

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

Istanbul Aydin University, Turkey
Technical University of Iasi, Romania
nky@cs.tuiasi.ro

1. Foreword

As the title points out present paper consists of a new approach to designing finite automata. The idea is that the two well-known methodologies we can use (synchronous automata / asynchronous automata) have advantages but also each of them present unacceptable disadvantages.

Let us consider the synchronous approach. Evidently we can easily design such an automaton because we do not encounter large problems as for as timing is concerned. In this situation we can simply disconsider the presence of digital hazards in the circuit that implements the input functions group. In this situation all we have to ensure is a correct value with, t_{set-up} before and t_{hold} after the active edge of the Synchronizing signal. However this structure presents disadvantages that in some situations are unacceptable. First of all we notice that the synchronous structure has, a significant time delay between the moment of excitation (change of the input vector) and the response of the automaton (a change of the output vector in accordance with the flow graph that defines it). That happens because the input vector change will be considered by the automaton only at the next active edge of the synchronizing signal. In some situations, when the structure should respond a.s.a.p. to an input change, this delay in response is unacceptable. Further more let us assume that the input change is very small in time width. Also let us assume that the input change occurs between two successive edges of the synchronizing signal. In this situation there is a good change that the automaton will not sense the input change (it will ignore it) and will not evolve in accordance with the defining flow graph. Evidently this is not acceptable.

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

So how do not we solve the disadvantages presented above? Well we should consider the asynchronous approach for the automaton. In this situation we have a very short response time (the same magnitude rank as the propagation time through the structure). Also any change of the input vector, no matter how short but of course if the same time magnitude as the propagation time, will be considered. These are the advantages of the asynchronous structures. However designing an asynchronous automaton proves to be laborious. Why? Because we cannot allow the CLC implementing the input group of functions to have hazards present. That would provoke an erroneous evolution towards the next state for the automaton. Also we must certify the fact that all the state variables will switch at the same moment of time in order to avoid races that could prove to be critical. Further more by using an asynchronous automaton we may find that the outputs can present very short pulses (as a result of a very short input change). That is not acceptable if these outputs are to be used by a hierarchically superior digital structure that has well defined minimal timing characteristics for its inputs.

These are the problems that the present work tries to solve. The idea is that we shall define a compromise that will have a fast response time, will sense no matter how short input changes but will have the ease of design characteristic to synchronous structures.

First we define two methodologies for the analysis of hazards. One of them is derived from the classic time constants method, but the improvements we suggest enhance the method's capability of identifying hazards. The second method suggested (brand new one) considers an original approach. We define a set of new variables (logic in nature but dependant of time) that are used for mapping the outputs of a CLC. So after using any if these two methods we have of clear information on how the outputs of the CLC look like, so we can easily define the earliest moment of time (the shortest resolution time) when they hold a correct value. At this moment we can generate an active edge for the synchronizing signal and lock these values in a memory element (Flip Flop, Register, ... etc.). So now we can see how we're reached a compromise – we design a synchronous automaton but afterwards by analyzing the input CLC we are able to define a time variable synchronizing signal. That will improve the overall response time of the structure up to 75÷80% of the equivalent asynchronous structure. Finally we have to find a solution for the problem of providing minimal timing parameters for the output signals. Here we suggest a couple of methodologies for the control of these parameters.

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

Keywords: hazard in combinatorial structures, hazard analysis, Mealy automaton, variable duty factor clock.

2. Analysis for a sequential system – performance improvement

As we have stated at the beginning of the paper, the reason for which we trace the output values of a combinatorial logic circuit is to define not only the presence of hazard but also the time frames when the output of the circuit has a correct value. We are interested by this in the situation when we want to use the compromise we have described in the foreword – meaning that we would like to design a sequential system as a synchronous one but achieve the performance of an asynchronous one.

The idea is quite simple. For a given automaton (described for example by its states transition graph) we shall consider the synchronous design methodology. So finally we'll have a fully synchronous structure. The difference is that we shall not use a synchronizing clock with standard parameters. We shall use a synchronizing clock that has a variable period (a variable HIGH and LOW period of time). Why we can do this? – because if the automaton is in a known state (from the states transition graph) and considering that the input vector is known at that moment of time, we shall be able to find out the time frame where the evolution would be a correct one (by analyzing with one of the methods presented above the functioning of the combinatorial circuit that implements the input group of functions). After mapping in time the values for the outputs of these circuits we shall easily see which is the earliest moment of time, when an active edge of the clock would imply a correct evolution for the automaton. That is exactly what we shall do – we'll generate the next active edge for the clock at that moment of time (the earliest). In this manner we shall not reach exactly the performance of an asynchronous automaton but we shall drastically improve the overall performance of the synchronous automaton. The general structure suggested for this system is presented in fig. 1.

