SIMULATIONS OF DIFFERENT HOMOGENOUS T GATES TRISTABLES

Oleg Cernian

University of Craiova
Computer and Communications Engineering Department
13, Alexandru Ioan Cuza Street
200585 CRAIOVA, ROMANIA

Abstract: This paper deals with a brief presentation of two types of particular tristables realized with hazard free T-type gates. By means of a ternary logic simulator the operation of these tristables was checked to prove correctness of their functional tables. A detailed explanation of both the operation and the outputs of the experiments is delivered.

Keywords: 5–10 keywords (taken from the IFAC list on this website), 127 mm (5 in) width (maximum), left justified.

1. INTRODUCTION

It is known that the set of ternary T-gates (Lee and Chen, 1956) are forming a logically complete set allowing convenient realizations of ternary combinational and sequential functional blocks (Cernian, 1998b; Higuchi and Kameyama, 1975ab; Thelliez, 1973). The T-gates technology was developed conjointly with VLSI implementations. In (Higuchi and Kameyama, 1977) it was proposed a new type of T-gates realization with multiple applications. For applications of sequential nature using structures with T-gates the main concern is related to the presence of the static hazard. Higuchi and Kameyama (1977) redefined the structure of a T-gate as implemented in the Post algebra technical system (Hurst, 1979; Porat, 1969) and proved that it works hazard free for certain Selection signal transitions (from 1 to 0 and from -1 to 0).

On the other hand, there are known many classification systems for elementary ternary memory components called, by extension, tristables. Some widely used criteria are discussed in (Cernian, 1982; Cernian, 1998a; Etiemble, 1974; Hurst, 1979; Porat, 1969). In this work the focus is given to T type gates tristable realizations with hazard free feature and their functional simulations on a simulation tool created for ternary logic operation. This simulation tool is based on Aldec Active-HDL platform (Active, 2005) and is briefly discussed in (Cernian, 2004). The simulations have the role to prove the correctness of tristable structures operation, as well as to reveal some of the specific timing characteristics.

According to the prime classification criterion, referring to the order of the tristables, the designed structures belong to order 1 class, where the state is given by a single output Q. From the functional criterion point of view, the analyzed set of tristables belongs to D-type class.

2. D-TYPE LATCH TRISTABLE

All presentations in this paper are carried out in ternary symmetric system using logic symbols {-1, 0, 1}. The following single structure containing two T-type gates (Fig. 1) corresponds to a D-type Latch tristable whose operation is controlled by the binary signal EN.

The operation of the analyzed structure corresponds to a typical Latch circuit where the “latching” action takes place on the transition of EN from 1 to 0. As long as EN = 1 the operation is characterized by the transparency, where any change at D is transmitted to the output Q.

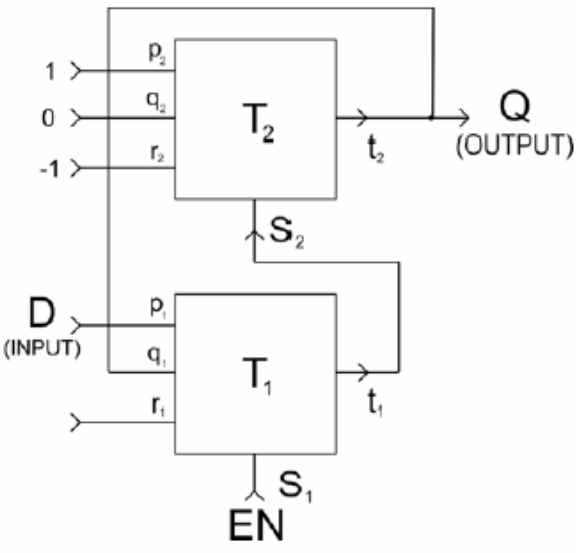

Fig. 1. D-type Latch tristable

From the schematics it is seen that $t_2 = S_2 = t_1$. If EN = 1, then $S_1 = 1$ and $t_1 = p_1 = D$, and consequently Q = D. When EN is changed to 0, $t_1 = q_1 = t_2 = Q$, thus the input D is disregarded and the past value of the output Q is preserved by the feedback from t_2 to q_1.

