International Journal of Electrical and Computer Engineering (IJECE)
Vol. 8, No. 5, October 2018, pp. 3550~3559
ISSN: 2088-8708, DOI: 10.11591/ijece.v8i5.pp3550-3559

Design of Digital Parity Generator Layout using 0.7 micron Technology

Wahab Musa[1], Sri Wahyuni Dali[2], Ade Irawaty Tolago[3]
[1,3]Department of Electrical Engineering, State University of Gorontalo, Indonesia
[2]Department of Telecommunication Engineering, State University of Gorontalo, Indonesia

Article Info

Article history:

Received Feb 21, 2018
Revised May 29, 2018
Accepted Jun 6, 2018

Keyword:

Layout design
Parity generator
Technology process

ABSTRACT

The proposed digital parity generator circuit is an integrated circuit functions to detect data errors at the transmitter end, and check it at the receiving end. In digital communications, the digital messages are transmitted in the form of 1's and 0's between two points. It is an error free if both are the same. The purpose of this research is to implement a design method of digital parity generator layout with 0.7 micron process technology ECPD07 from Tanner Tools. Layout design starts from making schematic circuit, test function and make a layout. Next, check the layout results in terms of design rules and verify the desired functionality gradually. The results show that the circuit has functioned well as an odd parity generator. The simulation results obtained with loads CL = 25 fF, tpLH = 2nS and tpHL = 1.46 nS indicate that tp = 1.73nS or operating frequency of 578 MHz. The integrated digital parity generator circuit using transmission gate has a size of 14758 um2 (78.5 um x188 um), consisting of 74 gates.

Corresponding Author:

Wahab Musa,
Department of Electrical Engineering,
State University of Gorontalo,
Jln. Jenderal Sudirman No. 06 Kota Gorontalo-96128, Indonesia.
Email: wmusa@ung.ac.id

1. INTRODUCTION

In everyday life, we cannot be separated from the use of fixed and mobile telephone, sending and receiving data both at close and long distance, and various other types of communication. Utilization of IT in communication has spread to all areas of human life in the current era of information technology advances.

Information sent over long distances has gone through various media that can naturally reduce the quality of information. One important factor in data and voice communications is the quality of information transmitted and arriving at the recipient. Ideally the information received is exactly the same as the information sent. In reality, however, this naturally occurring disorder cannot be eliminated at all but can be minimized. While the disturbance caused by human error may be suppressed or eliminated altogether.

The use of Parity Generator is one way to minimize the errors of digital data on transmitted information by detecting unwanted digital data. There are two digital data formats used in the digital communication, data format with odd number of parity and data format with an even number of parity [1]. In a communication that uses the odd parity data format, all existing data is odd, so that if any of the data appears to have even parity, then this indicates a data error occurs. If this happens, then the recipient can recover the data that has the error either by repairing itself or by returning the data to the sender.

To design a circuit layout to be implemented on an IC (Integrated Circuit), a tool is needed to realize the desired design. The layout of ICs is bound by a set of design rules [2], [7]. One of the tools for IC layout used for this Parity Generator is Tanner Tools that uses 0.7 micron technology. With 0.7 micron technology,

it is expected that space on CHIP IC can be saved for other designs. The space is often considered as a critical parameter in maximized the operating speed [10].

2. DESIGN CONCEPT

In general, there are two known concepts in the process of designing electronic circuits, namely top-down design and bottom-up design. In this concept the term top refers to the highest level while the bottom refers to the lowest level [3].

In top-down design, the designer only knows the function and specification of the circuit. The designer then partitions the circuit into blocks at a lower level. Then each block is partitioned back into blocks at a lower level. The process continues until the primitives are obtained which need not be further partitioned.

A partitioning process in top-down design is influenced by design criteria, such as cost, speed, chip area and so on. Typically, an increase in the value of a criterion will decrease the value of other criteria so that consideration of these criteria will result in a non-unique design. Partitioning process does not depend on whether or not there is a primitive at the lowest level.

