THE 1ST LEWIS
The 1st Lawang Sewu International Symposium
Volume 1, November 2022

E-ISSN : 2987 - 2685
page 103 - 115

Research on The Application of Digital Electronics Education and Design Suiter Simulator (DEEDS) in Electronic Design

Laily Muntasiroh[1*], Ilham Yustar Afif[1]
[1]Faculty of Engineering, Universitas Muhammadiyah Semarang, Indonesia
[*]Corresponding author : lailymuntasiroh@unimus.ac.id

ABSTRACT

The research aims to determine the effectiveness of the Digital Electronics Education and Design Suite Simulator to design a series of detector primes number. This research method begins by determining the problem boundary, where this binary number circuit is only able to perform 2 (two) bit binary operation to detect the primes number more less than 9. Those, the minimum value of the multiplication results is 0000 with a decimal value of 0 to 1111 with a decimal value of 15. The design is done by simplifying the problem by using the Boolean theorem. The results of this research are the construction of simulation circuit that can be used to perform detect primes number more less than 9 at 2 (two) binary numbers. Simplification of the circuit with the Boolean theorem method using Digital Electronics Education and Design Suite Simulator is able effectively design series of binary multipliers with an accuracy value of 100%.

Keywords: binary numbers, digital circuit, Boolean theorem

1. INTRODUCTION

At this time technology is developing very rapidly. This has an impact on computer programming which is closely related to the field of electronics design, one of which is digital circuits. The design of logic and digital circuits is a representation of the theorem of Boolean Algebra. In compiling a logic circuit can consist of various logic gates[1]. The required logic gates can be one or more. The number of logic gates used is of course in accordance with the level of complexity of a planned circuit[2]. Complex digital circuits consisting of various logic gates or commonly known as combination circuits are also interesting for research. Complex digital circuits consisting of various logic gates or commonly known as combination circuits are also interesting for research[3].

Given the importance of basic digital circuits to be used as control and control circuits in the electronics field, there are many studies that discuss digital circuits, including; research conducted by [3] which discusses combinational circuits. Combinational circuits are an advanced part of some logic circuits [4] create a prototype to convert basic logic gates. The research roadmap related to the prototype is continued [5] to assemble the digital Half Adder ALU circuit. Furthermore, the development of research conducted by [6] who makes models and simulations of prototype digital gate conversion circuits. In this study

E-ISSN : 2987 - 2685
page 103 - 115

the logic gates used are basic AND, OR, NOT gates into Not AND and Not OR gates. To complete the research roadmap for the digital circuit prototype, a comparison study of the model and simulation of the half Adder ALU digital circuit prototype was carried out [5]. Advances in technology have led to research on Digital Circuits that can be represented by Virtual Reality. In other studies, digital circuits can be represented by mobile devices [7].

Boolean algebra is a systematic and logical way of writing, analyzing and simplifying logical equations [8]. Boolean algebra is used in several studies in the field of electronic control. In Boolean algebra there are 2 terms that we must understand, namely variables and complements. A variable is a symbol that represents an action, condition or data [9]. Examples of variables: A, B, C, X, Y and so on. While the complement is the opposite of the variable and is indicated by a line symbol above the letter or overbar. Example: $\overline{A}$, $\overline{B}$, $\overline{C}$, $\overline{X}$, $\overline{Y}$, and so on [10].

Boolean algebraic functions contain a variety of very complex operations. Also contains terms that are more than one kind, where the function can actually be simplified [11]. This simplification of the function is necessary to minimize the logic gates required [12]. Boolean algebra that has been simplified and has not been simplified will undergo a very significant change. Boolean functions can be simplified in various ways, including; by the tabulation method, the mapping method, and by writing a simplification table. Research conducted by [2] using the Karnaugh Map method. The tabulation method or better known as the Quine McCluskey method was developed by W.V Quine and E.J. McCluskey in 1950 [10].

If applied with computer programming, the tabulation method, a very systematic method, is the right choice. Boolean functions that can be simplified by this method must first be converted into Sum of Product (SoP) form. Those, the Boolean function in the form of Product of Sum must be converted into the form of SoP. Another study was conducted by applying the Karnaugh Map method in simplifying the Boolean function [2].