Thus, the latch action took place on the transition of EN from 1 to 0. As long as EN = 0 the state of the latch remains unchanged, regardless logical values of the input D. The delay on the path D – Q is equal to the sum of delays corresponding to a T gate.

Assuming that a RESET condition is requested, the EN signal will be changed to -1, whereas the "not connected" input r1 will be tied to -1. In this way, since EN = -1, S1 = -1, t1 = r1 = -1, S2 = -1, t2 = Q =- 1, through feedback q1 = Q = -1, so that when EN returns to 0, the RESET state (Q = -1) is preserved (t1 = q1 = S2 = r2 = Q = -1). The functional table for this D type latch tristable is given in Table 1.

Table 1. Functional table for D-type latch tristable

Operation Mode	EN	D	Q
Reset	-1	X	-1
Transparency	1	-1	-1
	1	0	0
	1	1	1
Latch	0	X	d

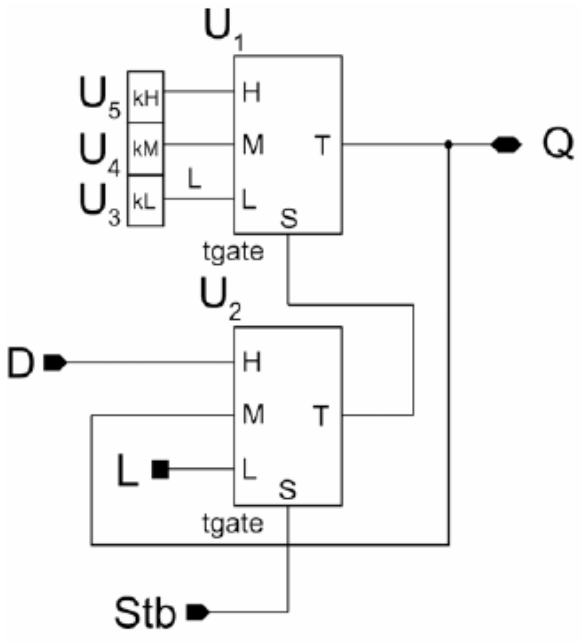

Fig. 2. Screenshot for D-type latch tristable

3. SIMULATION OF D-TYPE LATCH TRISTABLE

The screenshot presenting the implemented latch tristable in the ternary simulation is given in Fig. 2. The logic values -1, 0 and 1 are represented by symbols L, M and H, respectively.

In Fig. 3 it is presented a screenshot of the experiments performed on D-type latch tristable. The EN signal is symbolized Stb (Strobe). Both D and EN inputs were manually defined to encompass all cases to prove the correctness of the analyzed structure. It is seen that when Stb is H the latch is transparent (cases 1, 2, 3, 6, 7, 8) i.e. all changes at D are transferred to the output Q. When Stb is changing to M the latch action takes place by "catching" the last value of D (cases 4, 9). When Stb is M any change at D is ignored by the output Q (cases 5, 10, 11). The initial value of the output Q was chosen -1 (Reset condition – case 0).

By means of two cursors it can be exactly measured the delay Z of Q change. As mentioned before the input state D is sensed at the output Q with a delay given by propagation through two T gates (each T gate has a predefined delay Δ = 10 nsec, as chosen by the experimenter).