At the bottom-up design, the design process also starts from the definition of the circuit at the highest level. However, the partitioning process in the bottom-up design is determined by primitive availability. The primitive or block must have been created on another occasion and may be created by another designer. The designer is forced to use the primitives or blocks that are already available when it will partition the circuit to a lower level [3].

Most designers combine both design techniques because they utilize their respective advantages. The design process is also customized with the implementation of the design. The concept of top-down design is more ideal than bottom-up design. However, this design process will require many components that are not standard, so it can increase the cost of designing. Bottom-up design techniques are considered more economical, but the performance of the design obtained is not as well as the top-down design.

2.1. Design phase

Stage 1: System specification

In designing, the first thing that needs to be determined is the system specification to be made. This specification will be a representation of the system at the top level. Things to consider are the performance, function, and system dimensions to be created. It also considered the fabrication technology and design techniques.

Stage 2: Functional design

At this stage the designer considers the behavioral aspects of the system. Usually the system is divided into sub-units to reduce the complexity. The result of this stage is usually a timing diagram or other relationship between sub-units [4], [10].

Stage 3: Logic design

At this stage, the logical structure representing the functional design is made. Usually the logic design is represented in boolean expressions.

Stage 4: Circuit design

The purpose of this design stage is to obtain a circuit representation of the logical design. Boolean expressions are converted into circuit representations taking into account the electrical characteristics of each component. The design of this circuit is expressed in the form of circuit diagrams.

Stage 5: Physical design

At this stage, the circuit representation of each component is transformed into a geometry representation. This representation is a set of geometric patterns that show the logical functions of each component. This geometry representation is called a layout. The layout design process also depends on the design rule, which is a design guide based on the limitations of the fabrication process and the electrical properties of the fabrication material [8].

Step 6: Verify design

At this stage the Layout is checked to ensure that the layout meets the specifications and fabrication rules. Design verification includes checking design rules and checking the suitability between layout and circuit representation.

Stage 7: Fabrication

After verification, the layout is ready to be fabricated. Fabrication includes several process steps. Before the chip is mass-produced, a complete design is created and tested [7].

Stage 8: Packaging, testing and debugging

Eventually, each chip is packaged and then tested to ensure that the chip meets all the design specifications and is working properly.

2.2. Alternative designs

With the growing market demand for reliable, low cost and fast time-to-market IC products, various design and implementation alternatives emerged in response to these demands. The implementation and design alternatives are classified into full custom and semi custom (standard cell, gate array, field programmable gate array and sea of gates) [5].

2.2.1. Full custom design

The full custom design process is very complex and takes more time than any other alternative. So this design is only suitable for designs that require high performance that cannot be achieved through other approaches. The full custom design process is a considerable to solve problem in the engineering which are computational intensive [12].

In the full-custom design, the circuit is partitioned into blocks. The size, placement, and design of each block are free, but the design results must be solid and compact. Routing of interconnection lines using unused block areas. Commonly used multiple metal layers for interconnection lines [6], [9].

2.2.2. Semi custom design

With semi-custom design, the designer can use existing cells and utilize the cells as needed so as to obtain the desired function.

3. RESULTS AND ANALYSIS

Proposed Parity generator designed is 8-bit which gets input from the shift register. The design can be realized with only two basic blocks: a D-Flip-Flop (DFF) and an XOR gate. A N-bit parity generator is a block that produces a combination of Boolean functions that have N-parallel input and one output.

If parity is odd, then the output bit is:

a. Logic "1" if the number of bits-1 in the input vector is even
b. Logic "0" if the number of bits-1 in the input vector is odd

For example, in Table 1, we can see the condition of the output data on a 4-bit parity generator circuit with various possible combinations of input data.

Table 1. The truth Table of the 4-bit Parity Generator

D_0	D_1	D_2	D_3	Q
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

For even parity, the output is the complement of the odd parity output. Although it looks a bit complicated, it can be realized easily using the XOR gate. The parity function can be written in Equation (1):

$$P = D_0 \; xor \; D_1 \; xor \; D_2 \; xor \; D_3 \; xor \; \ldots \quad (1)$$

The 4-bit odd parity generator is expressed by (2).

$$P = ((D_0 \text{ xor } D_1) \text{ xor } (D_2 \text{ xor } D_3)) \quad (2)$$

Where,

P: Parity, D_n: Data bit n

The parity function is used to verify the correctness of data transmitted in digital communications. In some communication protocols, data bits are sent together with corresponding parity values. The receiver checks the parity. If one of the bits cannot be decoded during transmission, then parity will not match, and the recipient will request the correct transmission of data repeated from the sender.

