Comparative Analysis of Boolean Function's Minimization in Terms of Simplifying the Synthesis

A. Popov
popov_andrey@abv.bg

Kr. V. Filipova
kfilipova@abv.bg

St. Mihailov
stefan12d@yahoo.com

N. Kasev
nicola@idp.bg

TU – Sofia

2002

Abstract: **The last stage in the design of intelligent systems is the minimization of its functional description. This paper compares two of the well known methods for minimization of Boolean functions and proposes software solution, based on them. The comparison analysis continues with the synthesis stage, which is done in the MATLAB software environment and uses different building blocks**.

Key Words: **Boolean functions, Comparison analysis, Intelligent control, Minimization, Synthesis**

A mandatory stage in the design of all types of systems is the minimization in a given sense or in a given way. This depends on the type and the purpose of the system. The most popular and simple solution of the problem is the minimization of the functions for fielding the basic automated machines, used as building blocks in the complete automated machine. The role and the importance of this stage grow in more sophisticated and more detail described systems, like the intelligent ones.

The current paper contains a comparative analysis of the minimization of Boolean functions using available means such as Karnaugh maps and the Quine/McClasky algorithm. The synthesis of the automated machine is done in the MATLAB software environment, using two types of triggers from SIMULINK. The appropriate selection of two types of basic automated machines introduces the implication of an additional component in the quality of the synthesis.

Let the following automat is given with its table of inputs and outputs:

	a_0	a_1	a_2	a_3	a_4	a_5	a_6
x_0	a_1	a_2	a_2	a_1	a_5	a_2	a_1
	y_0	y_0	y_0	y_0	y_0	y_0	y_1
x_1	a_0	a_0	a_3	a_4	a_0	a_6	a_0
	y_0	y_0	y_0	y_0	y_0	y_0	y_1

The coding of the inputs (x), internal states (a) and outputs (y) are as follow:

	x
x_1	0
x_2	1

	Q_1	Q_2	Q_3
a_0	0	0	0
a_1	0	0	1
a_2	0	1	0
a_3	0	1	1
a_4	1	0	0
a_5	1	0	1
a_6	1	1	0

	y
y_0	0
y_1	1

In a coded view, with the chosen JK-triggers representing the seven inner states, the result table is:

N	x	Q_1	Q_2	Q_3	Q'_1	Q'_2	Q'_3	J_1	K_1	J_2	K_2	J_3	K_3	y
0	0	0	0	0	0	0	1	0	d_0	0	c_0	1	b_0	0
1	0	0	0	1	0	1	0	0	d_1	1	c_1	b_1	1	0
2	0	0	1	0	0	1	0	0	d_2	c_2	0	0	b_2	0
3	0	0	1	1	0	0	1	0	d_3	c_3	1	b_3	0	0
4	0	1	0	0	1	0	1	d_4	0	0	c_4	1	b_4	0
5	0	1	0	1	0	1	0	d_5	1	1	c_5	b_5	1	0
6	0	1	1	0	0	0	1	d_6	1	c_6	1	1	b_6	1
8	1	0	0	0	0	0	0	0	d_8	0	c_8	0	b_8	0
9	1	0	0	1	0	0	0	0	d_9	0	c_9	b_9	1	0
10	1	0	1	0	0	1	1	0	d_{10}	c_{10}	0	1	b_{10}	0
11	1	0	1	1	1	0	0	1	d_{11}	c_{11}	1	b_{11}	1	0
12	1	1	0	0	0	0	0	d_{12}	1	0	c_{12}	0	b_{12}	0
13	1	1	0	1	1	1	0	d_{13}	0	1	c_{13}	b_{13}	1	0
14	1	1	1	0	0	0	0	d_{14}	1	c_{14}	1	0	b_{14}	1

Note: combinations equal to 7 and 15 are not allowed and they are not in this table (but we assume that we could use them in the minimization as d_7 and d_{15})

For the purpose of minimizing the functions, the Karnaugh maps are used:

J_1	$\overline{Q}_2\overline{Q}_3$	$\overline{Q}_2 Q_3$	$Q_2 Q_3$	$Q_2\overline{Q}_3$
$\overline{x}\overline{Q}_1$	0	0	0	0
$\overline{x}Q_1$	d_4	d_5	d_7	d_6
xQ_1	d_{12}	d_{13}	d_{15}	d_{14}
$x\overline{Q}_1$	0	0	1	0