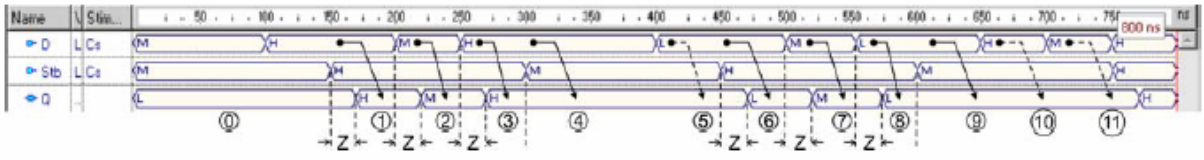

Fig. 3. Screenshot for experiments on D type latch tristable

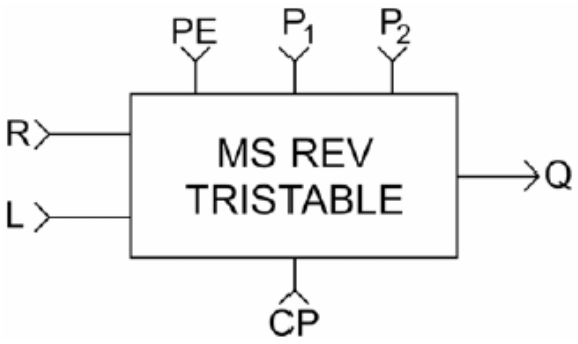

Fig. 4. Block diagram for MS REV Tristable

4. PRESENTATION OF THE MS REV TYPE TRISTABLE

The block diagram of the considered tristable is given in Fig. 4.
where:

R is the synchronous input for right shift
L is the synchronous input for left shift
PE is the control for parallel load and selection of the inputs P_1 or P_2
P_1 is the asynchronous parallel input for right shift
P_2 is the asynchronous parallel input for left shift
Q is the output
CP is the clock pulse considered either
$0 \rightarrow 1 \rightarrow 0$ or $0 \rightarrow -1 \rightarrow 0$.

The role of PE control is to select between Preset Enable and Count Enable conditions. If PE = 0 then the Count Enable condition is validated allowing the synchronous operation of the tristable under control of the CP. If PE ≠ 0 then the Preset Enable control is selected and, depending upon the effective value of PE, either the parallel input P1 or the parallel input P2 is selected. Hence, the parallel load is an asynchronous operation since it does not depend upon the CP.

The mnemonic REV is assigned to indicate that it is a reversible tristable allowing a proper selection of inputs from a selection set of two.

Thus if PE = 0 then the sequence $0 \rightarrow 1 \rightarrow 0$ at CP realizes transfer of the input R to the output Q. But ,if the opposite sequence $0 \rightarrow -1 \rightarrow 0$ is applied on the CP, then the input L is transferred to the output Q. This facility is very useful for construction of reversible registers and ternary counters.

When PE = 0 the parallel asynchronous inputs are ignored.

If PE = 1, then CP is ignored and the asynchronous parallel input P_1 is transferred to the output Q. Eventually, if PE = -1, then CP is ignored and the asynchronous parallel input P_2 is transferred to the output Q.

Table 2. Function table for MS REV tristable

PE	R	L	CP	P1	P2	Q	Operating Mode
0	R	X	$0\rightarrow1\rightarrow0$	X	X	R	Right Shift
0	X	L	$0\rightarrow-1\rightarrow0$	X	X	L	Left Shift
1	X	X	0	P_1	X	P_1	Preset P_1
-1	X	X	0	X	P_2	P_2	Preset P_2

Therefore P1 is considered a preset for “right shift”, whereas P2 is considered a preset for “left shift”.

By summarizing these descriptions the function table corresponding to the operation of this type of tristable is derived (Table 2).

5. THE LOGIC DIAGRAM AND OPERATION OF THE MS REV TRISTABLE

The detailed logic diagram of this MS REV tristable realized with 4 T-type gates is depicted in Fig. 5.

The tristable is a Master-Slave type consisting of two distinct stages, the first formed of the gates T1 and T2, the second formed of the gates T3 and T4, where {T1,T2} represents the Master stage, while {T3,T4} represents the Slave stage.