The retransmission method has a disadvantage, that is, when the communication channel is interrupted. However, parity checking on the retransmission method is very simple and is still a popular method of data communication.

4. SHIFT REGISTE

If it is assumed that the receiver receives the data in a serial form, and generates the parity bit continuously for the last 8-bit input data. To solve it, the incoming serial data stream is converted to parallel (vector) as the parity generator input. For this purpose, we will use the shift register.

A N-bit shift register has N D-Flip-Flop (DFF) connected cascade. At each clock cycle, the input of one DFF is transferred to the next DFF input in the register. If input of each DFF is provided, this structure can use serial input-parallel output (SIPO).

5. IMPLEMENTATION

To implement the layout design of 8-bit parity generator circuit, this research used Tanner Tools aids software which is a collection of some software. Programs incorporated into Tanner Tools are S-Edit version 1, L-Edit version 6, NetTran version 1.00, GateSim version 2.00 and LVS version 2.19.

The design using Tanner Tools includes several stages, starting from schematic design, functional simulation stage, then layout design and the last is verification layout.

5.1. Schematic circuit design

The circuit to be created must be represented in the form of a schematic circuit. In the schematic circuit design, the components used in the schematic circuit are limited by the cells available in the cell library. The schematic circuit design used graphic editor in the schematic editor program. The schematic editor available on Tanner Tools is S-Edit. This program requires circuit level cell library files. This file contains modules that are representations of standard cells in the cell library for layouts. The circuit design is created by compiling the cell modules [6].

The circuit diagram of a parity generator that uses an 8-bit shift register as shown in Figure 1.

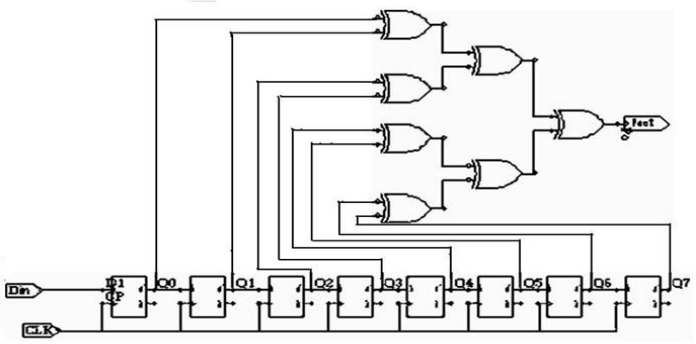

Figure 1. 8 bit parity generator circuit

5.2. Designing mask layout

Designing mask layout consists of several stages, namely partitioning, floor plan, placement, routing, and compaction. However, for the design of simple circuits, there is a design stage mask layout that

can be ignored. Design stage mask layout done in this research include floor plan, placement and routing and verification.

Based on the parity generator circuit in Figure 1, the circuit consists of 7 XOR-2 input gates and 8 D-Flip-Flops. Then the two groups of components will be used as basic cells as shown in Figure 2.

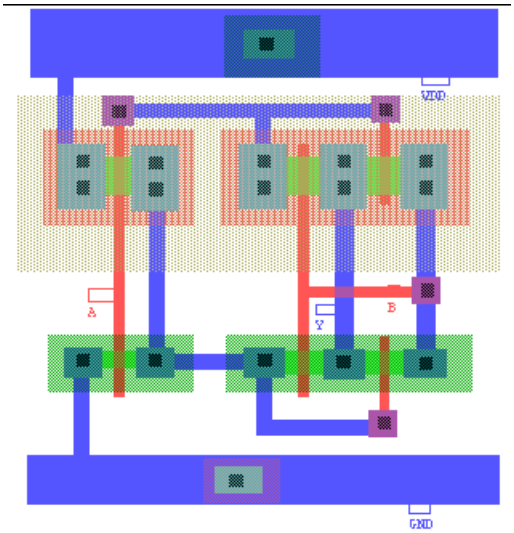

Figure 2. Layout of Basic Cell XOR 2 input

6. RESULTS AND ANALYSIS

The circuit design result created by S-Edit must be extracted to get the Netlist of the circuit. Meanwhile, from the Databook should be created a macro file that contains information about the logical functions of each standard cell and its delay value.

