

Title: Ternary Digital System: Concepts and Applications

Authors: A P Dhande, V T Ingole, V R Ghiye

Published by SM Online Publishers LLC

Copyright © 2014 SM Online Publishers LLC

ISBN: 978-0-9962745-0-0

All book chapters are Open Access distributed under the Creative Commons Attribution 3.0 license, which allows users to download, copy and build upon published articles even for commercial purposes, as long as the author and publisher are properly credited, which ensures maximum dissemination and a wider impact of the publication. Upon publication of the eBook, authors have the right to republish it, in whole or part, in any publication of which they are the author, and to make other personal use of the work, identifying the original source.

Statements and opinions expressed in the book are these of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published chapters. The publisher assumes no responsibility for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained in the book.

First published October, 2014

Online Edition available at www.smgebooks.com

For reprints, please contact us at ebooks@esciencemedicine.com

CAD Tool Simulation of Basic Ternary Logic Gates and Map Method for Ternary Function Minimization

INTRODUCTION

There are many EDA (Electronic Design Automation) tools are available for the simulation of ternary circuits. EDA tools are listed as: Electronic work bench, Multisim, p-spice, Micro wind, Tanner spice (T-spice) and VHDL. Results given in the chapter are simulated using tanner T-spice software and multisim tools. Chapter covers complete simulated logic circuit for T-NOT and T-AND/NAND gate by using T-spice, Electronic work bench, VHDL tool with prototyping which can be fabricate directly from foundry. This chapter also covers map reduction method along with algorithm and map reduction software is provided with CD ROM.

TANNER T-SPICE

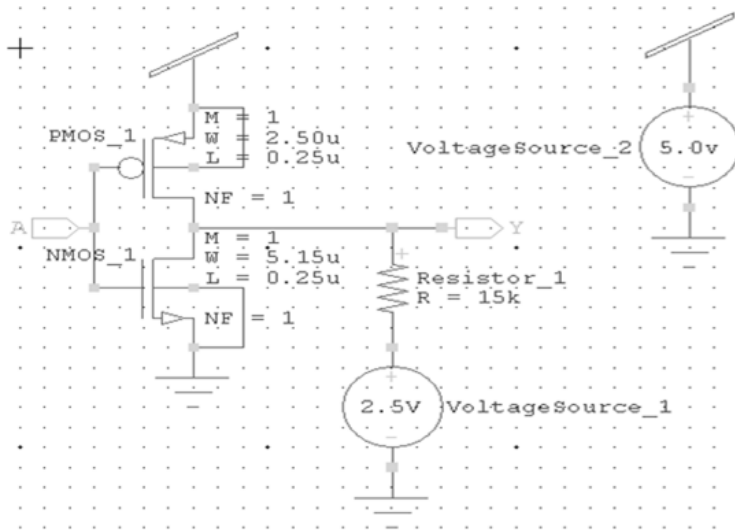
T-Spice is a complete design, capture and simulation solution that provides accuracy and convergence with proven reliability. To transform an idea into designs, Tanner spice (T-spice) EDA tool facilitates to simulate large circuits quickly and with a high degree of accuracy. This simulation tool offers fast run times, integrates with other design tools, and is compatible with industry standards. Tanner spice (T-spice) EDA consists of four sub-packages namely T-Spice, S-Edit, L-Edit, and W-Edit. T-Edit is an analog simulation of a circuit, S-Edit is for schematic capture, L-edit is layout editor and W-Edit is for wave form viewing and analysis. Generally sequence flow is from S-Edit, L-Edi, T-Splice and W-Edit. Detail about the software is available on website www.tannereda.com.

Here as an example, simulation of Simple Ternary inverter ST-NOT and simple ternary ST-NAND operation is shown that involves the use of software sequence mentioned above up to prototyping, lay outing and mask preparation. Figure 7.1 (a) is a T-Spice simulation of ST-NOT gate. Figure 7.1 (b) involves both S-Edit and L-Edit procedure and Figure.7.1(c) shows result of the simulated circuit with W-Edit. Further, it is possible to use SILVACO platform to fabricate a ICs for simulated circuit.