If PE = 1, then it is selected the input p2 of the gate T2. Since CP must be logic “0”, the input q1 of the gate T1 is selected. Consequently, the asynchronous input P1 is transferred to the output t2 of the gate T2, hence Q2 = P1, which is applied on the input q1 of T1, forcing thus t1 of gate T1 to become P1, thereby Q1 = P1. Thus, the master stage yields the output Q1 = P1.

In the Slave stage, since CP = 0, the input q4 of the gate T4 is selected, transferring the constant 1 to the output Q4, which represents the selection input S3 at the gate T3.

A similar explanation would be given for asynchronous left input, when PE = -1 and CP = 0.

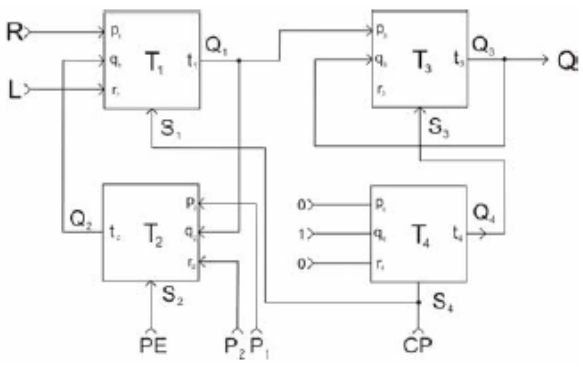

Fig. 5. Logic Diagram of MS REV tristable

Next, it is considered PE = 0 selecting thus the synchronous mode. If PE = 0, then the inputs p2 and r2 are ignored and, therefore, the parallel preset data P1 and P2 are disregarded. Initially CP = 0 and, consequently, the inputs q2 and q4 are selected. Hence, Q1 = Q2 in the loop, preserving in the Master stage the previous logic value. In the Slave stage, Q4 = 1and, therefore, S3 = 1, selecting the input p3 which is equal to Q1, resulting that the global output Q is equal to Q1.

Hence, initially the output for the Slave stage is identical to that of the Master stage. It is assumed a change at the CP from 0 to 1. Then, in Master stage S1 = 1, selecting now the input p1 and, therefore, the output of Master becomes R, hence, Q1 = R; simultaneously in Slave stage the selection input S4 = 1 and the output Q4 is changed to 0 (Q4 = p4 = 0). Consequently, S3 = 0 and, hence, the input q3 is selected, realizing a loop by retransferring the output Q to q3; hence, when CP =1, the new value is stored in the Master stage, while the Slave stage is preserving the previous value.

When CP returns to 0 it is repeated the scenario described previously, i.e. the new input R is preserved inside the Master stage, whereas the new value R is transferred through the gate T3 to the global output Q of the tristable. It results that the essential switch at CP is changing from 1 to 0, that is the falling edge to 0, when the new value from Master is passed to Slave.

A similar operation is valid for a negative pulse at CP, when for a double change 0 → -1 → 0, the asynchronous input L is written in the tristable (i.e. Q = L); the essential edge is now the rising one, i.e. passing from -1 to 0.

By using the developed simulator, the above discussed logic diagram is entered by defining correspondingly the inputs R, S, P1, P2, PE, Clk, the output Q and the constants KH and KM (for logic values 1 and 0 that are inputs at the gate T4) the following screenshot is derived (Fig. 6).

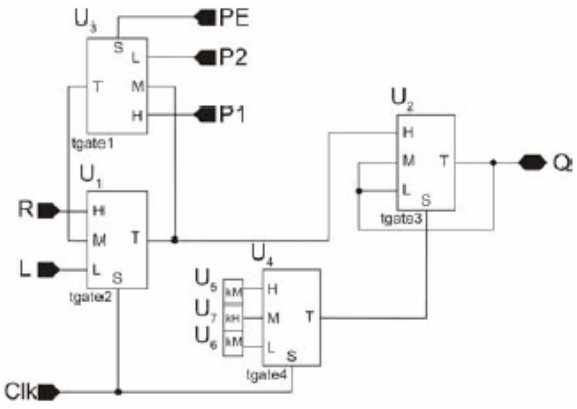

Fig. 6. Screenshot obtained with Active-HDL simulator

6. SIMULATIONS WITH MS REV TRISTABLES

The following four simulations are carried out on the presented structure:

Simulation No.1: synchronous operation mode, where PE = 0, i.e. PE = "M"; then the inputs P1 and P2 were arbitrarily put at -1, i.e. "L". For CP it was defined the sequence 0 ⟶1 ⟶ 0 ⟶1, i.e. "M" ⟶ "H" ⟶ "M" ⟶ "H" etc., specified by the formula. There was input manually a set of values for the relevant synchronous input R (right shift) defining different scenarios such that for each 0 ⟶1 ⟶ 0 sequence of CP to pinpoint the corresponding output Q. The initial value of Q was chosen -1. Any changes of R, when CP = 0, are not sensed by Master stage, therefore no change at the output is expected. After CP changes to 1 the Master stage becomes transparent to R, but Slave will preserve the precedent logic value. The simulation run revealed 5 cases of evolution of the output Q, from -1 to 1, from 1 to -1, from -1 to 0, from 0 to -1 and from -1 to 0. More, it was shown that modifications of R are not relevant for Q, barring the falling edge of CP from 1 to 0 (Fig. 7).

The propagation time Z is clearly pinpointed in Fig 7.

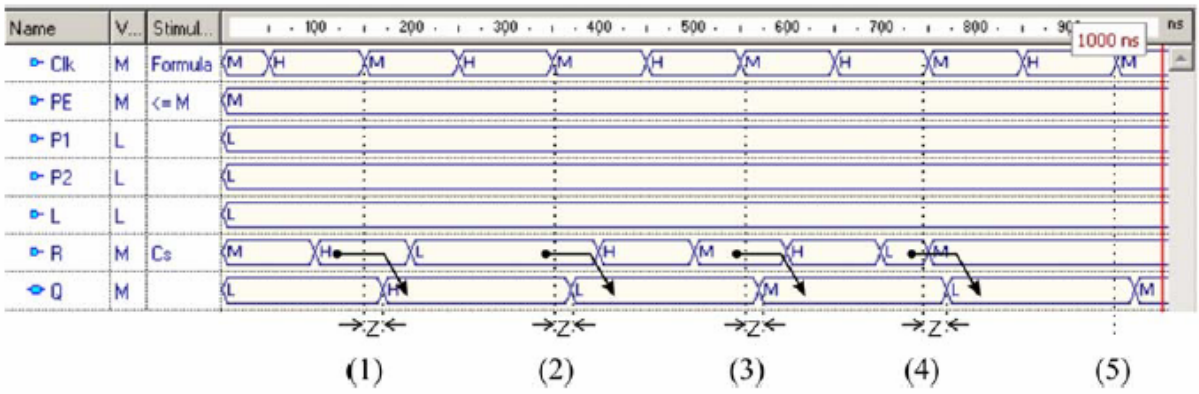

Fig. 7. Timing Diagram for Simulation 1

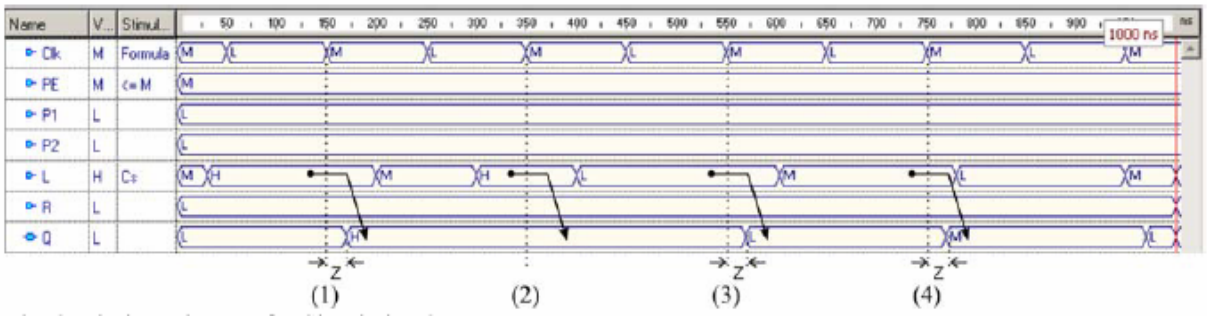

Fig. 8. Timing Diagram for Simulation 2

Simulation No.2: likewise the precedent simulation, the other synchronous input L was considered. The relevant edge this time it was the rising edge, from -1 to 0 (from "L" to "M"). There were considered four possible transitions, proving the correctness of the operation.