The logic function used should be adjusted to the logical functions that can be analyzed by the simulator. Both the extraction file and the macro file will be used by the NetTran program to create a new file containing the network of logic functions along with its delay which can be simulated by GateSim. Verification of results can be done in each module or as a whole series.

Figure 3 shows the basic cell simulation results of XOR 2 inputs. When the inputs (A, B) get the logic "1" marked by the blue line and the red line, the output Y (yellow line) is a logic "0", and when one input is "0" the output Y directly up to logic "1" but there is a delay of 0.7 nanoseconds.

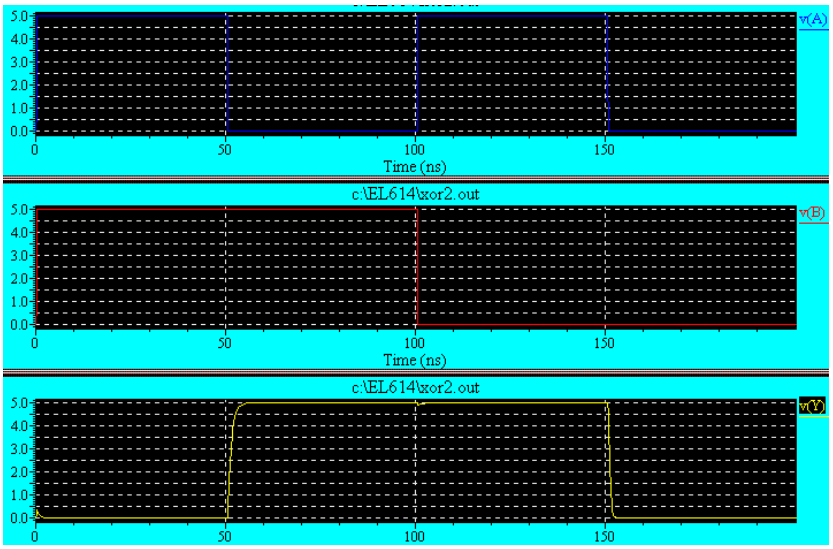

Figure 3. Basic cell XOR2 input results with spice simulation

The delay time that occurs after the output [Y] of XOR2 with load capacitor CL = 25fF is tpLH of 0.726 nS, tpHL of 0.5 nS. Delay time this happens should be taken into account because of a complete series that uses basic cell in large quantities, causing delay is also greater and this gives rise to errors when data is sent to the recipient.

6.1. Parity check circuit layout

Check the even parity using 7 pieces of XOR as in Figure 4. To find out whether the circuit is working, then the logic input pattern is given as Table 2.

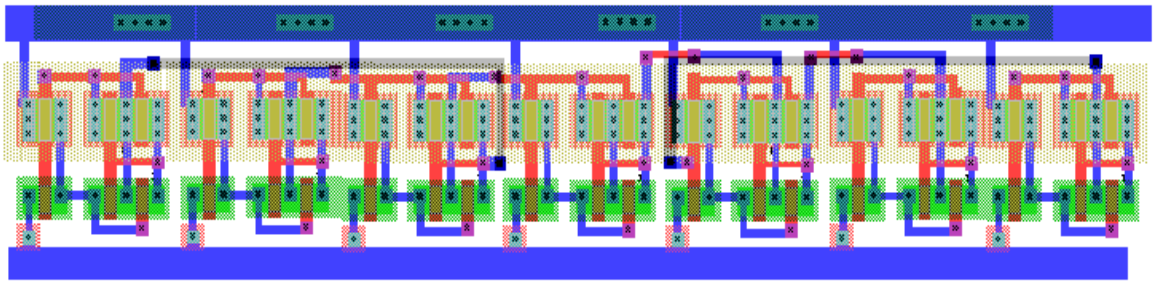

Figure 4. Layout of parity check

The Parity Check layout consists of 7 XOR 2 inputs using basic cell as shown in Figure 2. To verify the parity check circuit, a truth table as shown in Table 2 is used.