K_1	$\overline{Q}_2\overline{Q}_3$	$\overline{Q}_2 Q_3$	$Q_2 Q_3$	$Q_2\overline{Q}_3$
$\overline{x}\overline{Q}_1$	d_0	d_1	d_3	d_2
$\overline{x}Q_1$	0	1	d_7	1
xQ_1	1	0	d_{15}	1
$x\overline{Q}_1$	d_8	d_9	d_{10}	d_{11}

J_2	$\overline{Q}_2\overline{Q}_3$	$\overline{Q}_2 Q_3$	$Q_2 Q_3$	$Q_2\overline{Q}_3$
$\overline{x}\overline{Q}_1$	0	1	c_3	c_2
$\overline{x}Q_1$	0	1	c_7	c_6
xQ_1	0	1	c_{15}	c_{14}
$x\overline{Q}_1$	0	0	c_{10}	c_{11}

K_2	$\overline{Q}_2\overline{Q}_3$	$\overline{Q}_2 Q_3$	$Q_2 Q_3$	$Q_2\overline{Q}_3$
$\overline{x}\overline{Q}_1$	c_0	c_1	1	0
$\overline{x}Q_1$	c_4	c_5	c_7	1
xQ_1	c_{12}	c_{13}	c_{15}	1
$x\overline{Q}_1$	c_8	c_9	1	0

J_3	$\overline{Q}_2\overline{Q}_3$	$\overline{Q}_2 Q_3$	$Q_2 Q_3$	$Q_2\overline{Q}_3$
$\overline{x}\overline{Q}_1$	1	b_1	b_3	0
$\overline{x}Q_1$	1	b_5	b_7	1
xQ_1	0	b_{13}	b_{15}	0
$x\overline{Q}_1$	0	b_9	b_{11}	1

K_3	$\overline{Q}_2\overline{Q}_3$	$\overline{Q}_2 Q_3$	$Q_2 Q_3$	$Q_2\overline{Q}_3$
$\overline{x}\overline{Q}_1$	b_0	1	0	b_3
$\overline{x}Q_1$	b_4	1	b_7	b_6
xQ_1	b_{12}	1	b_{15}	b_{14}
$x\overline{Q}_1$	b_8	1	1	b_{10}

The connected elements are in gray color.

y	$\overline{Q}_2\overline{Q}_3$	$\overline{Q}_2 Q_3$	$Q_2 Q_3$	$Q_2\overline{Q}_3$
$\overline{Q}_1$	0	0	0	0
Q_1	0	0	g	1

The following minimized answers are formed after the grouping:

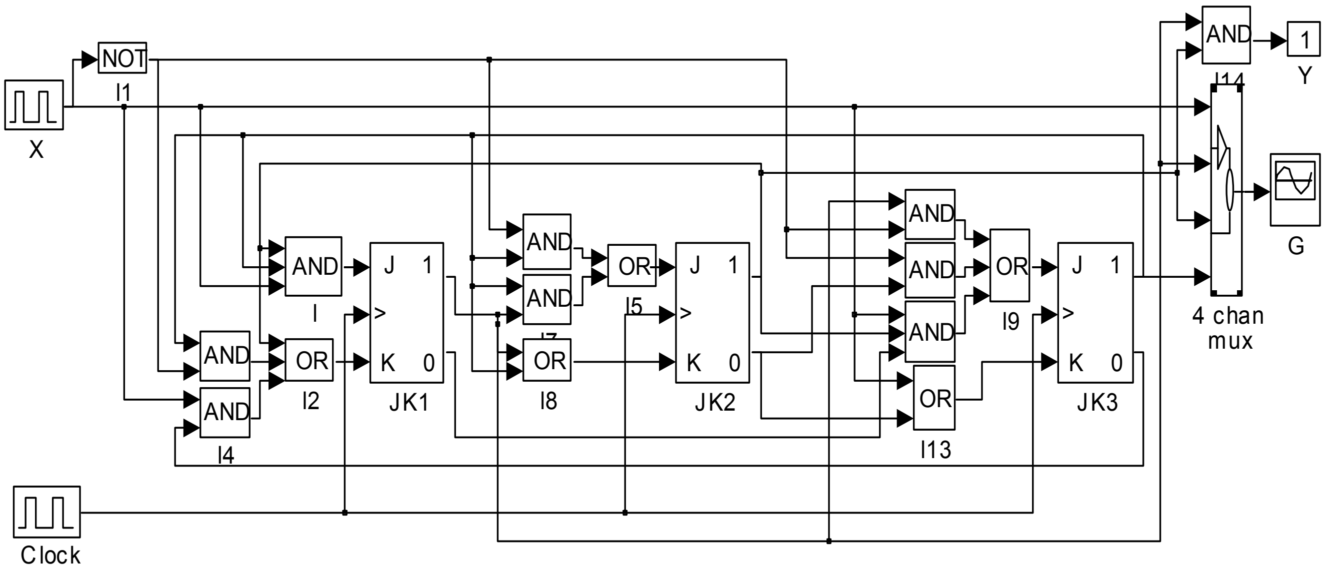
$$J_1 = xQ_2Q_3 \qquad K_1 = Q_2 + x\overline{Q}_3 + \overline{x}Q_3$$