With the development of technology also has an impact on the simulators used in planning a circuit. As we know there are several kinds of simulators including; Liveware, Proteus, Electronic Workbench, Ni Multisim, Simulink and DEEDS. On research [4] using Matlab Simulink application to design logic gates. In another study, using the Ni Multisim simulator to execute digital circuit design [5]. In addition, other research conducted by [3] using Electronic Workbench to represent digital circuits. Digital Electronics Education and Design Suite Simulator or known as DEEDS is able to represent user needs to design or plan a logic circuit. The choice of the DEEDS simulator is because the application is equipped with supporting components, one of which is the seven segments as an output indicator.

2. METHOD

The limitation of this research problem is a series of multipliers of 2 2-bit binary numbers. For more details, see the system flow diagram in Figure 1.

The stages of the research method are as follows:

1) Problem formulation

E-ISSN : 2987 - 2685
page 103 - 115

The formulation of the problem is carried out to analyze the input and output in the circuit, as well as to limit the problems to be discussed.

2) Create a circuit design
In addition to making a circuit design, at this stage the author also analyzes the needs of the required components.
3) Literature study
Literature studies are needed to strengthen the state of the art of this research.
4) Analysis of system requirements
At this stage the author analyzes the needs of the simulation both in determining the input and output needed in the design of a binary number circuit.
5) System design
At this stage the author designs the system by designing and simulating the circuit.
6) Circuit testing
At this stage the author conducts testing of the scenarios that have been made.

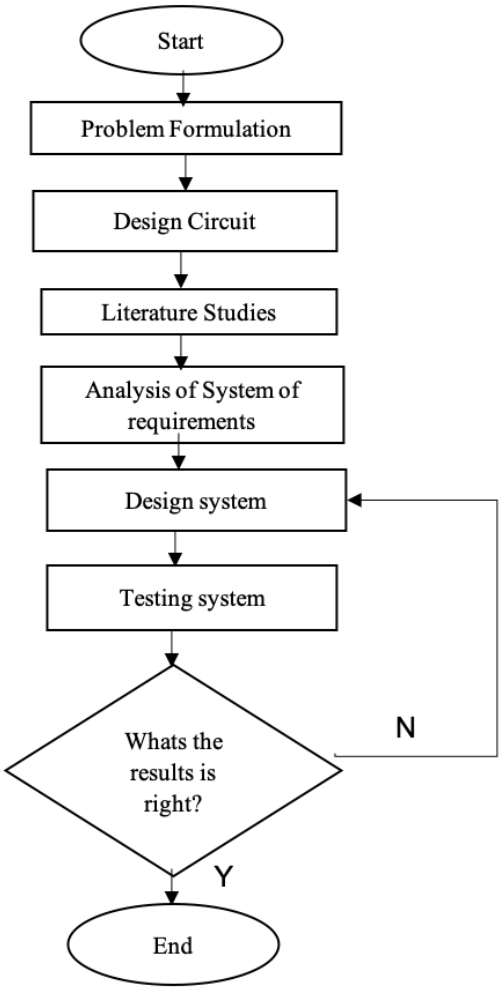

Figure 1*. Diagram System*

E-ISSN : 2987 - 2685
page 103 - 115

2.1. Analysis of System of requirements

In this section, it is explained about the need for designing a DEEDS software system to design a prime number detector circuit provided that the number has 4 (four) binary input numbers and the output will be logical 1 (one) if the number is a prime number.

2.2. Analysis Input and Output

The prime number detector circuit has 4 (four) binary numbers as input and will have 1 (one) output which will be worth 1 if the binary number is a prime number less than 9, namely; 2, 3, 5, and 7.

a. Input initialized with the ABCD
b. Output is initialized with the X

2.3. Truth Table

From the input and output relationships, the next step is to write a truth table. This truth table will be tested on a circuit that has been designed using the DEEDS application.

From the truth table then the next step is to group the output logic 1 by adding. And for the effectiveness of the circuit, the output equation must be simplified using the Boolean Algebra theorem. The final form of the equation will be s-of p (sum of product).