Table 2. Parity Checks Logic Circuit Pattern

Data Input (Parity)								Output
A	B	C	D	E	F	G	H	Y
1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	1
1	1	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	1
1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	0	0	1
1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1

The results obtained from testing with logical test patterns such as Table 2 are shown in Figure 5.

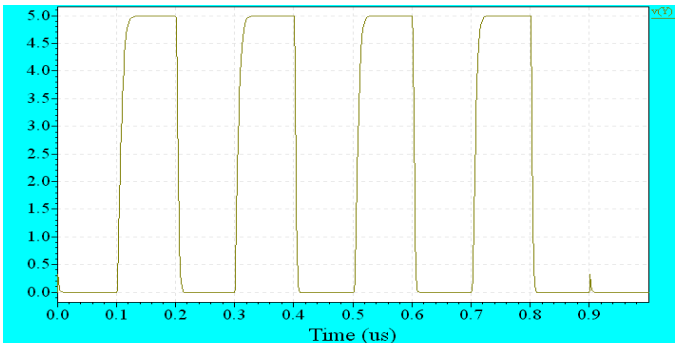

Figure 5. Spice Simulation Results of parity check circuit (odd)

Figure 5 illustrates that the logic state in the (Y) output of the parity check circuit will change from logic "0" to logic "1" when it gets logical input with even and odd parity bit numbers. It generates an output of 0 if the number of 1's in the input sequence is even and 1 if the number of 1's in the input sequence is odd.

6.2. Shift register circuit layout

The shift register circuit shifts the serial input data with 8-bit data shift. Basic cell used is formed from the transmission gate to obtain a simple circuit that serves as a shift register. The transmission gate is used to reduce the complexity of the circuit and to maximize the operating speed [10] as shown in Figure 6.

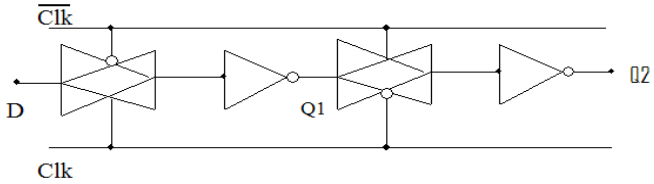

Figure 6. The 2-bit shift register circuit is formed from the transmission gate

The basic cell layout for the 1 bit shift register is shown in Figure 7. This basic cell has been tested for good performance and the result of a spice simulation of this cell is shown in Figure 8. The amount of tpLH and tpHL is measured by loading C Gate of XOR, each output bit of the shift register will be connected with XOR to detect its parity. To form an 8-bit shift register, the basic cell layout as in Figure 7 is used. The length of the shift register is limited to reduce the require more circuitry which is causing more delays [11]. The convenience to combine the 8 basic basic cells is determined by the standard cell created, so that VDD and GND and Clock can be connected easily. The result of spice simulation of SR-1bit is shown in Figure 8.