$$J_2 = \overline{x}.Q_3 + Q_1Q_3 \qquad K_2 = Q_3 + Q_1$$

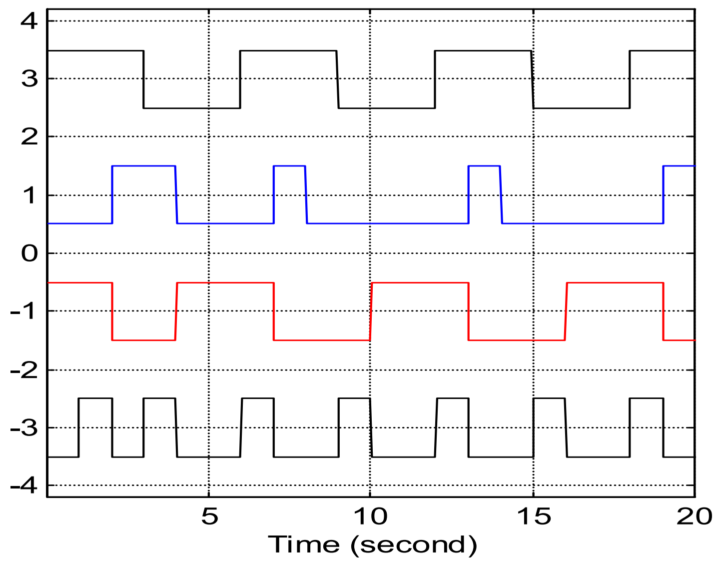
$$J_3 = \overline{x}\overline{Q}_2 + \overline{x}Q_1 + x\overline{Q}_1Q_2 \qquad K_3 = \overline{Q}_2 + x$$

$$y = Q_1Q_2$$

The synthesized automated machine is the presented in the MATLAB and SIMULINK environment:



Inputting a clock frequency and a defined input signal, the inner states of the automated machine are the following:



The states of the input variable and the outputs of the JK-triggers Q1, Q2 and Q3 are positioned vertically.

The minimization of J_1's activation function according to Quine/McClasky method is as follows:

Let's assume that $b_i=c_i=d_i=1$ for every i.

N	x	Q_1	Q_2	Q_3	J_1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

Step 1

		x	Q_1	Q_2	Q_3
I)	4	0	1	0	0
II)	5	0	1	0	1
	6	0	1	1	0
	12	1	1	0	0
III)	7	0	1	1	1
	11	1	0	1	1
	13	1	1	0	1
	14	1	1	1	0
IV)	15	1	1	1	1

Step 2

		x	Q_1	Q_2	Q_3
I)	4, 5	0	1	0	-
	4, 6	0	1	-	0
	4, 12	-	1	0	0
II)	5, 7	0	1	-	1
	5, 13	-	1	0	1
	6, 7	0	1	1	-
	6, 14	-	1	1	0
	12, 13	1	1	0	-
	12, 14	1	1	-	0
III)	7, 15	-	1	1	1
	11, 15	1	-	1	1
	13, 15	1	1	-	1
	14, 15	1	1	1	-

Step 3

		x	Q_1	Q_2	Q_3
I)	4, 5, 6, 7	0	1	-	-
	4, 5, 12, 13	-	1	0	-
	4, 6, 12, 14	-	1	-	0
II)	5, 7, 13, 15	-	1	-	1
	6, 7, 14, 15	-	1	1	-
	12, 13, 14, 15	1	1	-	-
III)	11, 15	1	-	1	1