2.4. Simplification with Boolean Algebra

❖ Output X

X = ABCD + ABCD + ABCD + ABCD
= ABC (D + D) + ABD (C + C)
= ABC + ABD

Where:
A, B, C, D is inverse of A, B, C and D

2.5. Design System

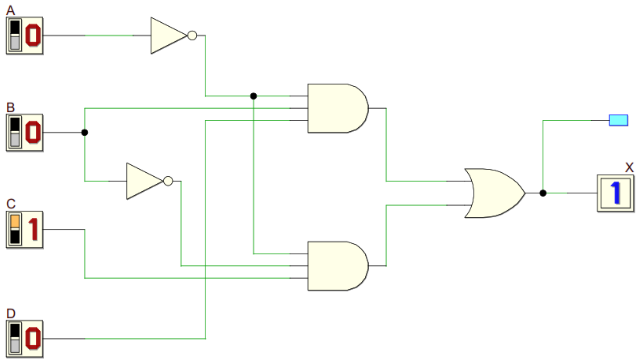

Figure 2. *Circuit design*

E-ISSN : 2987 - 2685
page 103 - 115

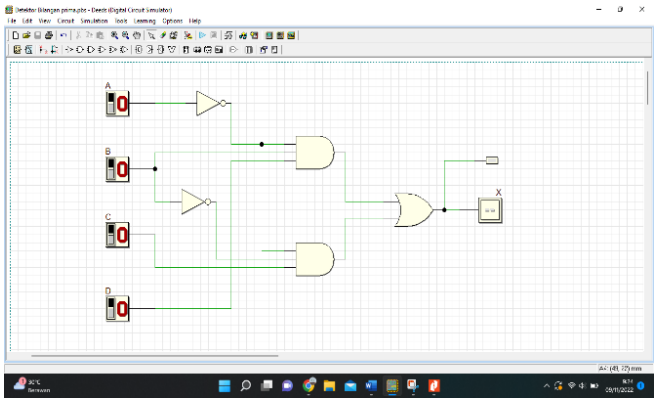

Figure 3*. Display on the DEEDS app*

Table 1. The Truth Table of System

A	B	C	D	X
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Table 2. Scenario Testing System

A	B	C	D	X0	LED
0	0	0	0	0	OFF
0	0	0	1	0	OFF
0	0	1	0	1	ON
0	0	1	1	1	ON
0	1	0	0	0	OFF
0	1	0	1	1	ON
0	1	1	0	0	OFF
0	1	1	1	1	ON
1	0	0	0	0	OFF
1	0	0	1	0	OFF
1	0	1	0	0	OFF

E-ISSN : 2987 - 2685
page 103 - 115

A	B	C	D	X0	LED
1	0	1	1	0	OFF
1	1	0	0	0	OFF
1	1	0	1	0	OFF
1	1	1	0	0	OFF
1	1	1	1	0	OFF

3. RESULTS AND DISCUSSION

After conducting a series experiment simulation on the DEEDS application, the circuit output shows results that are in accordance with the test scenario. The input consists of 4 binary numbers A B C and D with values 0 0 0 0 to 1 1 1 1 with decimal values 0 to 15. The results of the detection of these prime numbers are represented at output X. This output is in the form of binary numbers, namely logic 0 and 1, and in order to make it easier to read the number detector circuit, the output is connected to an LED. The LED will light when the circuit detects a prime number and goes out when it is not a prime number. The detailed results of the experiment are shown in Figure 5-16.

