A NOVEL METHODOLOGY TO IMPROVE POWER CONSUMPTION FOR ADDERS

Le Thanh Toi[1]*, Truong Tri Lac[2]

[1]*Ho Chi Minh City University of Industry and Trade*

[2]*Uniquify Viet Nam One Member LLC*

*Email: *toilt@huit.edu.vn*

Received: 24 December 2024; Revised: 3 April 2025; Accepted: 9 April 2025

ABSTRACT

Adders are one of the important components of forming a computational system. Optimizing power consumption, speed and delay of adders has contributed to the economical and efficient use of energy. This paper proposes a general flow to implement the combinational logic circuits using Null Convention Logic (NCL) for asynchronous circuits. Carry Look Ahead (CLA) adder and Ripple Carry Adder (RCA) are chosen in order to illustrate the proposed flow. These adders are implemented by DC tool using conventional cell libraries. In addition, we also make the comparison of the implemented results of adders above by asynchronous and synchronous techniques. The synthesis results indicate that the power of the NCL designs decreases by 62.88% (RCA), and 75.09% (CLA) compared to the Boolean logic combinational designs.

Keywords: RCA, CLA Adder, Null Convention Logic, NCL combinational circuit, asynchronous method.

1. INTRODUCTION

The addition is a basic operation of the arithmetic operations that executes the addition of binary numbers. It keeps a vital position in arithmetic systems. It is widely used not only for executing algorithms and transforms in digital signal processing systems [1], but also for other functions [2]. In recent years, there have been studies on adders such as a high-speed energy efficient carry skip adder [3], an energy and area efficient CSA [4], different basic adder topologies [5] and several related works found in [6-13]. Most of the adders mentioned above are designed by the synchronous technique and synthesized using conventional cell libraries. Besides, there are many adders designed by basing on NCL, and synthesized by commercial tools [14, 15, 16].

However, each study also has its limitations. In [14], the authors improved the area of the adder thanks to the common output evaluation method. However, there is no flow proposed in [14] to help researchers who would like to go on with inheritance and development. In [15], NCL RCAs have been implemented and compared with the corresponding RCAs using the normally synchronous method. These adders are implemented in Cadence Virtuoso, so synthesis results of power, area, and delay are difficult to achieve at the most optimal level. In [16], the NCL exponent adder was designed with the flow implemented at both transistor and RTL levels. However, the flow mentioned in [16] does not express the test step on the input Completeness and Observability which is important in the NCL combinational logic design flow. Hence, this work proposes a general scheme for designing logic combinational circuits by a novel method based on NCL. Besides, adders, including RCAs and CLA adders are implemented based on the proposed flow mentioned above.

The NCL based models are delay-insensitive logic models used for asynchronous circuit designs and particularly for very-large-scale and application specific integrated circuit designs [17]. This technique has interested the researchers and it is also utilized for many different goals like the synthesis of quasi-delay insensitive (QDI) combinational circuits using NCL gates [18], studies of the types NCL gates [19], high-performance and low-power mobile digital systems [20], bus choices for designing asynchronous circuits [21], AES algorithm[22], low power circuit designs [23-25], and some

relevant studies can be found in [26, 27].

Besides, threshold gates, which are usually utilized in the semi-custom design of the QDI circuits are not in the commercial standard cell libraries. Designers often synthesize the NCL-based asynchronous designs by using the conventional cell library. In this paper, the selected adders are implemented by the two methods and are also synthesized by using the conventional cell library. The synthesis results of the proposed method are compared with the synthesis results of the synchronous method.

The remainder of the paper consists of three parts. Part 2 introduces the design scheme for NCL based combinational logic circuits and the summary of NCL. Subsequently, Part 3 gives the simulation and synthesis results. Finally, conclusions are presented in Part 4.

2. RESEARCH METHOD

2.1 Null Convention Logic

NCL is not only a QDI logic style but also a symbolically complete logic style. NCL is a novel logic design technique that does not utilize control signals and uses for asynchronous circuits [28]. The NCL-based asynchronous models always works correctly despite the latency in the transmission [29]. In order to reach the objectivs mentioned above, NCL circuits employ dual-rail logic [29]. A traditional logic signal has only one rail while an NCL signal A is formed by two wires A0 and A1. Table 1 illustrates a traditional logic signal convertwd into a dual-rail signal [14]. Because A1 and A0 are mutually exclusive, so they cannot be confirmed simultaneously.

Table 1. NCL signal

		code	
traditional logic	NCL logic	A1	A0
0	DATA1	1	0
1	DATA0	0	1
	ILLEGAL	1	1
	NULL	0	0

NCL circuits are formed from a set of twenty seven NCL gates [14]. Basically, ThxyWk1k2 is a symbol for the NCL gate, where y is the total number of inputs, W is the weight of the inputs with values k1, k2 and x is the threshold value that means at least x of y inputs must be confirmed before the output is confirmed. Figure 1 shows a fundamental NCL gate.

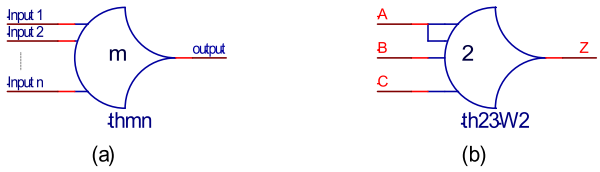

Figure 1. NCL gates

NCL combinational circuits created from a set of twenty seven NCL gates have tasks similarly to normal logic circuits. NCL circuits always uphold the two attributes of observation and input completion.