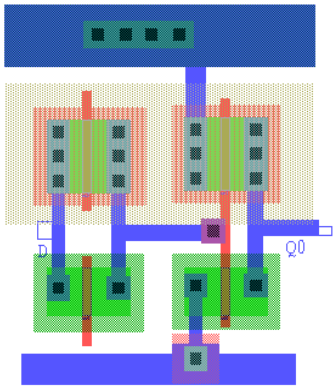

Figure 7. 1-bit shift register circuit layout

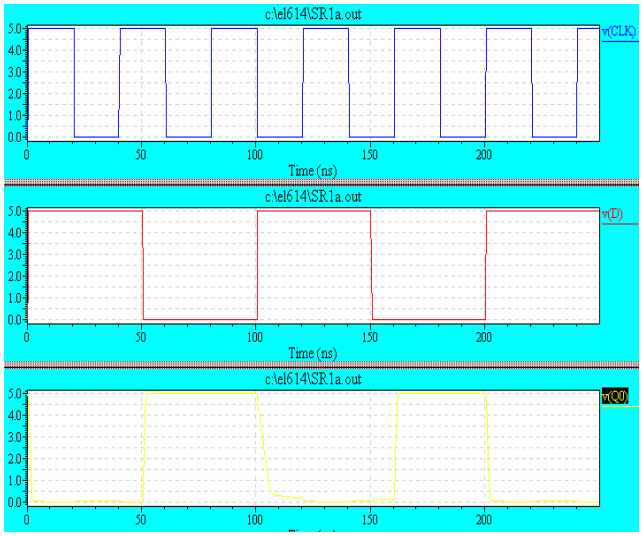

Figure 8. Spice simulation of SR-1

To obtain 8-bit Shift Register (SR 8), the basic cell of SR-1 is combined with 8 units. The result of this incorporation as shown in Figure 9.

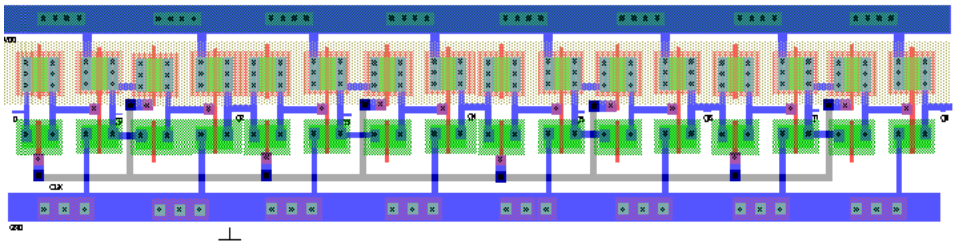

Figure 9. 8 bit shift register layout

Before the 8-bit shift register is combined with the Parity Check circuit, the function testing is performed first. The output of each bit of the shift register is determined by the serial input data and the delay that occurs at each level.

It is rather difficult to obtain an output that can actually represent the desired data, since the output that occurs at each level depends on the condition of the data obtained from the previous level output. If the incoming clock is not correct with the output data changes from the previous level, then the data obtained will also show an error. Figure 10 shows the complete layout of 8 bit parity generator.

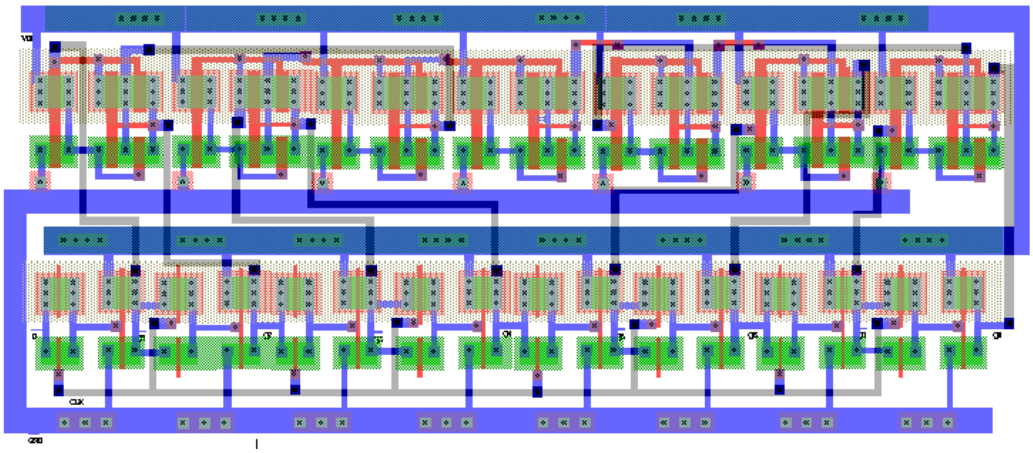

Figure 10. Complete layout of 8 bit parity generator

Table 3 shows the result of 8 bit parity generator simulation. The output and input relationships of the parity generator, i.e. when the input parity check circuit (output Q7, Q6, Q5, Q4, Q3, Q2, and Q1 on the shift register) changes the value due to serial and clock data.