More, on a rigorous timescale it can be seen the propagation time, Z, for each change at the output Q. In both simulations the period of CP was chosen 200 nsec. The set-up and hold time would not be pinpointed, as they depend on the electrical characteristics of the T gates, which were not taken into consideration in these simulations. The prime concern was afforded to the correct logic running of the tristable (Fig. 8). It is seen that Z = 20 nsec, a delay defined by propagation through two T gates.

Simulation No.3: asynchronous mode of operation, when PE = 1 and CP fixed at 0. The synchronous inputs R and L are not considered in this simulation and were fixed at -1. The preset action is controlled by PE; therefore the variation of PE was imposed by a formula where with a period of 200 nsec.

PE changes values between 1 and 0. It is shown that when PE = 0 the actual values of P1 are disregarded, the output Q being preserved from the precedent value. Only when PE changes to 1, the parallel asynchronous input P1 is to be considered and , with a delay Z, it is supplied at the output Q. Any change of P1, while PE = 1, is transmitted with the delay Z at the output Q. The screenshot of the simulation includes four valid intervals of asynchronous writing, each of 200 nsec. It is seen that when PE = 0 any change of P1 is not sensed by the output Q (Fig. 9).

Simulation No.4: it is similar to Simulation No.3, but refers to parallel preset P2, called "left shift preset". This asynchronous operation mode is defined for PE = -1 while CP = 0. For PE evolution it was chosen a formula for running through values 0 and -1, with a period of 400 nsec. It is shown that whenever PE = -1 the logic values from P2 are transferred to the output Q with a delay Z, while when PE = 0 the output Q does not sense any change at P2 preserving the preceding value (Fig. 10).