Input completion forces all NCL circuit outputs not to be converted NULL into DATA until all inputs are confirmed DATA and all NCL circuit outputs cannot be converted DATA into NULL until all inputs are confirmed NULL. For circuits with multiple outputs, it is acceptable to comply with Seitz's standard for latency-insensitive signal. For some outputs, they are possible to change without a complete set of inputs, as long as all outputs cannot be transitioned in advance all inputs arrive.

Observation requires that no 'orphan' can pass through a component. An 'orphan' is defined as a wire that transitions during the current DATA wavefront but is not used to determine the output. The

observability condition is also known as stability, to ensure that every transition at the gate is observable at the output, and is needed to transition at least one of the outputs.

2.2 Methodology to design NCL combinational circuits

The design flow of the NCL combinational circuits shown in Figure 2 and the algorithm for this flow is shown in Figure 3. This design flow which consists of six steps is similar to the design flow of the Boolean combinational logic circuits.

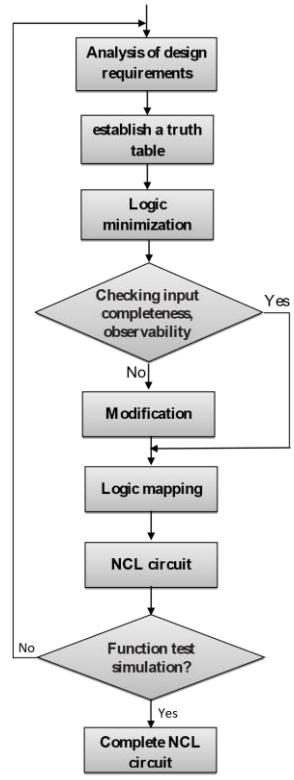

Figure 2. NCL combinational circuit design flow chart

Algorithm 1 NCL combinational circuit algorithm

```
1:  Input: Design requirements.
2:  Analysis the design requirements.
3:  Establish the formulation of design.
4:  Minimize logic of design
5:      if design is suitable for the design requirements then
6:          Technology mapping
7:          Characterize design
8:        if design pass the function verification then
9:          Print output
10:     else
11:         Modify logic of design
12:     end if
13: Complete NCL circuit
```

Figure 3. NCL combinational circuit algorithm

Step 1: Analysis of design requirements

From the design requirements, the designer must conduct an analysis to determine which elements are input variables, which factors are output functions, and their logical relationship. To analyze them properly, it is necessary to understand and consider deeply the design requirements, which is a difficult and important task in the design.

Step 2: Establish a truth table

First, tabulate to describe the corresponding functional relationship between the state of the input variables and the state of the output functions. It is a table of the logic function request enumeration or a function table. Next, substitute the logical values for the states, i.e use the binary numbers 0 and 1 to represent the respective states of the input and output. As a result, we have a logical table of values, also known as a truth table.

Step 3: Logic minimization

Similar to the Boolean combinational logic circuit, if the number of input variables is small, we can reduce the logic expression of the outputs by the direct reduction method or the usual reduction rules. If the number of input variables is large then the Karnaugh cover (K), other reduction techniques can be applied to determine output expressions in the form of the most simplified Sum of Products (SOP) for each output. However, for NCL functions, SOP expressions are performed for both output functions corresponding to rail1 and rail0. Note that states 0 in cover K represent the logical state of the rail0, and states 1 in cover K represent state 1 of the rail1, the logic function reduction for rail0 is done the same as a reduction method for rail1.

Step 4: Checking input completeness and observability

This step is not included in the design process of Boolean logic combinational circuits. After the output expressions are reduced, the evaluation must be performed to ensure that the circuit is input complete and observable, because NCL combinational circuits must satisfy both the input completion criteria and the observability criteria [9]. If the circuit does not meet the input completion criteria, a correction must be made by finding any inputs that are not present in the product terms to consider and add the appropriate product terms.

Step 5: Logic mapping

The output logic functions could be split into groups of four variables or fewer to be mapped to the 27 NCL gates, while ensuring that the resulting circuit is observable. To minimize area and delay, it is necessary to divide the output equations so that the minimum number of groups is obtained, which happens when the number of product terms in each group is maximized.

Step 6: Function test simulation

After mapping NCL output expressions into NCL threshold gates, NCL circuits are simulated to test their functions. If the simulation results are correct, the design process will finish. Otherwise, the design process will go back to the step of the analysis of design requirements.

2.3 Case study

In this section, 4-bit RCA and CLA are used as case studies to illustrate our flow. These adders are converted from boolean logic to Null convention logic.

2.3.1. Ripple Carry Adder

RCA is made up of 1-bit full adders. This sub-section will cover the conversions of a 1-bit synchronous adder to a 1-bit NCL based asynchronous full adder.

Step 1: Determine the inputs and outputs

Table 2. Full adder ports description

Signal name	Input/Output	Width (bits)
X	Input	1
Y	Input	1
Cin	Input	1
Sum	Output	1
Cout	Output	1

Step 2: Establish a truth table

The truth table shows the logical operators between inputs and outputs.

