Improved Technology Mapping for PAL-based Devices Using a New Approach to Multi-Output Boolean Functions

Dariusz Kania;
Silesian University of Technology, Institute of Electronics, Poland
kdariusz@zeus.polsl.gliwice.pl

Abstract

The effective technology mapping for PAL-based devices is presented in this paper. The aim of this method is to cover a multiple-output function by a minimal number of PAL-based logic blocks. The product terms included in a logic block can be shared by several functions. Experimental results are compared to the classical technology mapping method.

1. Introduction

The PAL-based logic block constitutes the kernel of many CPLDs. The classical technology mapping of the function $f : B^n \rightarrow B^m$ within the PAL-based structures is related to implementation of the minimised functions $f_i : B^n \rightarrow B^1$ (i=1,2,…,m) by means of blocks consisting of k terms.

2. Method description

The minimised multi-output functions $f : B^n \rightarrow B^m$ can be described by a set of multi-output implicants, including an input part consisting of components {0,1,-} and an output part consisting of {0, 1} components. Let's assume, that $G < Y, \vec{U} >$ is the directed graph, where Y is the set of the output part of multi-output implicants, while $\vec{U}$ is a set of graph edges connecting the such nodes that the code distance of the output part is 1. On the basis of analysis of the graph nodes, solutions that use less logic blocks than classical approach can be found. Implementation of the group of implicants, which correspond to the selected node, during the i^{th} step, may lead to minimisation of the number of used PAL-based blocks consisting of k terms. The theoretical backgrounds of selection of the node in the i^{th} step of the algorithm are presented in [1]. Representation of the exemplary function by means of the graph and implementation based on the PAL-based logic blocks with three terms are shown in the Fig. 1. Table 1 shows how much PAL-based logic blocks with k-terms is needed for implementing the respective benchmarks by means of two implementation methods.

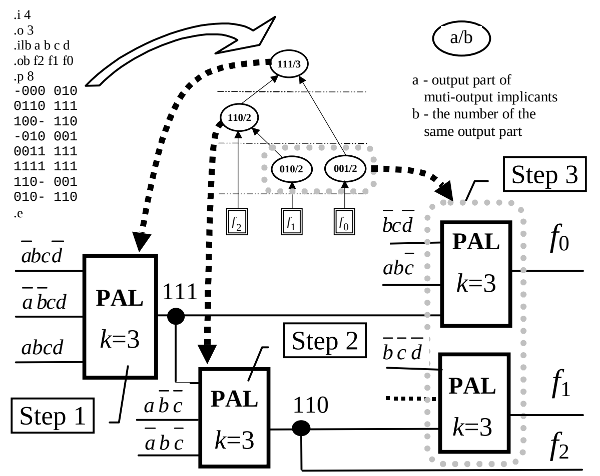

Fig. 1. Implementation of the multi-output function

Table 1. Results of experiments (*k*-number of terms)

	Proposed method				Classical method			
	k=3	*k=4*	*k=5*	*k=6*	*k=3*	*k=4*	*k=5*	*k=6*
5xp1	37	27	22	20	35	26	22	18
b12	26	19	15	15	27	19	15	14
Clip	66	45	37	29	73	49	38	30
Duke2	103	80	66	59	91	66	57	45
Ex1010	263	184	151	138	322	215	163	131
Misex2	19	19	18	18	19	19	18	18
Misex3	457	314	222	218	612	410	309	249
Rd53	15	11	8	7	15	11	8	6
Rd73	63	43	32	26	70	47	36	29
Rd84	128	87	66	54	160	107	80	58
Root	33	22	21	18	35	24	18	16
Squar5	17	12	11	9	14	11	9	9
Sqn	20	14	12	10	21	15	11	10
Sqr6	33	26	21	20	34	25	21	18
Sao2	29	20	16	13	36	24	19	15
Vg2	53	37	28	22	53	37	28	22
Table3	162	114	93	82	264	181	135	110
Σ	1561	1101	861	758	1918	1313	1009	798

3. Conclusion

The essence of the presented method is to search for implicants that can be shared by several functions. The proposed strategy of implementation of the multi-output function in many cases may lead to better solutions

References

[1] D. Kania, "A technology mapping algorithm for PAL-based devices using multi-output function graphs", *Proceedings of 26-th Euromicro Conference, IEEE Computer Society Press, Maastricht,* 2000, pp. 146-153