 X_4 y_1$ in Fig. 3 we add the dotted loops $P_2' = X_1 X_3 y_2$ and $P'_3 = X_2 \overline{X}_3 X_4 y_2$. The minimal expression of G is

$$\begin{aligned} G = {} & X_2 \overline{X}_3 \overline{X}_4 \overline{y}_1 \overline{y}_2 \vee X_1 X_3 y_1 \vee X_1 X_3 y_2 \vee X_2 \overline{X}_3 X_4 y_1 \vee X_2 \overline{X}_3 X_4 y_2 \vee \overline{X}_1 \overline{X}_2 \overline{X}_4 \overline{y}_2 \\ & \vee \overline{X}_2 \overline{X}_3 X_4 \overline{y}_1 \vee \overline{X}_1 X_2 \overline{X}_3 y_2 \vee X_1 \overline{X}_2 X_3 . \end{aligned} \tag{10}$$

The AND-OR multiple-output network based on (10) is shown in Fig. 5, and after deletion of redundant (dotted) connections, it is found to achieve a minimal number of 10 gates (as a primary objective) and a corresponding minimal number of 31 input connections (as a secondary objective), as can be verified by other techniques [1,9] or by the logic minimizer Espresso [10,11].

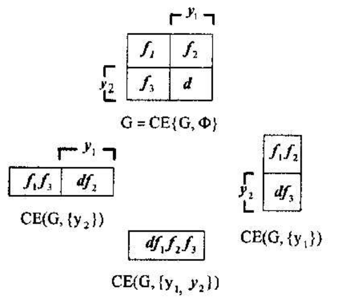

Fig. 1. VEKM representation of all the conjunctive eliminations CE(F,Y) of the auxiliary function F for n=3, with respect to all sets Y of auxiliary variables.

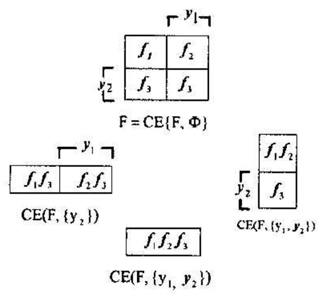

Fig. 2. VEKM representation of all the conjunctive eliminations CE(G,Y) of the auxiliary function G for n=3 with respect to all sets Y of auxiliary variables.

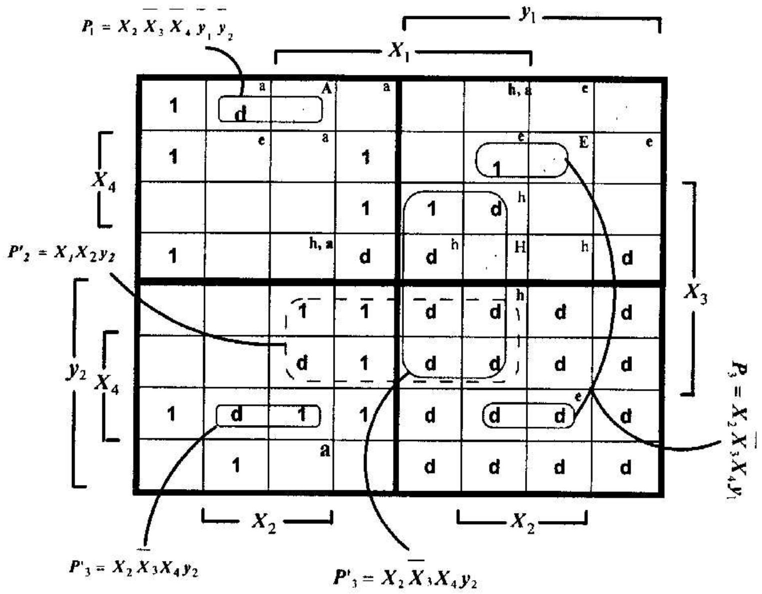

Fig. 3. Construction of all essential prime-implicant loops for the auxiliary function F1=$y_1y_2f_1$ v $y_1y_2f_2$ v $y_1y_2f_3$ v d(y_1y_2).

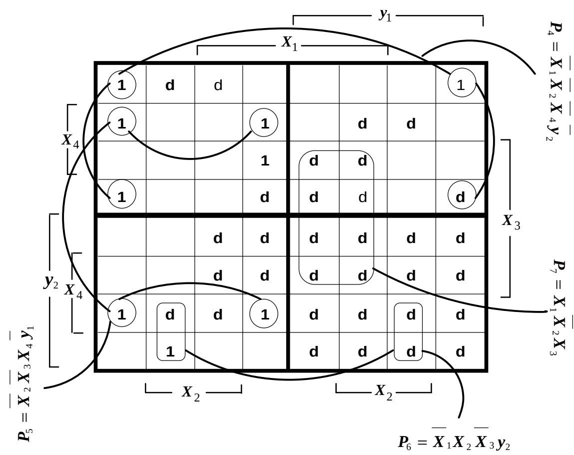

Fig. 4. Construction of non-essential prime-implicant loops that can be used to the essential ones of G to produce a minimal sum.