Table 3. Full adder's truth table

X	Y	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Step 3, 4: Logic optimization and input completeness, observability check

Since the number of inputs is three, the Karnaugh map was applied.

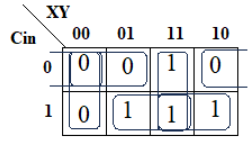

Figure 4. Karnaugh map for Cout signal

Based on the Karnaugh map in Figure 4, the equations of Cout (carry out) are shown in (1), and (2).

$$Cout^0 = X^0Y^0 + Cin^0X^0 + Cin^0Y^0 \quad (1)$$

$$Cout^1 = X^1Y^1 + Cin^1X^1 + Cin^1Y^1 \quad (2)$$

Because Cout is unsatisfied with all three inputs' completeness, the Sum signal must be input complete with all inputs X, Y, and cin shown in equations (3), and (4).

$$Sum^0 = X^0Y^0Cin^0 + X^0Y^1Cin^1 + X^1Y^0Cin^1 + X^1Y^1Cin^0 \quad (3)$$

$$Sum^1 = X^0Y^0Cin^1 + X^0Y^1Cin^0 + X^1Y^0Cin^0 + X^1Y^1Cin^1 \quad (4)$$

In terms of observability, the Sum and Cout signals are satisfied. The Cout signals contain X^0Y^0, Cin^0X^0, Cin^0Y^0, X^1Y^1, Cin^1X^1, and Cin^1Y^1 which are also included in (3), and (4). Thus, the Cout signal is considered as the forth input of the Sum. Based on the Karnaugh map shown in Figure 5, (3) and (4) are simplified to (5) and (6), respectively.

$$Cum^0 = Cout^1X^0 + Cout^1Y^0 + Cout^1Cin^0 + X^0Y^0Cin^0 \quad (5)$$

$$Sum^1 = Cout^0X^1 + Cout^0Y^1 + Cout^0Cin^1 + X^1Y^1Cin^1 \quad (6)$$

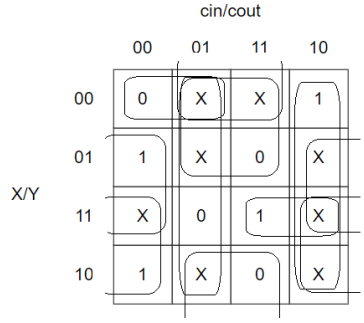

Figure 5. Karnaugh map for Sum signal

Step 5: Logic mapping

In this step, 27 NCL threshold gates [17] are mapped to (1), (2), (5), and (6) which result in the final full adder shown in Figure 6.

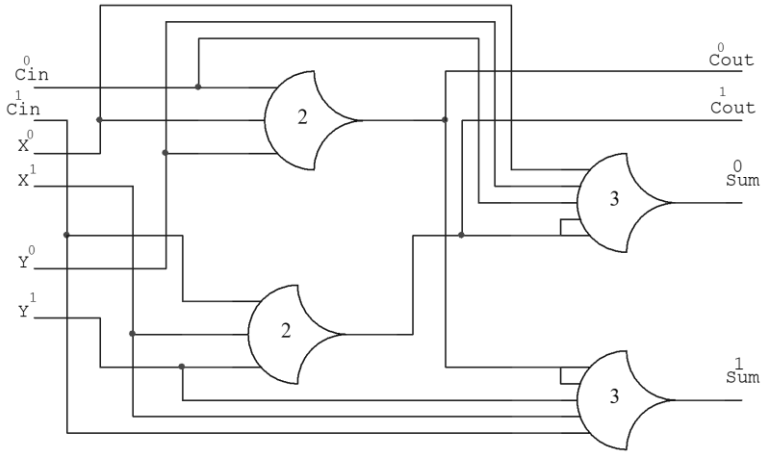

Figure 6. 1-bit NCL full adder

Step 6: Function test simulation

The 4-bit NCL RCA shown in Figure 10 was used to simulate and synthesize with the TSMC 65nm technology library. These step results are presented in section 3.

2.3.2. Carry Look Ahead adder

Steps 1 and 2 are the same as the steps in section 2.3.1.

Step 3, 4: With the Sum output signals, their equations are the same as the equation (3) and (4). This is because the CLA adder is only optimized for speed by optimizing the Cout output signal compared to the full-adder.

Since the basic equations of CLA for the Cout output signal depend on the variables P, G, and Carry input signal as shown in (7), the Karnaugh map for the NCL Carry output signal is implemented with three of those variables

$$C_i = G_i + P_i C_{i-1} \tag{7}$$

Where G = XY, P = X ^ Y

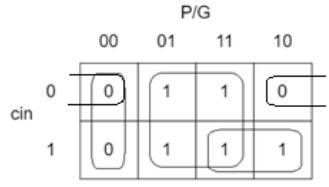

Figure 7. Karnaugh map for CLA Cout signal

Based on the Karnaugh map shown in figure 7, the equations for the NCL carry output signal are:

$$Cout^0 = P^0 G^0 + G^0 Cin^0 \tag{8}$$

$$Cout^1 = G^1 + P^1 Cin^1 \tag{9}$$

About the observability, the Sum and Cout signals are satisfied. In term of input completeness, since the NCL CLA Sum output signal is the same as the Sum signal of the full-adder, the 1-bit NCL CLA adder satisfies the input completeness.