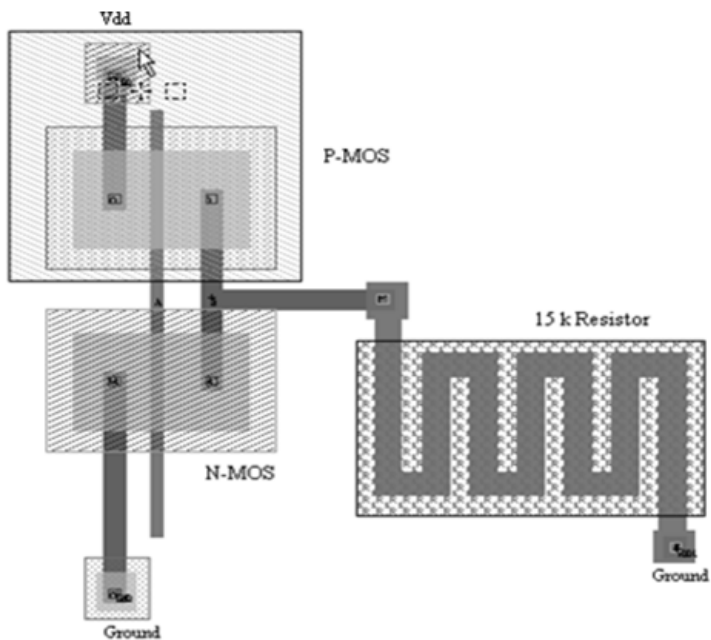
In Figure 7.1(c), waveform in pink colour is an input waveform and in blue is corresponding output. Truth table for simulated circuit is shown in Table 7.1.

Table 7.1: Truth table for ST-NOT.

Input	Output
0	2
1	1
2	0



(a)



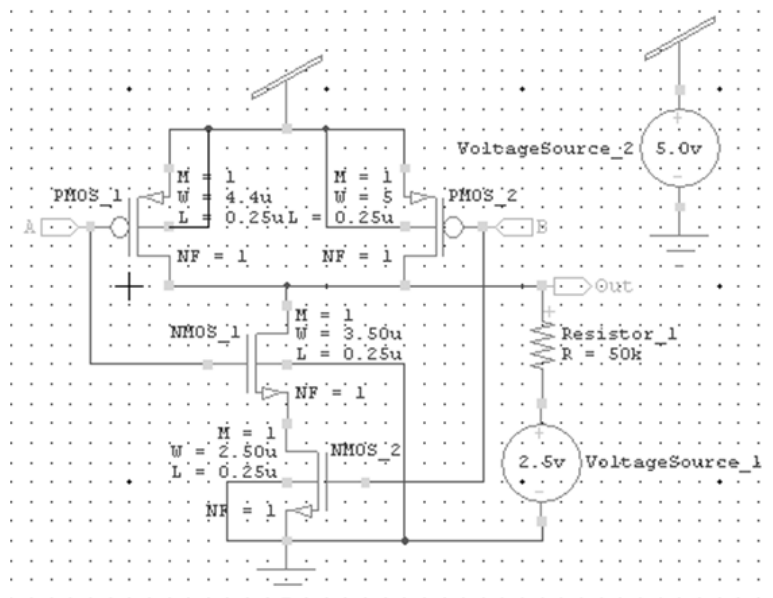
(b)