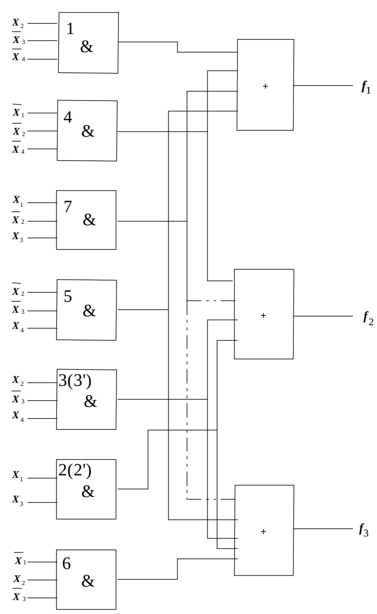

Fig. 5. The AND – OR network obtained in (8), with the unnecessary or redundant connections deleted (shown dotted).

5. Conclusion

This note has presented an efficient approach for the construction of a compact two-level multiple-output logic network. This approach transforms the original problem to a problem of minimizing a single output function, and can proceed by map, algebraic or tabular techniques. The cost of implementing the new approach is only minimally greater than that of individual minimization of the pertinent functions. However, it leads to better networks which are almost minimal.

References

[1] **Rushdi, A. M.** and **Ba-Rukab, O. M.**, Two-level multiple-output logic minimization using a single function, *International Journal of Computer Mathematics*, **80**(8) : 979-985 (2003).

[2] **Muller, D. E.,** Application of Boolean algebra to switching circuit design and to error detection, *IRE Transactions on Electronic Computers*, **EC-3**(1) : 6-12 (1954).

[3] **Hong, S. J., Cain, R. G.** and **Ostapko**, **D. L.**, MINI: A heuristic approach for logic minimization, *IBM Journal of Research and Development*, **18**(5) : 443-458 (1974).

[4] **Muroga, S**., *Logic Design and Switching Theory*, Wiley, New York, NY (1979).

[5] **Rushdi, A. M.,** Improved variable-entered Karnaugh map procedures, *Computers and Electrical Engineering*, **13**(1):41-52 (1987).

[6] **Rushdi, A. M.** and **Al-Yahya, H. A.,** Further improved variable-entered Karnaugh map procedures for obtaining the irredundant forms of an incompletely-specified switching function, *Journal of King Abdulaziz University (Engineering Sciences*), **13**(1):111-152 (2001).

[7] **Brown, F. M.**, *Boolean Reasoning: The Logic of Boolean Equations*, Second Edition, Dover Publications (2003).

[8] **Rushdi, A. M.** and **Al-Yahya, H. A.**, Derivation of the complete sum of a switching function with the aid of the variable-entered Karnaugh map, *Journal of King Saud University (Engineering Sciences)*, **13**: 239-269 (2001).

[9] **Rushdi, A. M.** and **Ba-Rukab O. M.**, A Map Procedure for Two-Level Multiple-Output Logic Minimization. *Proceedings of the Seventeenth National Computer Conference*, pp. 517-528, Al-Madinah Al-Munaw'warah (2004).

[10] **Brayton, R. K., Hachtel, G. D., McMullen, C.,** and **Sangiovanni-Vincentelli, A**., *Minimization Algorithms for VLSI Synthesis*, Kluwer Academic Publishers, Hingham, MA, (1984).

[11] **Hill, F. J.,** and **Peterson, G. R.**, *Computer Aided Logical Design with Emphasis on VLSI*, Fourth Edition, Wiley, New York, NY (1993).

التصميم السريع لدائرة منطق متعددة المخارج ثنائية المستوى شبه أصغرية

علي محمد على رشدي و عمر محمد عمر باركب[*]

قسم الهندسة الكهربائية وهندسة الحاسبات، *جامعة الملك عبدالعزيز*، [*]*قسم تقنية الحاسب الآلي، كلية الاتصالات والإلكترونيات*، جدة ، *المملكة العربية السعودية*

المستخلص. تقدم هذه المقالة المقتضبة تعديلاً طفيفاً لطريقة تم تطويرها حديثاً تحقق التصغير الأعظمي للدوائر المنطقية متعددة المخارج ثنائية المستوى عن طريق التصغير الأعظمي المقيد لدالـــة وحيـــدة. تتـــسم الطريقة المعدّلة بالبساطة والسرعة إذا ما قورنت بالطريقـــة الأصـــلية، ولكنها على خلاف الطريقة الأصلية لا تضمن تمام تحقيـــق التـــصغير الأعظمي إلا في حالة الدوائر صغيرة الحجم.