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

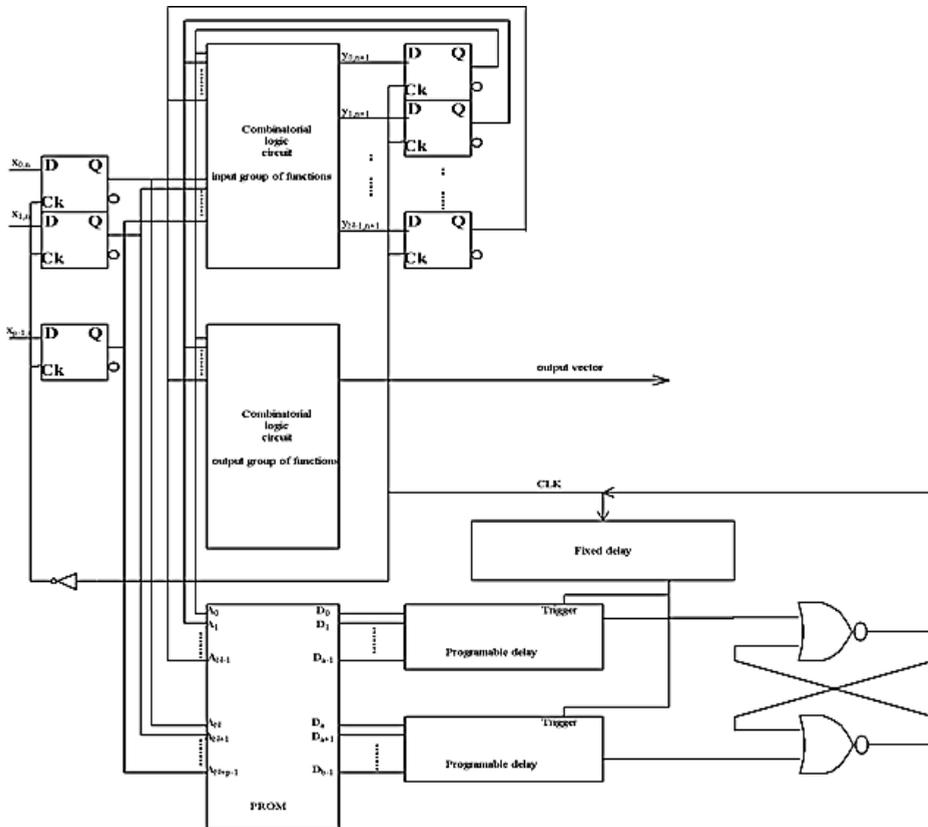


Fig.1. Typical Structure for a Variable Clock Synchronous Automata

3. Example

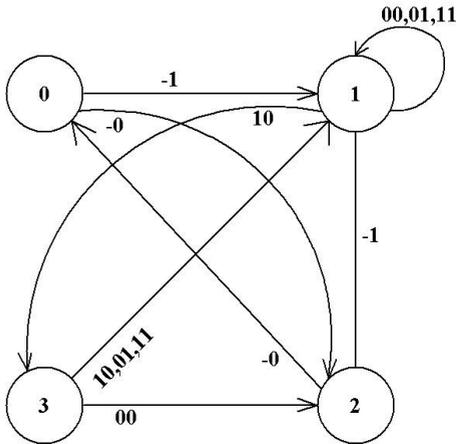
For a better understanding we'll present an example. Let be the sequential system described by the states transition graph presented in figure 2. The excitation table is presented in Tab.1 and a possible implementation of this system is presented in figure 3.

We'll analyze the presence of hazard, according to the method presented above, for the specific implementation of the function group F. Because the flip-flops use have an identical propagation time towards Q and NQ, we shall use, for

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

analysis purposes only, a logic operator characterized by 0/0 ns propagation time and an inverting characteristic.