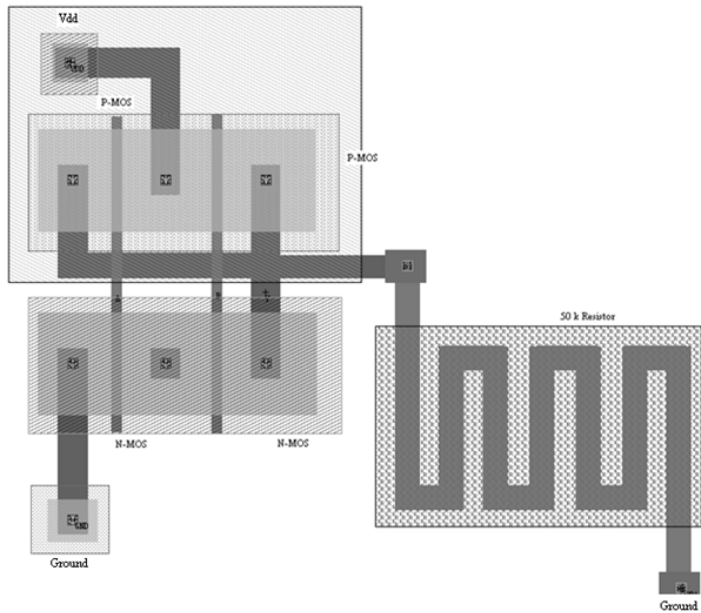
(c)

Figure 7.1: (a) T-Simple Inverter circuit (b) Lay out for Simple T-Interter, (C) Waveforms for T-Simple Inverter

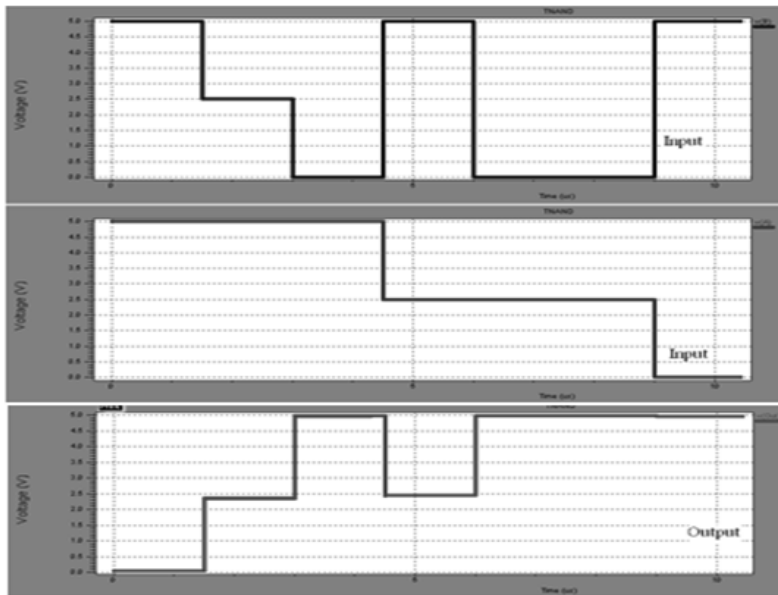
The circuit for ST-NOT is designed according to the calculated parameters and then simulated with the software tanner tool. A first part of it is a schematic editor; the circuit is drawn and simulated using S-edit part of software. Simulated circuit is exported to Tanner spice where the simulated results are obtained by spice techniques.



(a)



(b)



(c)

Figure 7.2: (a) schematic Circuit for ST-NAND (b) Layout for ST-NAND (c) Waveform for ST-NAND.

After spice simulation, the circuit is exported to Layout editor for lay outing them and DRC rules and net listing is automatically carried out by software. Once these steps are completed, the

GDS 2 file is ready and this file is send to lab for fabrication. The Same steps are repeated with ST-NAND gate. Results can be verified according to truth table for ST-NAND GATE in chapter 3.

Tanner EDA provides a complete line of software solutions that catalyze innovation for the design, layout and verification of digital and Mixed-Signal (A/MS) Integrated Circuits (ICs). Researchers can create breakthrough applications in areas such as digital and Mixed-Signal (A/MS) Integrated Circuits (ICs)

ELECTRONIC WORK BENCH/MULTISIM

Electronic Work Bench/ Multisim are an industry-standard, best-in-class SPICE simulation environment. It is used for practical application in designing, prototyping, and testing digital circuits.