Step 4

		x	Q_1	Q_2	Q_3
I)	4, 5, 6, 7, 12, 13, 14, 15	-	1	-	-
II)	11, 15	1	-	1	1

	+	+	+	+	+	+	+	+	+
	4	5	6	7	11	12	13	14	15
Q_1	x	x	x	x		x	x	x	x
xQ_2Q_3					x				x

$$J_1 = Q_1 \vee xQ_2Q_3$$

Note: sometimes in the final table there is a choice between few conjunctions, describing the same states (variables). In the table below the conjunctions marked in cyan describe 8 and 10 and both conjunctions in green describe 5.

	+	+	+	+				+	+	+	+
	0	1	2	3	5	8	10	12	13	14	15
$\bar{x}_1.\bar{x}_2$	x	x	x	x							
$\bar{x}_2.\bar{x}_4$	x		x			x	x				
$x_1.\bar{x}_4$						x	x	x		x	
$x_1.x_2$								x	x	x	x
$\bar{x}_1.\bar{x}_3.x_4$		x			x						
$x_2.\bar{x}_3.x_4$					x				x		

In red are marked the states, described only by one conjunction

Changing the values of the coefficients (b_i,c_i,d_i and g) and using the Karnaugh map, provides us the possibility of searching for a minimal form of the functions of the automated machine. The increase of the inner states and the triggers renders difficulties in specifying the values of coefficients b_i,c_i,d_i and g. This happens due to the increased size of the Karnaugh map. While the Karnaugh Maps do this visually, the Quine/McClasky method allows us to approach this task textually. In losing the visual intuition, we gain a more detailed analysis that applies to functions where the number of inputs makes visualizing too difficult. We also end up with something that can be relatively easily programmed, thus automating the minimization process.

In such cases, the proper software environment has to be used, in order to solve the synthesis problem of the automated machine (***macmin*** or ***minimize*** functions for MATLAB).

Finding a prime implicant using the Quine/McClasky method is easier than programming a function following the Karnaugh maps algorithm. Another problem exists here. Finding a prime implicant requires strictly defined input variables. This proves that, in order to synthesize an automated machine using the Quine/McClasky method, the coefficients (b_i,c_i,d_i and g) have to be promptly specified.

There is a possibility to make all possible combination of coefficients and to choose the best among them, but some other problems will appear:

- a proper method for comparison should be found
- the number of possible combination depends on the number of inputs and internal states

Example: For the given automat we have 2 (***i***) inputs and 7 (***s***) internal states. The first integer ***k*** (respectively ***p***) bigger or equal than ***i*** (respectively ***s)*** and equal to power of 2 is 2 (respectively) 8. The number (***n***) of coefficients in one JK trigger is: $n = k.p = 2.8 = 16$. This means that we have 2^8=256 possible combination for J input of the trigger and 256 combination for the K input.

Let the coefficients b_i, c_i, d_i be equal to 1 for every i. Using the MATLAB software environment and the ***macmin*** function, the following results are produced:

J1=macmin([4 5 6 7 11 12 13 14 15])

$$J_1 = xQ_2Q_3 + Q_1$$

J2=macmin([1 2 3 5 6 7 10 11 13 14 15])

$$J_2 = \overline{x}Q_3 + Q_1Q_3 + Q_2$$

J3=macmin([0 1 3 4 5 6 7 9 10 11 13 15])

$$J_3 = x\overline{Q_1}Q_2 + \overline{x}\overline{Q_2} + \overline{x}Q_1 + Q_3$$

K1=macmin([0 1 2 3 5 6 7 8 9 10 11 12 14 15])

$$K_1 = \overline{x}Q_3 + x\overline{Q_3} + \overline{Q_1} + Q_2$$

K2=macmin([0 1 3 4 5 6 7 8 9 11 12 13 14 15])

$$K_2 = Q_1 + \overline{Q_2} + Q_3$$

K3=macmin([0 1 2 4 5 6 7 8 9 10 11 12 13 14 15])

$$K_3 = x + Q_1 + \overline{Q_2} + \overline{Q_3}$$

Let the coefficients of b_i, c_i, d_i be equal to 0 for every i. Using the same function, the results are:

J1=macmin([11])

$$J_1 = x\overline{Q_1}Q_2Q_3$$

J2=macmin([1 5 13])

$$J_2 = \overline{x}\overline{Q_2}Q_3 + Q_1\overline{Q_2}Q_3$$

J3=macmin([0 4 6 10])

$$J_3 = x\overline{Q_1}Q_2\overline{Q_3} + \overline{x}\overline{Q_2}\overline{Q_3} + \overline{x}Q_1\overline{Q_3}$$

K1=macmin([5 6 12 14])

$$K_1 = \overline{x}Q_1\overline{Q_2}Q_3 + Q_1Q_2\overline{Q_3} + xQ_1\overline{Q_3}$$

K2=macmin([3 6 11 14])

$$K_2 = \overline{x}Q_1\overline{Q_2}Q_3 + xQ_1Q_2\overline{Q_3} + \overline{Q_1}Q_2Q_3$$

K3=macmin([2 5 9 11 13])

$$K_3 = \overline{x}\overline{Q_1}Q_2\overline{Q_3} + Q_1\overline{Q_2}Q_3 + X\overline{Q_1}Q_3$$

The comparison of the results of the three minimizations finishes the proof of the synonymity of the Karnaugh map method. The same result can be obtained using the Quine/McClasky method when specifying strict coefficients.

The problem of synthesizing such a searching procedure can be omitted choosing triggers that define the move from one state of the automated machine to other using exact values of the input signals.

If D-triggers are used, the truth table is the following:

N	x	Q_1	Q_2	Q_3	Q_1'	Q_2'	Q_3'	D_1	D_2	D_3	y
0	0	0	0	0	0	0	1	0	0	1	0
1	0	0	0	1	0	1	0	0	1	0	0
2	0	0	1	0	0	1	0	0	1	0	0
3	0	0	1	1	0	0	1	0	0	1	0
4	0	1	0	0	1	0	1	1	0	1	0
5	0	1	0	1	0	1	0	0	1	0	0
6	0	1	1	0	0	0	1	0	0	1	1
8	1	0	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	0	0	0
10	1	0	1	0	0	1	1	0	1	1	0
11	1	0	1	1	1	0	0	1	0	0	0
12	1	1	0	0	0	0	0	0	0	0	0
13	1	1	0	1	1	1	0	1	1	0	0
14	1	1	1	0	0	0	0	0	0	0	1

When minimizing the table with the MATLAB ***macmin*** function, the product functions are:

D1=macmin([4 11 13])

$$D_1 = \overline{x}Q_1\overline{Q_2}\,\overline{Q_3} + x\overline{Q_1}Q_2Q_3 + xQ_1\overline{Q_2}Q_3$$

D2=macmin([1 2 5 10 13])

$$D_2 = \overline{x}\,\overline{Q_2}Q_3 + \overline{Q_1}Q_2\overline{Q_3} + Q_1\overline{Q_2}Q_3$$

D3=macmin([0 3 4 6 10])

$$D_3 = \overline{x}\,\overline{Q_1}Q_2Q_3 + x\overline{Q_1}Q_2\overline{Q_3} + \overline{x}\,\overline{Q_2}\,\overline{Q_3} + \overline{x}Q_1\overline{Q_3}$$

We should have in mind that the results of the ***macmin*** function are slightly transformed for better visual comprehension.

The results when using of D-triggers are not better than the results from the JK-triggers, but in this case the number of the result functions is twice less. The use of D-triggers combined with a proper software environment, results in the possibility of a design of automated machines with infinite number of input variables. Almost the same result can be obtained when choosing T-triggers. The three functions will be quite complex, but in some cases this is the only way of synthesizing an automated machine with the MATLAB and SIMULINK software environments.

In order or make fast minimization with a low probability of making mistakes of an automat, a recommendation could be made D and T triggers to be used, or JK and RS triggers with predefined coefficients.

References

[1] Filipova Kr., Petrakieva S. - Synthesis of the logical control with Petri Net, Second intern. School on “Optimization of production”, Sozopol, 2001.

[2] Jahn D., Automatennetze als Beschreibungsmittel fur verteilte Stenernngsysteme, TU- Ilmenau, JWK, 1994.

[3] Filipova K. – “Автоматни модели”, TU-Sofia, 2000

[4] Ivanov S. and other. – “Анализ и синтез на логически схеми”, Varna, 1999