Step 5: Logic mapping

Similar to the full-adder, 27 NCL threshold gates [17] are mapped to (1) and (2) shown in Figure 8.

Step 6: Function test simulation

The 4-bit NCL CLA adder shown in Figure 12 was used to simulate and synthesize with the TSMC 65nm technology library. These step results are presented in Section 3.

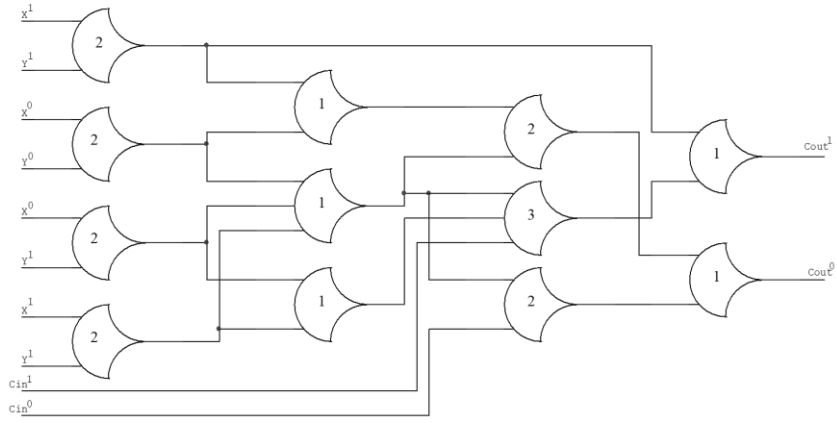

Figure 8. 1-bit NCL CLA adder

3. SIMULATION AND SYNTHESIS RESULTS

3.1 Simulation results

The simulation of synchronous and asynchronous based on NCL RCA and CLA adder is detailed in this section.

The architectures for 4-bit RCA and CLA adders are shown in Figures from 9 to 12 with both synchronous and asynchronous versions.