The Electronic Work Bench /Multisim design approach helps to save prototype iterations and optimize Printed Circuit Board (PCB) designs earlier in the process. The Electronic Work Bench /Multisim Professional Edition provide the SPICE simulation and prototyping tools for reliable circuit design. Multisim has been optimized for usability to ensure domain experts and researchers can quickly realize their PCB designs.

Stepwise Simulation

Consider design of Simple ternary inverter. Here it is needed to have two MOSFETs family P and N channel enhancement. These are dragged from device window provided by software. Device parameters are set by double clicking/editing the device parameters and connections are made.

The supply connections of +, - and ground are made as per Figure shown in 7.3. Inputs are given to the gate of each transistor as shown and output is observed. Same procedure can be repeated for design and simulation of other gates. Circuits are shown in Appendix II.

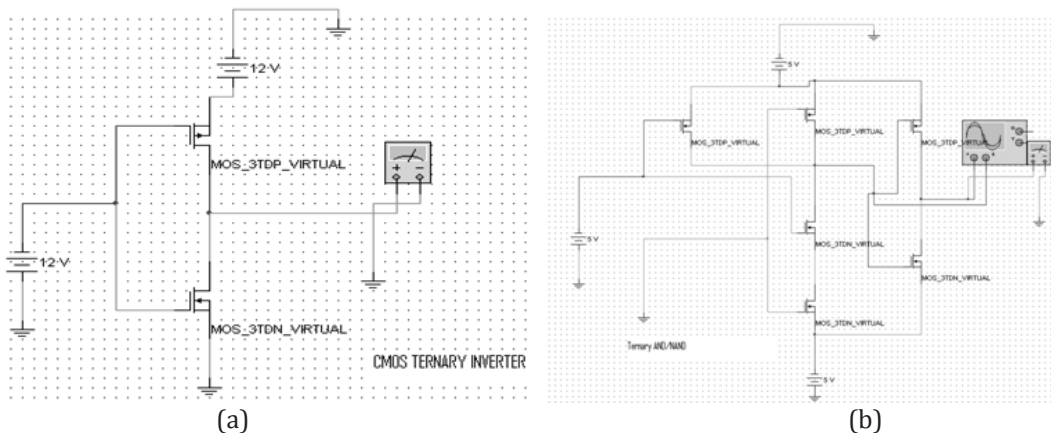


Figure 7.3: (a) Ternary inverter, (b) Ternary AND/NAND gate

VHDL FOR TERNARY GATE SIMULATION

An existing Very High-Speed Integrated Circuit Hardware Description Language (VHDL) which is basically binary circuit's code simulator how it can be use as a logic simulator and to evaluate the performance of ternary circuits is illustrated.

To demonstrate the use of VHDL as a ternary logic simulator, it uses Logic 0 to represent logic level 0 state, High impendence Z to represent logic 1 state and Logic 1 to represent logic level 2 state. The logic state table in VHDL is as below.

Table 7.2: Logic state in VHDL.

Symbols	Values assigned
U	Uninitialized
X	Unknown
0	Logic level 0
1	Logic level1
Z	High impedance
W	Weak Unknown
L	Weak Zero
H	Weak One
-	Don't Care

Stepwise Procedure for Implementation of T-Gates Using VHDL

The procedure for implementing T-gates can be summarized as follows:

Step 1: Establish the truth table for T-gates.

Step 2: Decide the description style of VHDL. This can be behavioral, Dataflow, structural or mixed of all offered by VHDL.

Step 3: Develop the VHDL constructs i.e. packages, libraries, components and entities.

Step 4: Use the construct developed in step 3 to implement ternary functions.