3.1. Scenario 1

Input A B C D = 0 0 0 0
Decimal = 0
Prime Number < 9 = No
Output = 0
LED = OFF

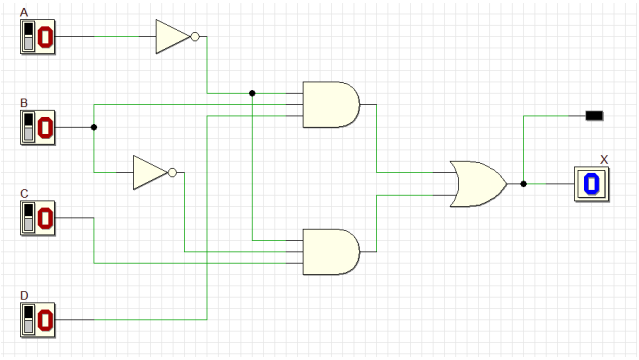

Figure 4*. Scenario testing 1*

3.2. Scenario 2

Input A B C D = 0 0 0 1
Decimal = 1
Prime Number < 9 = No
Output = 0
LED = OFF

E-ISSN : 2987 - 2685
page 103 - 115

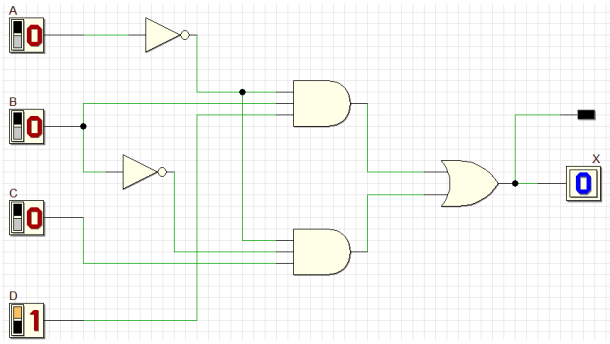

***Figure 5**. Scenario testing 2*

3.3. Scenario 3

Input A B C D = 0 0 1 0
Decimal = 2
Prime Number < 9 = Yes
Output = 1
LED = ON

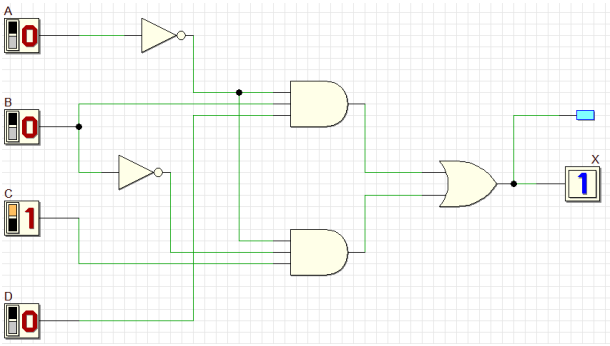

***Figure 6**. Scenario testing 3*

3.4. Scenario 4

Input A B C D = 0 0 1 1
Decimal = 3
Prime Number < 9 = Yes
Output = 1
LED = ON

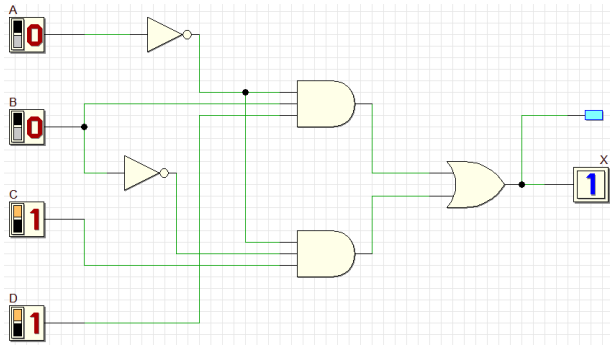

***Figure 7**. Scenario testing 4*

E-ISSN : 2987 - 2685
page 103 - 115

3.5. Scenario 5

Input A B C D = 0 1 0 0
Decimal = 4
Prime Number < 9 = No
Output = 0
LED = OFF

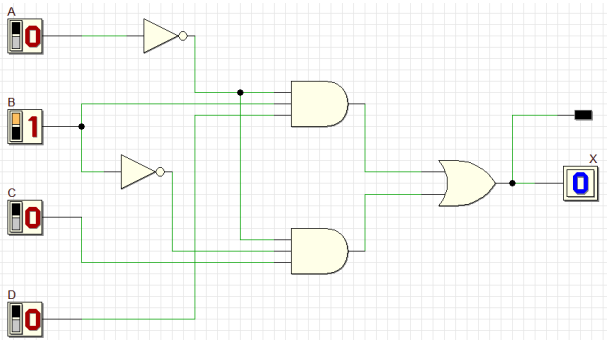

***Figure 8**. Scenario testing 5*

3.6. Scenario 6

Input A B C D = 0 1 0 1
Decimal = 4
Prime Number < 9 = Yes
Output = 1
LED = ON

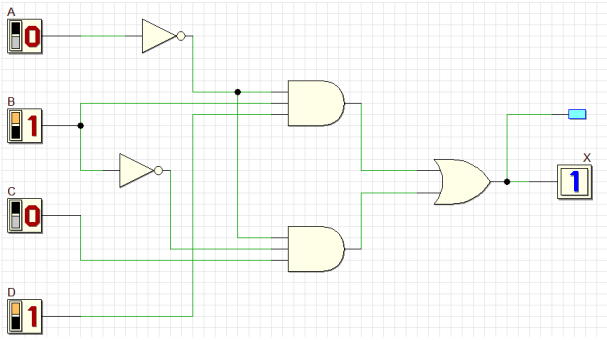