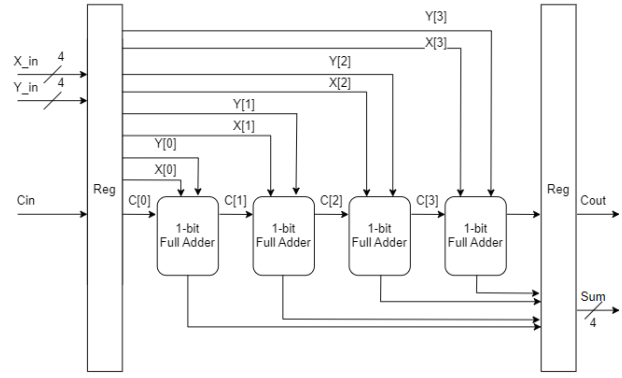

Figure 9. 4-bit synchronous RCA

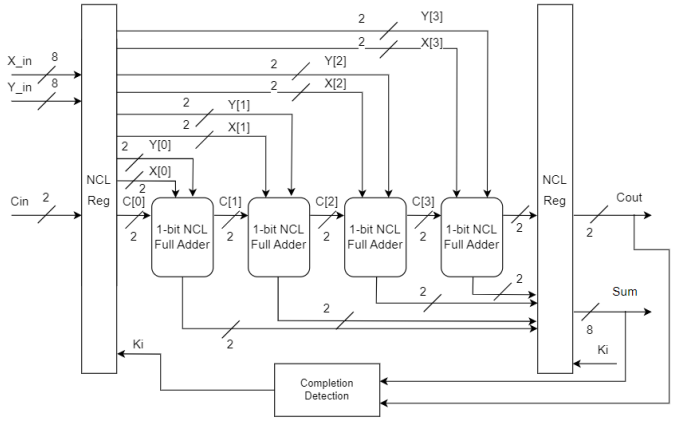

Figure 10. 4-bit NCL based asynchronous RCA

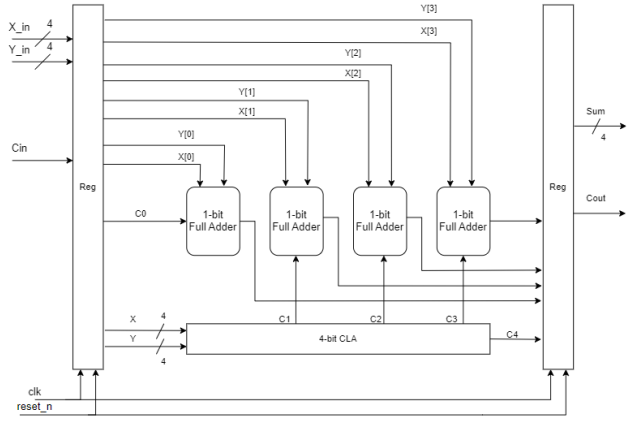

Figure 11. 4-bit synchronous CLA

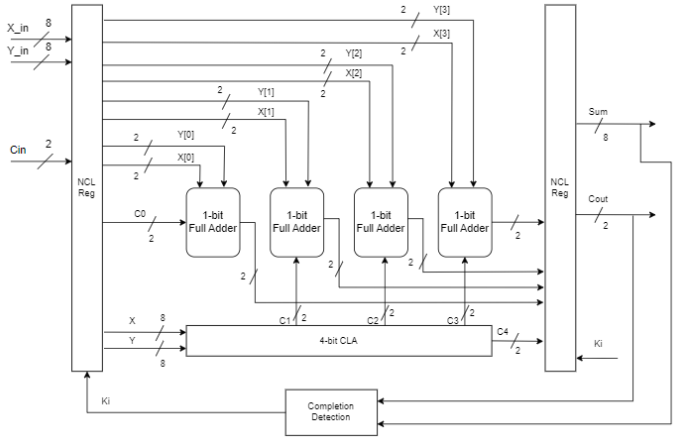

Figure 12. 4-bit NCL based asynchronous CLA

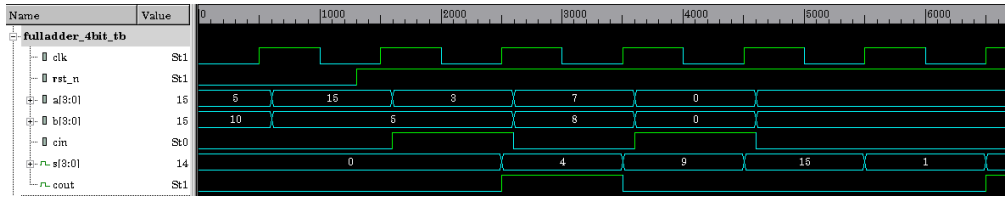

Figure 13. The simulation result of 4-bit synchronous RCA

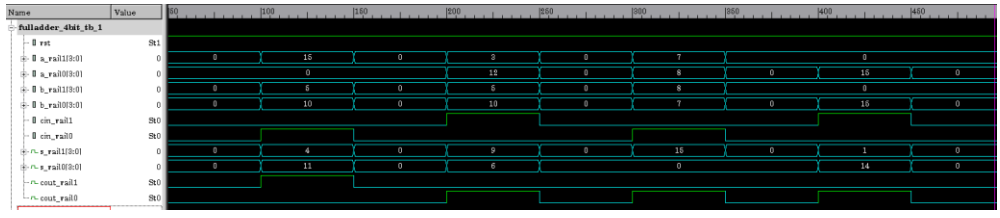

Figure 14. The simulation result of 4-bit NCL based asynchronous RCA

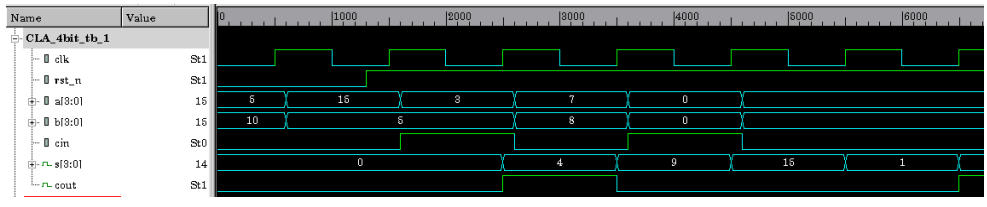

Figure 15. The simulation result of 4-bit synchronous CLA

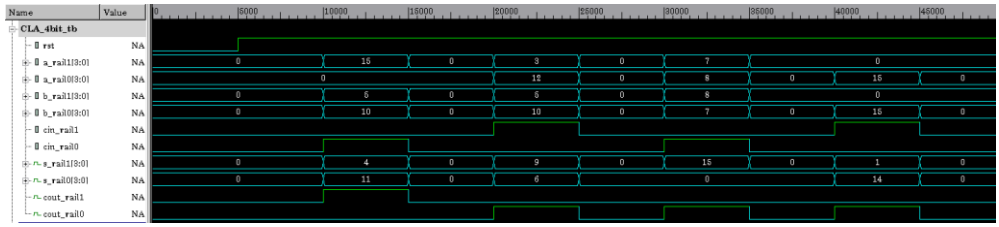

Figure 16. The simulation result of 4-bit NCL based asynchronous CLA

Figures from 13 to 16 show that the adder's simulation results, where a, and b are inputs. Cin is an input carry signal, and Cout is an output carry signal, S stands for sumary.

For synchronous adders, the ouputs are correct after the second positive clock edge, while the output of the asynchronous adder appears immediately since no clock signal is used. NCL asynchronous data are interleaved with NULL through the handshake mechanism, which prevents the next data from overwriting each other.

3.2 Synthesis results

In this section, the Design Compiler (DC) tool was used to perform the synthesis of adders with the TSMC 65nm technology library. The synthesis results of 4-bit synchronous and NCL based asynchronous full adders for RCA and CLA are shown in Table 4, and 5, respectively.

```
Number of ports:                           50
Number of nets:                            68
Number of cells:                           24
Number of combinational cells:              5
Number of sequential cells:                14
Number of macros/black boxes:               0
Number of buf/inv:                          1
Number of references:                       3

Combinational area:                41.760000
Buf/Inv area:                       1.440000
Noncombinational area:            168.000000
Macro/Black Box area:               0.000000
Net Interconnect area:      undefined  (No wire load specified)

Total cell area:                  209.760000
Total area:                 undefined
1
```