Q	x_1x_0	00	01	11	10
	y_1y_0				
0	00	11	01	01	11
1	01	01	01	01	10
2	11	00	01	01	00
3	10	11	01	01	01

Tab.1. State encoding & transition table

Fig.2. State transition graph

The system's implementation has been designed considering the reduced equations for the state variables:

$$\begin{aligned}
 \overline{y_{1,n+1}} &= \overline{y_{1,n}x_{1,n}y_{0,n}x_{1,n} + x_{0,n}} \\
 \overline{y_{0,n+1}} &= \overline{x_{1,n} + y_{1,n} + x_{0,n}y_{0,n}x_{0,n}}
 \end{aligned}
 \tag{3.1}$$

Next, for the specific implementation presented above, we'll make the timing analysis in accordance with the classical method. We shall define the limit conditions for the HIGH and LOW periods of the clock. For the LOW period of the clock we'll define the restrictive condition by assuming that the state variables are already stabilized at the moment when we shall strobe the input vector. The waveform that presents the functioning is shown below in fig.4.

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

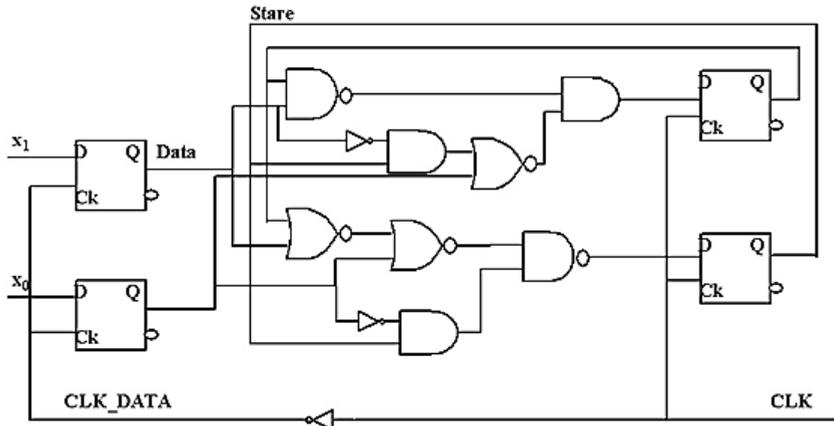
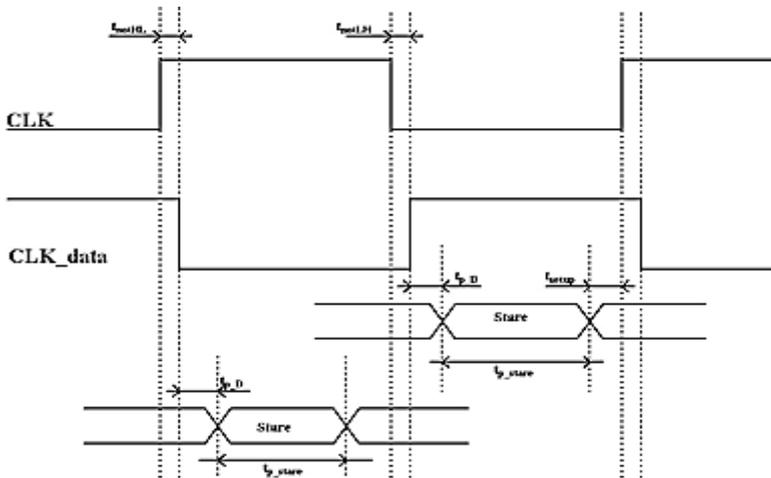


Fig.3 Implementation according to Eq. 3.1.



CLK-low

$$t_{CLK-low,min} = t_{not,max} + t_{D,max} + t_{pstate,max} + t_{setup} = 22ns + 40ns + (22 + 27 + 22 + 27) + 20ns = 180ns$$

CLK-high

$$t_{ps,max} + t_{p-stare,max} + t_{setup,min} \leq t_{CLK-high} + \dots + t_{notLHmax}$$

$$t_{CLK-high} \geq 40ns + 68ns + 20ns - 22ns$$

$$t_{CLK} \geq 110ns$$

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

So if we shall use a normal clock it should have a period larger than 190 ns. If the duty factor of the clock is 50% then the period should be at least 360ns.

4. Defining the variable clock's parameters

We shall identify the circuits that implement the state variables (in our case y_1 and y_0) and then by using one of the methods presented above we shall determine the minimum stabilization time.