***Figure 9**. Scenario testing 6*

3.7. Scenario 7

Input A B C D = 0 1 1 0
Decimal = 6
Prime Number < 9 = No
Output = 0
LED = OFF

E-ISSN : 2987 - 2685
page 103 - 115

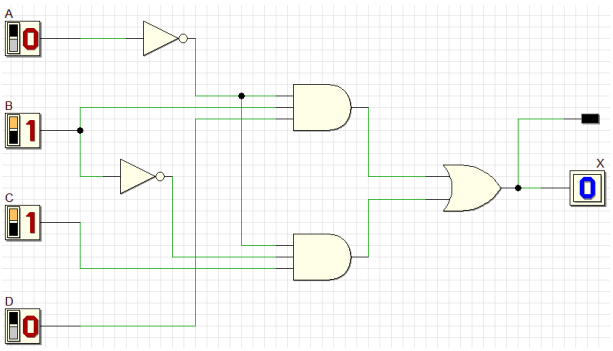

***Figure 10**. Scenario testing 7*

3.8. Scenario 8

Input A B C D = 0 1 1 1
Decimal = 7
Prime Number < 9 = Yes
Output = 1
LED = ON

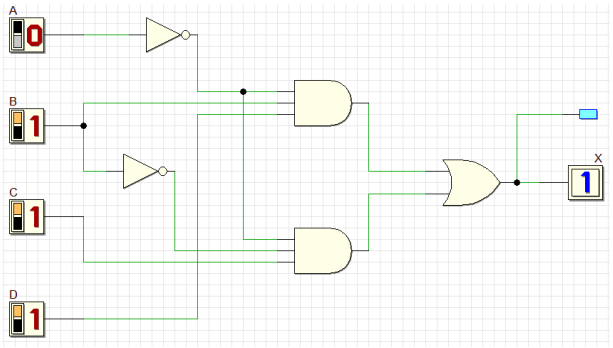

***Figure 11**. Scenario testing 8*

3.9. Scenario 9

Input A B C D = 1 0 0 0
Decimal = 8
Prime Number < 9 = No
Output = 1
LED = ON

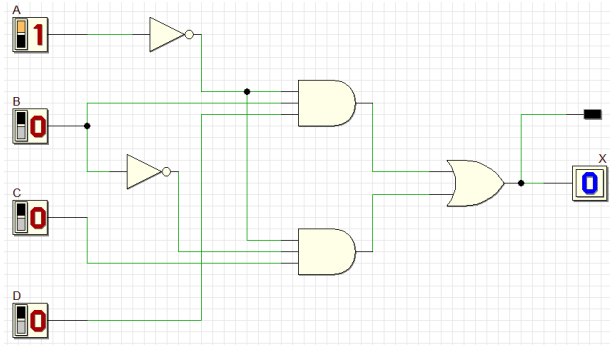

***Figure 12**. Scenario testing 9*

E-ISSN : 2987 - 2685
page 103 - 115

3.10. Scenario 10

Input A B C D = 1 0 0 1
Decimal = 9
Prime Number < 9 = No
Output = 0
LED = OFF

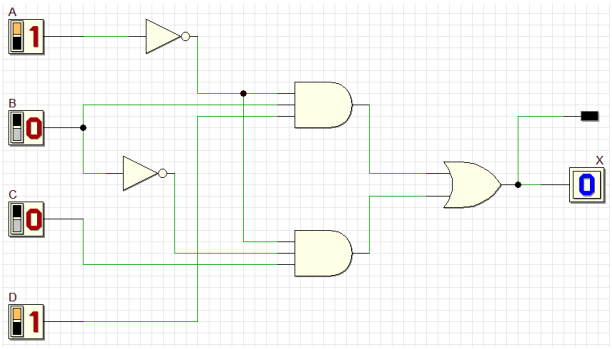

***Figure 13**. Scenario testing 10*

3.11. Scenario 11

Input A B C D = 1 0 1 0
Decimal = 10
Prime Number < 9 = No
Output = 0
LED = OFF

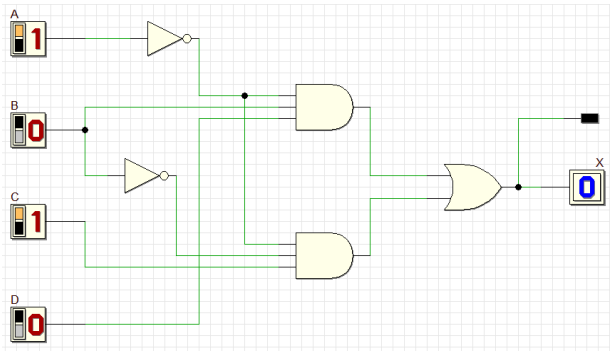

***Figure 14**. Scenario testing 11*

3.12. Scenario 12

Input A B C D = 1 0 1 1
Decimal = 11