Figure 17. The area result of 4-bit synchronous RCA

```
                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000              0.0000             0.0000  (   0.00%)
memory             0.0000           0.0000              0.0000             0.0000  (   0.00%)
black_box          0.0000           0.0000              0.0000             0.0000  (   0.00%)
clock_network      0.0000           0.0000              0.0000             0.0000  (   0.00%)
register       2.4594e-02       4.2690e-04          2.7022e+03         2.5023e-02  (  95.39%)
sequential         0.0000           0.0000              0.0000             0.0000  (   0.00%)
combinational  7.0000e-04       5.0791e-04          1.2982e+03         1.2092e-03  (   4.61%)
--------------------------------------------------------------------------------------------------
Total          2.5294e-02 mW     9.3482e-04 mW     4.0004e+03 pW      2.6232e-02 mW
1
```

Figure 18. The power result of 4-bit synchronous RCA.

```
Number of ports:                           30
Number of nets:                           162
Number of cells:                          142
Number of combinational cells:            124
Number of sequential cells:                18
Number of macros/black boxes:               0
Number of buf/inv:                         38
Number of references:                      14

Combinational area:               312.960001
Buf/Inv area:                      54.720002
Noncombinational area:            181.439999
Macro/Black Box area:               0.000000
Net Interconnect area:      undefined  (No wire load specified)

Total cell area:                  494.400000
Total area:                 undefined
1
```

Figure 19. The power result of 4-bit NCL asynchronous RCA

```
                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000            0.0000             0.0000  (   0.00%)
memory             0.0000           0.0000            0.0000             0.0000  (   0.00%)
black_box          0.0000           0.0000            0.0000             0.0000  (   0.00%)
clock_network      0.0000           0.0000            0.0000             0.0000  (   0.00%)
register       6.8230e-02       8.9036e-04        3.1086e+03         6.9124e-02  (  87.66%)
sequential         0.0000           0.0000            0.0000             0.0000  (   0.00%)
combinational  5.5856e-03       4.1439e-03        5.4590e+03         9.7350e-03  (  12.34%)
--------------------------------------------------------------------------------------------------
Total          7.3816e-02 mW     5.0343e-03 mW     8.5676e+03 pW      7.8859e-02 mW
1
```

Figure 20. The power result of 4-bit NCL asynchronous RCA

```
Timing Path Group (none)
-----------------------------------
Levels of Logic:              22.00
Critical Path Length:          1.93
Critical Path Slack:         uninit
Critical Path Clk Period:       n/a
Total Negative Slack:          0.00
No. of Violating Paths:        0.00
Worst Hold Violation:          0.00
Total Hold Violation:          0.00
No. of Hold Violations:        0.00
-----------------------------------
```

Figure 21. The max speed result of 4-bit NCL asynchronous RCA

```
Number of ports:                           16
Number of nets:                            34
Number of cells:                           19
Number of combinational cells:              5
Number of sequential cells:                14
Number of macros/black boxes:               0
Number of buf/inv:                          1
Number of references:                       3

Combinational area:                 41.760000
Buf/Inv area:                        1.440000
Noncombinational area:             168.000000
Macro/Black Box area:                0.000000
Net Interconnect area:      undefined  (No wire load specified)

Total cell area:                   209.760000
Total area:                 undefined
1
```

Figure 22. The area result of 4-bit synchronous CLA

```
                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000            0.0000             0.0000  (   0.00%)
memory             0.0000           0.0000            0.0000             0.0000  (   0.00%)
black_box          0.0000           0.0000            0.0000             0.0000  (   0.00%)
clock_network      0.0000           0.0000            0.0000             0.0000  (   0.00%)
register       2.4595e-02       4.6201e-04        2.7022e+03         2.5060e-02  (  95.56%)
sequential         0.0000           0.0000            0.0000             0.0000  (   0.00%)
combinational  6.9766e-04       4.6538e-04        1.2985e+03         1.1643e-03  (   4.44%)
--------------------------------------------------------------------------------------------------
Total          2.5293e-02 mW     9.2739e-04 mW     4.0007e+03 pW      2.6224e-02 mW
1
```

Figure 23. The power result of 4-bit synchronous CLA

```
Number of ports:                          27
Number of nets:                          137
Number of cells:                         118
Number of combinational cells:           100
Number of sequential cells:               18
Number of macros/black boxes:              0
Number of buf/inv:                        44
Number of references:                      9

Combinational area:               251.520005
Buf/Inv area:                      63.360003
Noncombinational area:            181.439999
Macro/Black Box area:               0.000000
Net Interconnect area:      undefined  (No wire load specified)

Total cell area:                  432.960003
Total area:                 undefined
1
```

Figure 24. The area result of 4-bit NCL asynchronous CLA

```
                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000              0.0000             0.0000  (   0.00%)
memory             0.0000           0.0000              0.0000             0.0000  (   0.00%)
black_box          0.0000           0.0000              0.0000             0.0000  (   0.00%)
clock_network      0.0000           0.0000              0.0000             0.0000  (   0.00%)
register       6.8190e-02       4.0110e-04        3.1088e+03         6.8594e-02  (  91.30%)
sequential         0.0000           0.0000              0.0000             0.0000  (   0.00%)
combinational  3.5328e-03       2.9963e-03        4.2726e+03         6.5333e-03  (   8.70%)
--------------------------------------------------------------------------------------------------
Total          7.1723e-02 mW     3.3974e-03 mW     7.3815e+03 pW      7.5127e-02 mW
1
```