Analysis for y_1 – the circuit implementing this variable is presented in fig. 5.

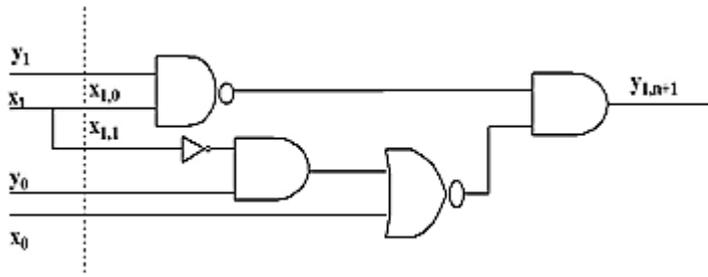


Fig.5. Circuit implementing state variable y_1

The state variable expressed with respect to the secondary input variables is:

$$y_{1, n+1} = (y_{1, n} x_{1,0} y_{0,1} + x_0)$$

The propagation times are:

$$t_{y1} = t_{x1,0} = (t_{and} \otimes t_{inv} + t_{and}) \otimes t_{inv} = 42/41 \text{ ns}$$

$$t_{x1,1} = ((t_{and} \otimes t_{inv} + t_{nor}) \otimes t_{inv} + t_{and}) \otimes t_{inv} + t_{not} \otimes t_{inv} = 73/79 \text{ ns};$$

$$t_{y0} = ((t_{and} \otimes t_{inv} + t_{nor}) \otimes t_{inv} + t_{and}) \otimes t_{inv} = 61/68 \text{ ns}$$

$$t_{x0} = (t_{and} \otimes t_{inv} + t_{nor}) \otimes t_{inv} = 42/31 \text{ ns}$$

Analysis for y_0 – the circuit implementing this variable is presented in fig.6.

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS
AUTOMATA
Nicolae GALUPA, Member IEEE

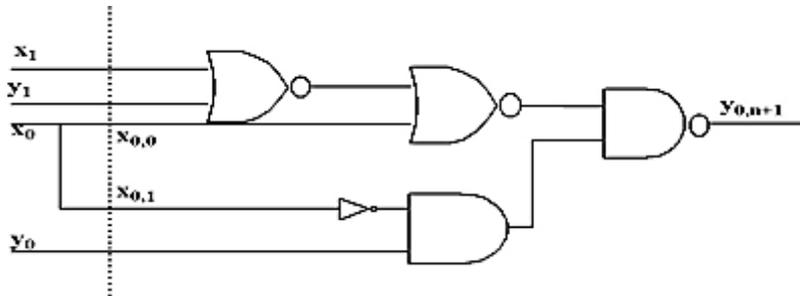


Fig.5. Circuit implementing state variable y_0

The state variable expressed with respect to the secondary input variables is:

$$y_{0,n+1} = \overline{\overline{\overline{x_1 + y_1 + x_{0,0}y_{0,n}x_{0,1}}}}$$

The propagation times are:

$$t_{x1} = t_{y1} = ((t_{\text{nand}} \otimes t_{\text{inv}} + t_{\text{nor}}) \otimes t_{\text{inv}} + t_{\text{nor}}) \otimes t_{\text{inv}} = 52/52\text{ns}$$

$$t_{x0,0} = (t_{\text{nand}} \otimes t_{\text{inv}} + t_{\text{nor}}) \otimes t_{\text{inv}} = 37/37\text{ns}$$

$$t_{x0,1} = ((t_{\text{nand}} \otimes t_{\text{inv}} + t_{\text{nor}}) \otimes t_{\text{inv}} + t_{\text{not}}) \otimes t_{\text{inv}} = 64/66\text{ns}; t_{y0} = (t_{\text{nand}} \otimes t_{\text{inv}} + t_{\text{nor}}) \otimes t_{\text{inv}} = 34/49\text{ns}$$

Generally the propagation times will be:

y_0	y_1
$t_{y1}, t_{x1,0} \rightarrow 42/41\text{ns}$	$t_{x0,0} \rightarrow 37/37\text{ns}$
$t_{x0} \rightarrow 42/31\text{ns}$	$t_{y0} \rightarrow 34/49\text{ns}$
$t_{y0} \rightarrow 61/68\text{ns}$	$t_{x1}, t_{y1} \rightarrow 52/52\text{ns}$
$t_{x1,1} \rightarrow 73/79\text{ns}$	$t_{x0,1} \rightarrow 64/64\text{ns}$

Next we'll determine the stabilization time of the outputs after the negative edge of the clock (input vector loading) and after the positive edge (next state vector loading).