Table 3. Relationship of Data Input on Shift Register and Output Parity

t (µS)	SR output	Logic Level Y(V)	Parity Check Output
0.0 - 2.0	10101011	0.0	0
2.0 - 2.2	10101001	5.0	1
2.2 - 2.4	10101101	0.0	0
2.4 - 2.65	10100101	5.0	1
2.65 - 2.9	10110101	0.0	0
2.9 - 3.1	10010101	5.0	1
3.1 - 3.4	11010101	0.0	0
3.4 - 3.6	01010101	5.0	1

6.3. Comparative study

For the purpose of comparing proposed layout design with other research results, then some of the following parameters are used. Comparison was made to the results of the study in [13] with the parameters of the size of the layout, the time delay, the number of gates, the number of cells, the existence of the shift register, and the number of bits. Nevertheless, the difference in technology used causes less fair yield due to different specifications between CMOS technology and QCA. A striking difference is the use of cells in the QCA and MOSFET in CMOS on the proposed design. This technological difference causes considerable deviations from the parameters compared.

QCA cells are very small and the systems based on QCA should be very low power, because there is no current flowing [13], [14]. However, the drawback of QCA based structure is suffering from various types of faults, such as displacement, misalignment, and omission fault. The defective quantum cell is displaced from its original position, not properly aligned and sometime the defective cell is missing.

Table 4 gives the comparison between the proposed circuit layout performance with previous designs [13].

Table 4. Comparative Study of Proposed Design with other Researches

	Technology		Area	delay	No. of gates	No. of cells	S-R	Bits
As in [13]	QCA	PG	52488 nm^2	2	2	60	NA	3
		PC	135432 nm^2	2.25	3	117	NA	4
Proposed	CMOS	PC	14758 um^2	3.46	7	NA	8-bit	8

* QCA indicates Quantum-dot Cellular Automata, PG indicate Parity Generator and PC indicate Parity Checker, S-R indicate Shift Register, CMOS indicate Complementary Metal Oxide Semiconductor

In [13], a parity bit is used for detecting errors during transmission. Including parity bit, message transmitted and checked at the receiving end for errors. The design layout that checks the parity at the receiver side is called a parity checker (PC) and the design layout that generates a parity bit at the transmission side is called a parity generator (PG). In the proposed design, an eight bit serial message from SR is forwarded as the 8-bit parallel message to PG with an even parity bit, as shown in Table 3. The proposed layout of even parity generator is shown in Figure 10, which is implemented by 7 numbers of two XOR gates and 8 numbers of SR. The circuit area is 14758 um^2.

7. CONCLUSION

From the test results data by simulation, it appears that the 8-bit Parity Generator circuit is functioning properly, that is, it has "0" output when the number of bits 1 in input data is odd, and has an output "1" when the number of bits 1 of the input data is even. At the output of Y occurs tpLH = 2 nS and tpHL = 1.46 nS.

In testing the function of the parity generator, the output signal is determined by the output of the shift register, so that the serial input data must be such that the output of the shift register can generate test signals representing all possible combinations.

To facilitate the testing, then this design is done gradual testing, that is the first test function of the parity check which consists of seven pieces XOR-2 input and then followed by testing the function of 8-bit shift registers.

In the layout, XOR two inputs and 1-bit shift registers are made as standard-cell so they are easy to combine with each other. Global routing like VDD, GND and Clock are made as easy as possible to connect.

The output of each level of the 8-bit shift register is determined by the input data obtained from the previous level and the clock state upon receiving the input data. An error can occur if the input data arrives out of sync with the clock. Errors occur, for example, because single-stuck so that the output Y only logic "0" or "1" and does not change even though the trigger has occurred repeatedly.

Layout of 8-bit Parity Generator 14758 um2 (78.5 um x188 um), consists of 74 gates. Its size is quite small because it uses transmission gate so the circuit is quite simple. This circuit has functioned as an odd parity generator, seen from the simulation results obtained with loads CL = 25 fF, tpLH = 2nS and tpHL = 1.46 nS so tp = 1.73nS, or operating frequency of 578 MHz.