Figure 25. The power result of 4-bit NCL asynchronous CLA

```
Timing Path Group (none)
-----------------------------------
Levels of Logic:              21.00
Critical Path Length:          1.08
Critical Path Slack:         uninit
Critical Path Clk Period:       n/a
Total Negative Slack:         0.00
No. of Violating Paths:       0.00
Worst Hold Violation:         0.00
Total Hold Violation:         0.00
No. of Hold Violations:       0.00
-----------------------------------
```

Figure 26. The power result of 4-bit NCL asynchronous CLA

The area and power results' synchronous RCA and CLA synthesized at 100MHz frequency are shown in figure 17, 18, 22, and 23, respectively. The maximum speed is the maximum frequency that synchronous RCA's timing is met. The area's synchronous RCA is equal to the area's synchronous CLA as showed in figure 17 and figure 22 because logic circuits of those adders are optimized by the DC tool. The power synthesis results are similar to the area synthesis results showed in figure 18 and figure 23. The Table 4 and Table 5 summarize the synthesis results in figures (from figure 17 to figure 26) for 4-bit RCA and 4-bit CLA implemented by the synchronous and asynchronous method.

Table 4. Synthesis results of 4-bit RCA

	Area (μm^2)	Power (μW) @100MHz	Max speed (MHz)
Synchronous	209.76	26.228	2700
Asynchronous	494.40	9.735	259.067
Ratio (Syn/Asyn)	0.4243	2.6942	10.422

Table 5. Synthesis results of 4-bit CLA

	Area (μm^2)	Power (μW) @100MHz	Max speed (MHz)
Synchronous	209.76	26.224	3000
Asynchronous	432.96	6.533	462.963
Ratio (Syn/Asyn)	0.4844	4.014	6.480

Table 4 and Table 5 indicate that the results of the area, and power, of the synchronous RCA, and CLA are the same because of the tool optimization. The area, and power were measured at 100Mhz frequency. With that frequency, the DC tool shall optimize the area to get the good result because of the met timing.

The total area of the asynchronous based on NCL RCA, and CLA adders increased by 57.57% and 51.55% compared to those of the synchronous RCA, and CLA adders, respectively. This is one of the disadvantages of asynchronous circuits.

About the maximum speed, the NCL based asynchronous adders produced more delay than synchronous adders, which results in the maximum speed of the asynchronous RCA, and CLA adders reduced 10.422, and 6.48 times compared to those of synchronous adders.

The area and delay of the NCL circuits are the most significant drawbacks. This is because the combinational logic creates the detection circuits in the NCL registers making the feedback paths large and cumbersome. These drawbacks could be overcome by a special design technique shown in [30]. Besides, we used the conventional standard cell libraries in order to synthesize the NCL asynchronous designs. Therefore, the synthesis results could not be achieved at the most optimal level. These results could be improved much more if the NCL asynchronous designs are synthesized by the NCL cell libraries.

In term of power consumption, the synchronous RCA and CLA adders consume 2.693 (increase 62,88%), and 4.014 times (increase 75,09%) power more than NCL based asynchronous RCA and CLA adders. Since the NCL circuits do not use clock pulses, they reduce a large amount of switching power when the clock edge is active.

Despite the disadvantage of area and speed, the NCL based asynchronous circuits achieves lower power when compared to synchronous circuits.

4. CONCLUSION

In this research, we proposed a general flow to implement the combinational logic circuits using asynchronous design method based on NCL. CLA adder and RCA are chosen as case studies to implement. These adders are synthesized by DC tool using the conventional cell library. In addition, we also make the comparison of the implemented results of adders mentioned above by using synchronous and NCL based asynchronous techniques. The implemented results indicate that the NCL based asynchronous RCA, and CLA adders' powers were reduced by 62.88% (RCA), and 75.09% (CLA) when compared to the Boolean logic combinational designs.

REFERENCES

[1] B. Penumutchi, S. Vella, H. Satti, "Kogge Stone Adder with GDI technique in 130nm technology for high performance DSP applications," *2017 Int. Conf. on Smart Technologies For Smart Nation (SmartTechCon)*, 2017, pp. 5-10, doi: 10.1109/SmartTechCon.2017.8358334.

[2] R. Katreepalli, *et al*., "A power-delay efficient carry select adder," *2017 2nd Int. Conf. Convergence in Technology (I2CT 2017),* 2017, pp. 1234–1238, doi: 10.1109/TPAS.1970.292653.

[3] D. Karthik and S. Jayamani, "High speed energy efficient carry skip adder operating at different voltage supply," *Proc. 2016 IEEE Int. Conf. on Wireless Communications, Signal Processing and Networking (WiSPNET 2016),* 2016, pp. 191–195, doi:10.1109/I2CT.2017.8226324

[4] H. You, *et al.*, "An Energy and Area Efficient Carry Select Adder with Dual Carry Adder Cell", *Electronics Journal*, vol.8, no. 10, 2019, pp. 2–10, doi: 10.3390/electronics8101129.

[5] Ananthakrishnan *et al.*, "FPGA Based Performance Comparison of Different Basic Adder Topologies with Parallel Processing Adder," *Proc. 3rd Int. Conf. on Electronics and Communication and Aerospace Technology (ICECA 2019)*, 2019, pp. 87–92, doi:10.1109/ICECA.2019.8821925.