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

5. Timing analysis with respect to input (Clk_low)

The reduced functions to be analyzed are:

y_1y_0	$f_{r,y1,n+1}$	$f_{r,y0,n+1}$
00	$\overline{x_0}$	1
01	$\overline{x_{1,1} + x_0}$	$\overline{x_1 + x_{0,0}x_{0,1}}$
11	$\overline{x_{1,0}x_{1,1} + x_0}$	$\overline{x_{0,0}x_{0,1}}$
10	$\overline{x_{1,0}x_0}$	1

The possible transitions are:

State 0: $y_1y_0=00 \rightarrow 01$

$y_1y_0=00 \rightarrow 11$

$f_{r,y1,n+1} = \overline{x_0}$ - stabilization time for y_1 is 42ns.
 $f_{r,y0,n+1} = 1$ - stabilization time for y_0 is 0ns.

The amount of time after witch the state is perfectly stable is 42 ns

State 1: $y_1y_0=01 \rightarrow 01$

$y_1y_0=01 \rightarrow 10$

the reduced functions are:

$$f_{r,y1,n+1} = \overline{x_{1,1} + x_0}$$

$$f_{r,y0,n+1} = \overline{x_1 + x_{0,0}x_{0,1}}$$

Stabilization times are:

x_1x_0	$ty_{1,n+1}$	$ty_{0,n+1}$	t_{max}
00	79ns	66 ns	79 ns
01	42 ns	37 ns	42 ns
11	42 ns	37 ns	42 ns
10	79 ns	32 ns	79 ns

We notice that in this situation for $y_{0,n+1}$ hazard is present. We shall not try to eliminate or mask it – we'll just go around it.

State 2: $y_1y_0=11 \rightarrow 00$

$y_1y_0=11 \rightarrow 01$

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

The reduced functions are:

$$f_{r_{y1,n+1}} = x_{1,0}x_{1,1} + x_0$$

$$f_{r_{y0,n+1}} = x_{0,0}x_{0,1}$$

In this case too for $y_{1,n+1}$ and $y_{0,n+1}$ hazard is present.

Stabilization times are:

x_1x_0	$ty_{1,n+1}$	$ty_{0,n+1}$	t_{max}
00	79ns	66 ns	79 ns
01	42 ns	66 ns	66 ns
11	42 ns	66 ns	66 ns
10	79 ns	66 ns	79 ns

State 3: $y_1y_0=10 \rightarrow 11$

$y_1y_0=10 \rightarrow 01$

The reduced functions are:

$$f_{r_{y1,n+1}} = x_{1,0}x_0$$

$$f_{r_{y0,n+1}} = 1$$

Stabilization times are:

x_1x_0	$ty_{1,n+1}$	$ty_{0,n+1}$	t_{max}
00	42ns	0 ns	42 ns
01	42 ns	0 ns	42 ns
11	42 ns	0 ns	42 ns
10	42 ns	0 ns	42 ns

By putting together all the times calculated above for the states variables we'll reach the global minimum stabilization times. These are presented in the table below.

x_1x_0	00	01	11	10
y_1y_0				
00	42ns	42ns	42ns	42ns
01	79ns	42ns	42ns	79ns
11	79ns	66ns	66ns	79ns
10	42ns	42ns	42ns	42ns

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

6. Timing analysis with respect to state (Clk_high)

The reduced functions are:

x_1x_0	$f_{r_y1,n+1}$	$f_{r_y0,n+1}$
00	$\overline{y_0}$	y_1y_0
01	0	1
11	0	1
10	y_1	y_0

Stabilization times will be:

$$\begin{aligned}
 x_1x_0=00 \rightarrow & \quad ty_{1,n+1}=68ns \\
 & \quad ty_{0,n+1}=0ns \quad \rightarrow t_{max}=68ns \\
 x_1x_0=01 \text{ si } x_1x_0=11 \rightarrow & \quad t_{max}=0ns \\
 x_1x_0=10 \rightarrow & \quad ty_{1,n+1}=42ns \\
 & \quad ty_{0,n+1}=49ns \quad \rightarrow t_{max}=49ns
 \end{aligned}$$