Step 5: Use the features of the simulator to conduct the verification of the functionality and timing specifications of the ternary system. The simulator normally displays signal waveforms using one level for logic 1, one level for logic 0, and a third level (shown half-way in between logic 1 and logic 0) for all other levels, strengths, disconnection, don't cares, unknowns, etc. (normally distinguished by colors and/or other attributes). One way to circumvent this problem is to build the VHDL constructs such that Z is intermediate level, which is displayed by blue color in the simulation. The VHDL code written for STI is given in Appendix III.

Example: 1-trit ternary multiplication. Table 7.2 shows ternary multiplier for 1-trit. Figure 7.4 shows K-map for product and carry. Figure 7.5 shows simulation results using VHDL.VHDL codes are given in Appendix III.

Table 7.3:1-trit ternary multiplier.

X1	X2	Product	Carry
0	0	0	0
0	Z	0	0
0	1	0	0
Z	0	0	0
Z	Z	Z	0
Z	1	1	0
1	0	0	0
1	Z	1	0
1	1	Z	Z

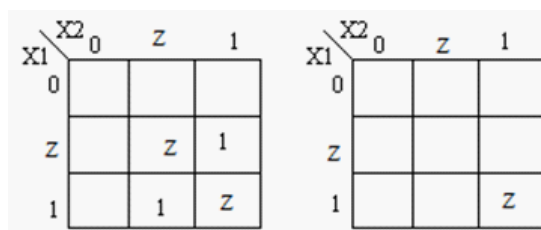


Figure 7.4: K-map for Product and Carry.

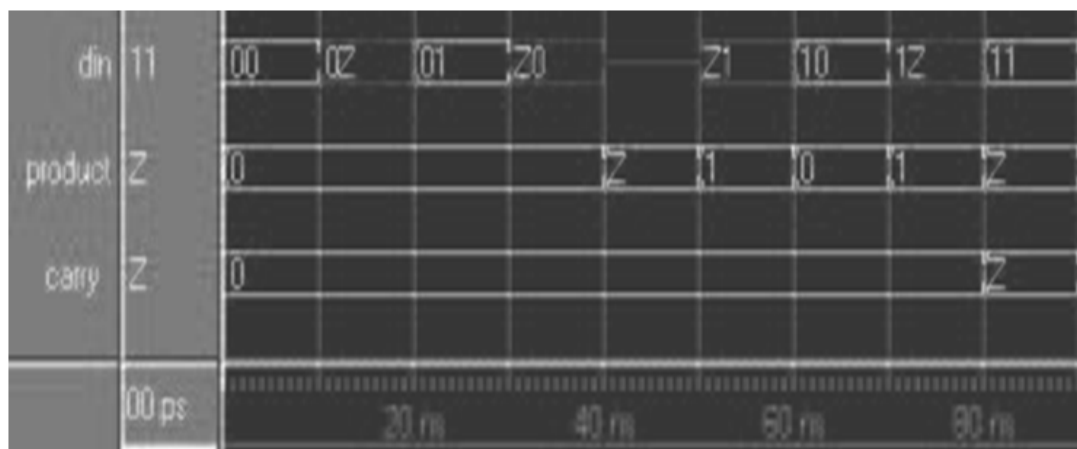


Figure 7.5: Simulation results.

TERNARY FUNCTION MINIMIZATION BY MAP METHOD

A software tool is developed for minimization of ternary function. We know that as the radix of system increases, the difficulties in the minimization or reduction of logic function is get increases. It becomes difficult and tedious for manual reduction for higher radix to reduce the function design equation. The proposed program is developed for the ternary system function minimization which directly gives minimized output only by taking user input in the map. It

incorporates all designed rules for ternary logic system design and gives the output in the form of Sum-of-Product (SOP) terms.