For a simple circuit, the size of proposed layout is still large enough, because the design is more focused on the circuit function. It is expected the use of efficient space and component for the next design.

ACKNOWLEDGEMENTS

Our gratitude goes to the Institute for Research and Community Service (LPPM), State University of Gorontalo, which has provided research funding through the DIKS-UNG 2016 scheme. Also to the members of the research team who have contributed to the implementation of this research.

REFERENCES

[1] Naveed Sherwani, "Algorithms for VLSI Physical Design Automation", Kluwer Academic Publisher, 1992.
[2] G. A. Allan and A. J. Walton, "An Yield Improvement Technique for IC Layout Using Local Design Rules", *IEEE Transactions on Computer-Aided Design*, vol. 11, no. 11, pp. 1355-1362, Nov 1992.
[3] John P. Uyemura, "Physical Design of CMOS Integrated Circuits Using L-EDIT TM," 1995.
[4] --, "ES2 ECPD07 Library Databook", Tanner Tools, 1994.
[5] W. E. Sulistiono, "Perancangan Mask Layout Parallel Access Circuit Menggunakan Standard Cell", Master Thesis, Institut Teknologi Bandung, 1998.
[6] R. Martins *et al*., "Multi-Objective Multi-Constraint Routing of Analog ICs Using a Modified NSGA-II Approach", *International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuits Design (SMACD)*, 2012.
[7] G. A. Allan, "Targetted Layout Modifications for Semiconductor Yield/Reliability Enhancement", *IEEE Transactions on Semiconductor Manufacturing*, vol. 17, no. 4, pp. 573-581, Nov 2004.
[8] C. L. Valenzuela and P. Y. Wang, "VLSI Placement and Area Optimization Using a Genetic Algorithm to Breed Normalized Postfix Expressions", *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 4, pp. 390-401, Aug 2002.
[9] H. Mahmoodi et al., "Full Custom IC Design Flow Tutorial using Synopsys Custom Tools", Nano-Electronics & Computing Research Lab School of Engineering, San Francisco State University, 2014.
[10] A. Boudaoud *et al*., " VHDL Design and FPGA Implementation of a High Data Rate Turbo Decoder based on Majority Logic Codes", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 4, pp. 1824-1832, August 2017.
[11] Zulfikar *et al*., "FPGA - based Design System for a Two-segment Fibonacci LFSR Random Number Generator", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 4, pp. 1882-1891, August 2017.
[12] G. Singh & N.R. Prakash, "FPGA Implementation of Higher Order FIR Filter", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 4, pp. 1874-1881, August 2017.
[13] S. Santara and U. Roy, "Design and Optimization of Parity Generator and Parity Checker Based On Quantum-dot Cellular Automata", *World Academy of Science, Engineering and Technology, International Journal of Nuclear and Quantum Engineering*, vol. 8, no. 3, pp. 491-497, 2014.
[14] M. Mustafa and M. R. Beigh, "Design and Implementation of Quantum Cellular Automata Based Novel Parity Generator and Checker Circuits with Minimum Complexity and Cell Count", *Indian Journal of Pure & Applied Physics*, vol. 51 pp. 60-66, 2013

BIOGRAPHIES OF AUTHORS

Wahab Musa, who completed his undergraduate education at Hasanuddin University in 1981, then completed his Master's degree in Microelectronics at the Bandung Institute of Technology in 2001 and Ph.D in Computer Sciences at Universiti Utara Malaysia in 2013. His current interests include Digital Design, Digital Signal Processing, and Artificial intelligence. Several articles in this area of interest have been published.

Sri Wahyuni Dali, who completed her undergraduate education at Hasanuddin University in 1997, then completed his Master's degree in Telecomunication at Gadjah Mada University in 2005. Her current interests include Digital Communication, and Software Engineering.

Ade Irawaty Tolago, who completed her undergraduate education at Muslim Indonesia University in 1998, then completed his Master's degree in Information System at Hasanuddin University in 2010. Her current interests include Information System, and Power Engineering.