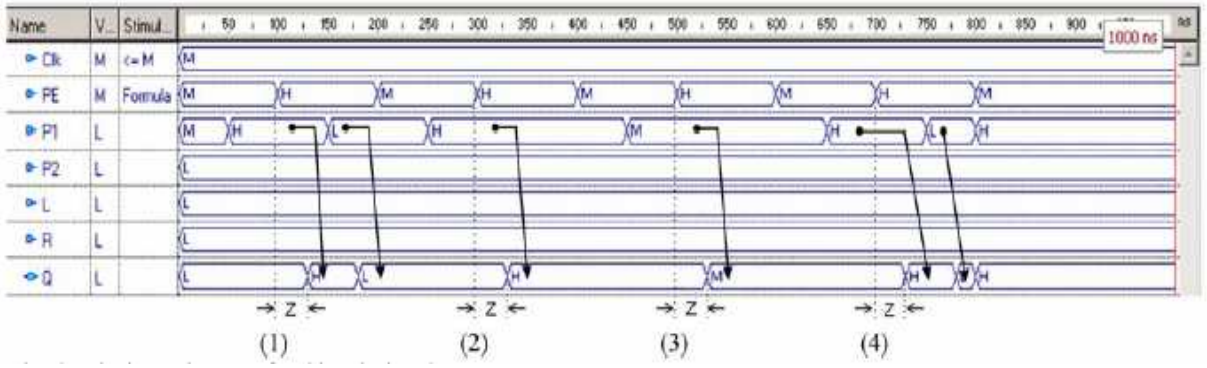

Fig. 9. Timing Diagram for Simulation 3

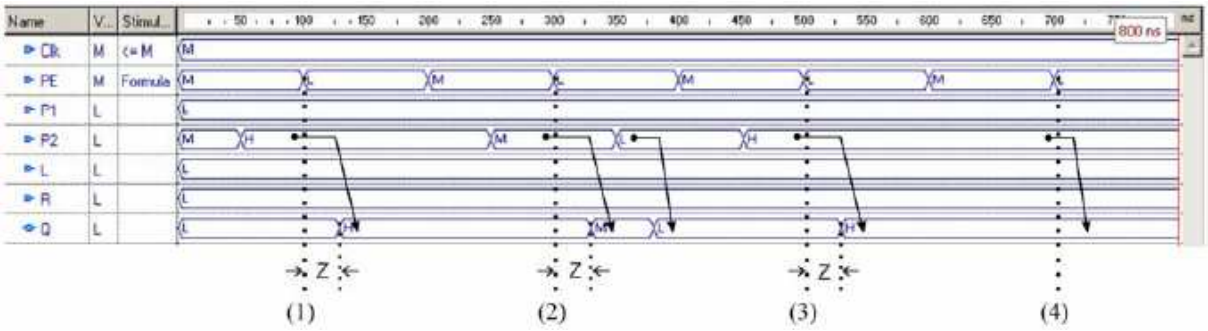

Fig. 10. Timing Diagram for Simulation 4

7. CONCLUSIONS

The ternary simulator realized with facilities offered by the Active-HDL platform proved its utility by allowing specification of ternary logic structures and running experiments on them. The outputs obtained by checking the operation of two particular types of tristables confirmed the versatility of this simulator providing efficient timing diagrams marked clearly with ternary symbols and revealing timing correlations between different signals.

REFERENCES

Cernian, O.and Ionescu, A. (1982). Variante de elemente tristabile binar-ternare. Proceedings al Simpozionului National de Teoria Sistemelor, Craiova.

Cernian, O. (1998a). Clasificări de circuite basculante tristabile. Universitatea din Craiova, Raport intern de cercetare, Catedra de Calculatoare.

Cernian, O. (1998b). Implementarea structuratã a funcþiilor ternare cu ajutorul portilor T. Universitatea din Craiova, Raport intern de cercetare, Catedra de Calculatoare.

Cernian, O. (2004). *Contribuþii la Analiza si Sinteza Sistemelor Logice Ternare.* Tezã de doctorat, Universitatea Tehnicã Gh. Asachi Iaşi.

Etiemble, D. (1974). *Contribution a l'étude et a la modelisation des circuits logique.* These docteur ingenieur, Université Paris VI.

Higuchi, T. and Kameyama, M. (1975a). Ternary System based on T-Gate. Proceedings of IEEE Int. Symp. on Multiple-Valued Logic.

Higuchi, T. and Kameyama, M. (1975b). Synthesis of Multiple Value logic networks based on Tree type Universal Logic Modules. Proceedings of IEEE Int. Symp. on Multiple-Valued Logic.

Higuchi, T. and Kameyama, M. (1977). Static Hazard Free T-Gate for Ternary Memory Element and its Application to Ternary Counters. *IEEE Trans on Computers*, **vol. C26, No.12**.

Hurst, S.L. (1979). *The Logical Processing of Digital Signals*. Crone-Russek

Lee, C. Y., Chen, N. H. (1956). Several Valued Combinational Switching Circuits, *Trans AIEE*, **vol. 75**.

Porat, D. I. (1969). Three Valued Digital Systems. *Proc of IEE*, **vol. 110**.

Thelliez, S. (1973), *Introduction a l'étude des structure ternaires de commutation*. Gordon & Breach

XXX (2005), *Active-HDL 6.3 Student Edition*. Prentice Hall.