Putting all the stabilization times together we'll find:

x_1x_0	00	01	11	10
y_1y_0				
00	68ns	0ns	0ns	49ns
01	68ns	0ns	0ns	49ns
11	68ns	0ns	0ns	49ns
10	68ns	0ns	0ns	49ns

7. Defining the synchronizing signal parameters

Now we shall put together all the minimum times calculated above and we shall, for each state, find the clock's parameters. Let us not forget that there are some time components that we have to consider, such as t_{p-s} , t_{setup} , t_{hold} ...etc. That is why we shall not reach the asynchronous sequential system's parameters – we shall only get close them. By using the times calculated above we'll find:

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

REFERENCES

- [1] Al. VALACHI, "Analysis, synthesis and testing of digital devices", Ed. NORD_EST Iasi, pp.278-288, 1993
- [2] N.GALUPA, "Analysis method for hazard determination in combinatorial structures", 8th international conference on control systems and computer science, vol III, pp.87-100, Bucharest, 1991
- [3] N.GALUPA, Al. VALACHI, "Enhancement of the time constants method for hazard analysis", Proceedings of AED (Advanced Engineering Design), Prague, 2003
- [4] N.GALUPA, "USING TIME DEPENDENT VARIABLES FOR HAZARD ANALYSIS", Proceedings of IEEE_CCECE, Montreal, Canada, 2003, DOI: 10.1109/CCECE.2003.1226003
- [5] R.B.McGHEE, "Some aids to the detection of hazards in combinatorial switching circuits", IEEE Trans.Comp.Vol C-18, pp. 561-565, 1969
- [6] E.J.McCLUSKEY, "Transients in combinational logic", in Redundancy Techniques for Computing Systems, R.H.Wilcox and W.C.Mann, Ed: Washington – Spartan, pp.9-46, 1962
- [7] E. J. McCluskey, "Logic Design Principles", Prentice-Hall, Englewood Cliffs, NJ, 1986.
- [8] J.Beister, "A unified approach to combinational hazards", IEEE Trans.Comp.VolC-23, pp. 566-575, 1974, 10.1109/T-C.1974.223996
- [9] Richard F Tinder, "Engineering Digital Design", Second Edition, Elsevier - ACADEMIC PRESS, 2001
- [10] R. Bryant, "Graph-based Algorithms for Boolean Function Manipulation," IEEE Trans.Comp., 677-691 (1986), 10.1109/TC.1986.1676819
- [11] S. Akers, "Binary Decision Diagrams," IEEE Trans.Comp., C-27, 509-516 (1978). DOI:10.1109/TC.1978.1675141
- [12] S.M.Nowick, C.W.O'Donnell "On the existence of hazard-free multi-level logic" Proceedings / IEEE ASYNC 03, May12-16,2003, DOI: 10.1109/ASYNC.2003.1199171
- [13] C.Jeong, S.M.Nowick "Fast hazard detection in combinational circuits" ACM DAC 04, June 7-11, 2004, 10.1109/DAC.2004.240453
- [14] N.Galupa, "Increase of Sequential Systems Performance Using Digital Hazard Analysis", ICCS/ISITA '92, 1096-1100, ISBN: 0-7803-0803-4, 10.1109/ICCS.1992.255092
- [15] K.S. Stevens, R. Ginosar, and S. Rotem, Relative timing [asynchronous design], in IEEE Transactions on VLSI Systems, pp. 129-140, ISSN 1063-8210, 2002, 1.1109/TVLSI.2002.801606
- [16] R. Ben Salah, M. Bozga and O. Maler, "On Timing Analysis of Combinational Circuits," In FORMATS'03, LNCS 2791, pages 204-219. Springer (2003)

ASYNCHRONOUS VS VARIABLE CLOCK SYNCHRONOUS AUTOMATA

Nicolae GALUPA, Member IEEE

- [17] R. Ben Salah, M. Bozga, O. Maler, "On Timed Components and their Abstraction," In SAVCBS'07 Workshop, ACM ISBN 978-1-59593-721-6/07/0009 (2007)
- [18] M. D. Riedel, J. Bruck, "Timing Analysis of Cyclic Combinational Circuits," Proceedings of the International Workshop on Logic and Synthesis, Temecula Creek, CA, 2004