There are different rules for the minimization of ternary logic based digital system that are different from the conventional binary digital system. Output equation of ternary digital system is in the form of $F = F_2 + 1 \cdot (F_1)$ where, $F_2 = 2$'s minterms and $F_1 = 1$'s minterms. Some of the example has been tested and output becomes accurate when compared with the truth table. An output functional set which has been reduced to its absolute minimal form is according to the priority cost metrics. The remaining reduced functional sets are evaluated based on the degree of approximation to the absolute minimal functional set. There degrees reflect the order and number of metric matches relative to the absolute functional set of a function basis. Functions associated with higher degrees shall have satisfied the requirements for any lower degrees. The coarse or lowest degree is connected to product term equivalence, succeeded by MIN-literal matching, and finally equal fundamental gate quantity having the highest degree. The biggest or finest degree refers to expressions which are reduced to their absolute smallest form.

There are limitations for using map minimization software. As the number of variables increases more than 4 variables, it becomes difficult to represent the map and also grouping of the cells becomes more time consuming for the system. Other methods for minimization i.e. modified Quine McClukey and Schinman's modified ternary method can be used for minimization of ternary function having variable more than 4. Algorithm for map minimization technique is given below along with illustrative example.

Algorithm

- 1) Accept the equation containing variables.
- 2) Identify the number of variables.
- 3) Depending upon the number of variables, identify the size of map i.e. 2,3 or 4 variables
- 4) Check the user entered values of 2's and 1's term for given k-map.
- 5) Follow the rules given for ternary function minimization in section 1.5 of chapter 1.
- 6) Form the groups of 2's and 1's as per rule.
- 7) Reduce the equation containing 2's and 1's term. 2's entry can be considered as don't care entry for grouping 1's term.
- 8) Represent output equation in the form of $F = F_2 + 1 \cdot (F_1)$

Example: Ternary Full adder: This circuit takes in to account carry generated in previous stage. Table 7.4 is truth table for full adder.

Table 7.4: Truth table for full adder.

A	B	C _{in}	Sum	Carry _{out}
0	0	0	0	0
0	0	1	1	0
0	0	2	2	0
0	1	0	1	0
0	1	1	2	0
0	1	2	0	1
0	2	0	2	0
0	2	1	0	1
0	2	2	1	1
1	0	0	1	0
1	0	1	2	0
1	0	2	0	1
1	1	0	2	0
1	1	1	2	0
1	1	2	0	1
1	2	0	0	1
1	2	1	1	1
1	2	2	2	1
2	0	0	2	0
2	0	1	0	1
2	0	2	1	1
2	1	0	0	1
2	1	1	1	1
2	1	2	2	1
2	2	0	1	1
2	2	1	2	1
2	2	2	0	2

$$F_2 = A_0B_2C_0 + A_0B_1C_1 + A_0B_0C_2 + A_1B_1C_0 + A_1B_0C_1 + A_1B_2C_2 + A_2B_0C_0 + A_2B_2C_1 + A_2B_1C_2$$

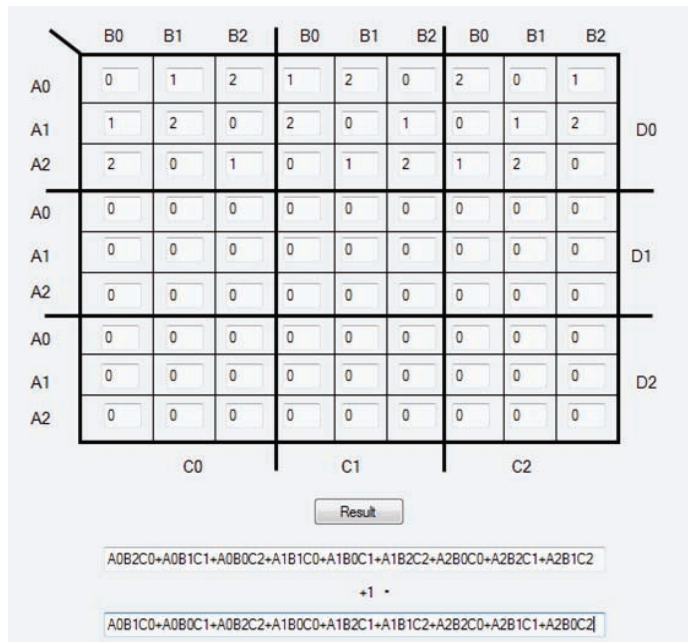
$$F_1 = A_0B_1C_0 + A_0B_0C_1 + A_0B_2C_2 + A_1B_0C_0 + A_1B_2C_1 + A_1B_1C_2 + A_2B_2C_0 + A_2B_1C_1 + A_2B_0C_2$$