Prime Number < 9 = No
Output = 0
LED = OFF

E-ISSN : 2987 - 2685
page 103 - 115

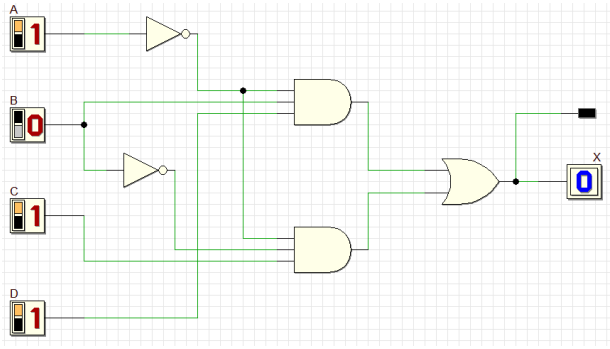

***Figure 15**. Scenario testing 12*

3.13. Scenario 13

Input A B C D = 1 1 0 0
Decimal = 12
Prime Number < 9 = No
Output = 0
LED = OFF

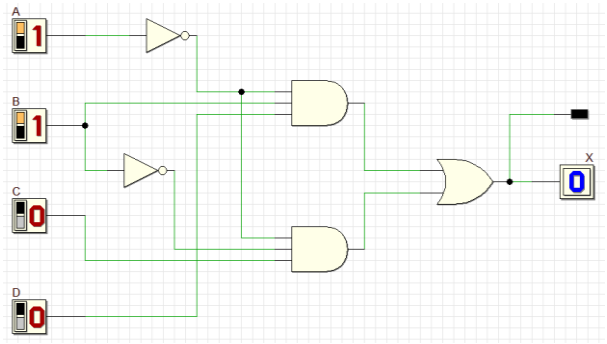

***Figure 16**. Scenario testing 13*

3.14. Scenario 14

Input A B C D = 1 1 0 1
Decimal = 13
Prime Number < 9 = No
Output = 0
LED = OFF

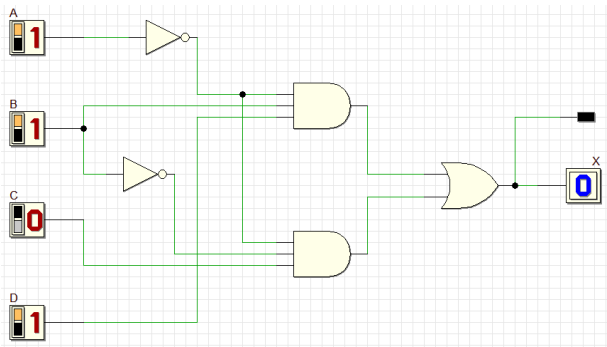

***Figure 17**. Scenario testing 14*

E-ISSN : 2987 - 2685
page 103 - 115

3.15. Scenario 15

Input A B C D = 1 1 1 0
Decimal = 14
Prime Number < 9 = No
Output = 0
LED = OFF

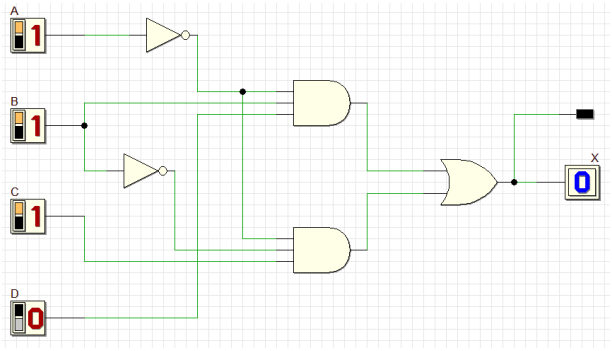

***Figure 18**. Scenario testing 15*

3.16. Scenario 16

Input A B C D = 1 1 1 1
Decimal = 15
Prime Number < 9 = No
Output = 0
LED = OFF

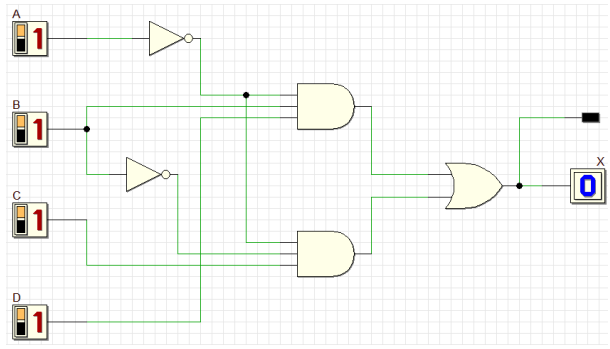

***Figure 19**. Scenario testing 16*

Scenario system at the truth table as same as on the Application of Digital Electronics Education and Design Suiter Simulator (DEEDS) in Electronic Design.