Therefore, final reduced equation is:

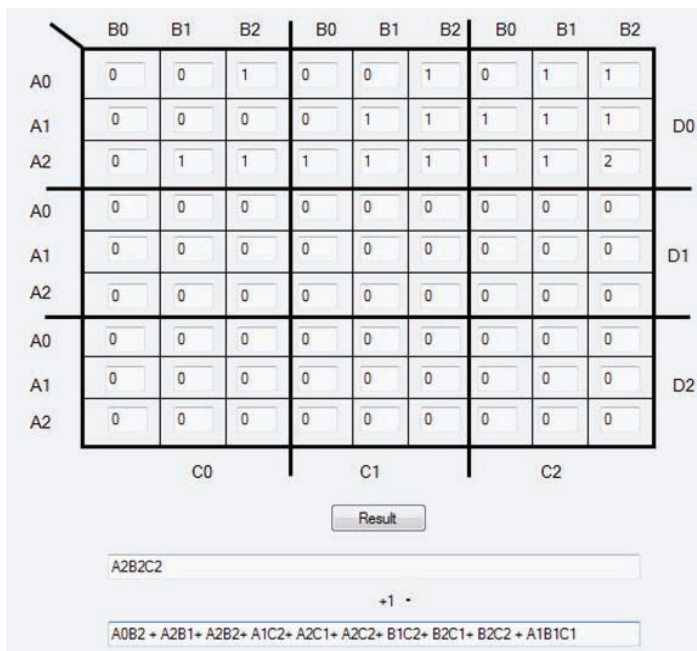
$$F_{\text{sum}} = A_0B_2C_0 + A_0B_1C_1 + A_0B_0C_2 + A_1B_1C_0 + A_1B_0C_1 + A_1B_2C_2 + A_2B_0C_0 + A_2B_2C_1 + A_2B_1C_2 + 1 \bullet (A_0B_1C_0 + A_0B_0C_1 + A_0B_2C_2 + A_1B_0C_0 + A_1B_2C_1 + A_1B_1C_2 + A_2B_2C_0 + A_2B_1C_1 + A_2B_0C_2)$$

$$F_{\text{carry}} = A^2B^2C^2 + 1 \bullet (A^2C^2 + A^0B^2 + A^2B^2 + A^1C^2 + A^2C^1 + B^2C^1 + B^1C^2 + B^2C^2 + A^1B^1C^1)$$

By using the software results for Sum and Carry obtained are same as shown in Figure 7.6 (a) and (b).



(a)



(b)

Figure 7.6: (a) Sum output, (b) Carry output.

Thus developed tool for map minimization is validated for minimization of ternary function.

References:

1. A.P.Dhande, S.S.Narkhede, S.S.Dudam 'Design and fabrication of CMOS ternary AND/NAND, OR/NOR, and NOT logic gates', BCUD research project report 2010.
2. A.P.Dhande, R.C.Jaiswal, S.S.Dudam 'Ternary logic simulator using VHDL', SETIT, 2007, 4 th international conference on Science of Electronic, technologies of information and Telecommunication, March 2007.
3. VR Ghiye, AP Dhande 'Ternary Function Minimization by Map Method' IEEE conference on Communication Systems and Network Technologies (CSNT), 2014.