4. CONCLUSION

The result of this research is the construction of a simulation circuit that can be used to detect 4-bit binary numbers that are less than 9. The accuracy obtained is in accordance with the expected results, which is 100%.

E-ISSN : 2987 - 2685
page 103 - 115

REFERENCES

[1] M. A. Desima, J. T. Elektro, R. Digital, K. Praktikum, I. Pendahuluan, and R. Logika, "RANCANG BANGUN KIT PRAKTIKUM RANGKAIAN ELEKTRONIKA DIGITAL," pp. 1–6.

[2] W. Wamiliana, S. S. Zahroh, and O. D. E. Wulandari, "Pengembangan Aplikasi Penyederhanaan Aljabar Boolean Dalam Bentuk Sum-of-Product Dengan Menggunakan Metode Quine Mccluskey," *J. Komputasi*, vol. 1, no. 2, pp. 50–58, 2013.

[3] F. Saputra, C. A. Ichsan, and M. Silalahi, "Physical Phenomenon Simulation in Circuit," *J. Tek. dan Ilmu Komput.*, vol. 07, no. 26, pp. 135–148, 2018.

[4] H. F. Siregar *et al.*, "PROTOYPE GERBANG LOGIKA (AND, OR, NOT, NAND, NOR) PADA LABORATORIUM ELEKTRONIKA STMIK ROYAL KISARAN," *J. Teknol. Inf.*, vol. 1, pp. 42–52, 2017.

[5] H. F. Siregar and M. D. Irawan, "Model Dan Simulasi Perbandingan Prototype Rangkaian Digital Half Adder Alu Standar Dengan Inovasi Rangkaian Digital Half Adder Alu," *J. Teknol. Inf.*, vol. 2, no. 1, p. 1, 2018, doi: 10.36294/jurti.v2i1.414.

[6] H. F. Siregar and M. D. Irawan, "Model Dan Simulasi Prototype Rangkaian Digital Konversi Gerbang AND, OR, NOT Menjadi Gerbang NAND Dan NOR," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 4, no. 1, pp. 161–166, 2019, doi: 10.30743/infotekjar.v4i1.1689.

[7] F. P. Marsyaly, "Pembelajaran Gerbang Logika Dasar Berbantuan Mobile Di Sekolah Menengah Kejuruan," *J. Edukasi Elektro*, vol. 1, no. 1, pp. 1–10, 2017, doi: 10.21831/jee.v1i1.13257.

[8] G. N. Tasse, S. James, and B. Rosman, "A Boolean task algebra for reinforcement learning," *Adv. Neural Inf. Process. Syst.*, vol. 2020-Decem, no. 1, pp. 27–34, 2020.

[9] R. Hastri, Faisal, V. Eminita, and F. Liani, "Analisis hubungan antara kecerdasan logis matematika, niai matematika sekolah dan nilai pada materi aljabar boolean mahasiswa pada program studi pendidikan teknologi informatika berdasarkan tingkat pendidikan," *Pap. Knowl. . Towar. a Media Hist. Doc.*, vol. 7, no. 2, pp. 143–154, 2014, [Online]. Available: jurnal.umj.ac.id/index.php/fbc

[10] M. Šeda, "Heuristic Set-Covering-Based Postprocessing for Improving the Quine-McCluskey Method," pp. 256–260, 2007.

[11] S. P. Tomaszewski, I. U. Celik, and G. E. Antoniou, "WWW-based Boolean function minimization," *Int. J. Appl. Math. Comput. Sci.*, vol. 13, no. 4, pp. 577–583, 2003.

[12] M. Fitria and F. Faisal, "Penggunaan Aljabar Boolean Dalam Menganalisis Kegagalan Pada Fault Tree Analysis," *J. Mat. Murni Dan Terap. Epsil.*, vol. 3, no. 2, pp. 27–38, 2009.