[6] S. Patel, *et al.*, "Area-delay efficient and low-power carry skip adder for high performance computing systems," *Proc. - 2019 IEEE Int. Symp. on Smart Electronic Systems (iSES 2019)*, 2019, pp. 300–303, doi: 10.1109/iSES47678.2019.00074.

[7] S. Erniyazov, and J. C. Jeon, "Carry save adder and carry look ahead adder using inverter chain based coplanar QCA full adder for low energy dissipation," *Microelectronic Engineering 211*, 2019, pp. 37–43, doi: 10.1016/j.mee.2019.03.015.

[8] S. Radhakrishnan, *et al.*, "Fault Tolerant Carry Save Adders - A NMR Configuration Approach," *2018 Int. Conf. on Control, Power, Communication and Computing Technologies (ICCPCCT 2018)*, 2018 pp. 210–215, doi:10.1109/ICCPCCT.2018.8574227.

[9] A. N. M. Hossain and M. A. Abedin, "Implementation of an XOR Based 16-bit Carry Select Adder for Area, Delay and Power Minimization," *2nd Int. Conf. on Electrical, Computer and Communication Engineering (ECCE 2019),* 2019, pp. 1–4. doi: 10.1109/ECACE.2019.8679293.

[10] K. Bhavani, *et al.*, "A comparative study on adders," *Proc. 2017 Int. Conf. on Wireless Communications, Signal Processing and Networking (WiSPNET 2017)*, 2017 pp. 2206–2230, doi: 10.1109/WiSPNET.2017.8300155.

[11] M. Hasan, *el al.*, "Overview and comparative performance analysis of various full adder cells in 90 nm technology," *2018 4th Int. Conf. on Computing Communication and Automation (ICCCA 2018)*, 2018, pp. 1–6, doi: 10.1109/CCAA.2018.8777684.

[12] C. H. Lai, *et al.*, "Radix-4 adder design with refined carry," *2017 IEEE Conf. on Dependable and Secure Computing*, 2017, pp. 300–304, doi: 10.1109/DESEC.2017.8073851.

[13] P. Saxena, "Design of low power and high speed Carry Select Adder using Brent Kung adder," *2015 Int. Conf. VLSI Systems, Architecture, Technology and Applications, (VLSI-SATA 2015)*, 2015, doi: 10.1109/VLSI-SATA.2015.7050465.

[14] B. G. Fawzy *et al.*, "Strong Indication Full-Adder Circuit for NULL Convention Logic Automation Flows," *18th Int. Symp. Communication and Information Technology (ISCIT 2018)*, 2018, pp. 416–421, doi: 10.1109/ISCIT.2018.8588000.

[15] A. Vakil, *et al.*, "Comparitive analysis of null convention logic and synchronous CMOS ripple carry adders," Proc. 2017 2nd IEEE Int. Conf. on Electrical, Computer and Communication Technologies (ICECCT 2017) (2017) 2–6, doi: 10.1109/ICECCT.2017.8117926.

[16] A. J. Albert and s. Ramachandran, "Static implementation of a null convention logic based exponent adder," *Int. Journal of Applied Engineering Research*, vol. 10, no. 3, pp. 7601–7614, 2015.

[17] J. Wu, "Null Convention Logic applications of asynchronous design in nanotechnology and cryptographic security," *Doctoral Dissertations*, 2012.

[18] D. L. D. Oliveira *et al.*, "Synthesis of QDI combinational circuits using null convention logic based on basic gates," *Advances in Science, Technology and Engineering Systems*, vol. 3 no. 4, pp. 308–317, 2018.

[19] A. A. Sakib *et al.*, "Implementation of FinFET Based Static NCL Threshold Gates: An Analysis of Design Choice," *Midwest Symp. on Circuits and Systems 2020-Augus*, 2020, pp. 37–40, doi: 10.1109/MWSCAS48704.2020.9184629.

[20] N. L. Huy and P. Beckett, "Null convention logic primitive element architecture for ultralow power high performance portable digital systems," *Proc. 2017 IEEE Regional Symp. on Micro and Nanoelectronics (RSM 2017)*, 2017, pp. 167–170, doi: 10.1109/RSM.2017.8069157.

[21] M. Howard *et al.,* "Investigation and Comparison of Bus Alternatives for Asynchronous Circuits," *Conf. Proc. - IEEE Southeastcon*, 2018, pp. 1–2, doi: 10.1109/SECON.2018.8478988.

[22] D. V. Supriya and R. Niranjan, "Realization of AES Encryption and Decryption Based On Null Convention Logic," *Int. Research Journal of Engineering and Technology (IRJET)*, vol. 2, no. 7, pp. 77–81, 2015.

[23] K. Haque *et al.,* "Low Power Spatial Computing Using Null Convention Logic," *Proc. 2015 IEEE Int. Conf. on Data Science and Data Intensive Systems*, 2015, pp. 325–329, doi: 10.1109/DSDIS.2015.91.

[24] H. J. Lee and B. Y. Kim, "Low power Null Convention Logic circuit design based on DCVSL," *Midwest Symp. on Circuits and Systems*, 2013, pp. 29–32. doi: 10.1109/MWSCAS.2013.6674577.

[25] P. Metku *et al.*, "Low-Power Null Convention Logic Multiplier Design Based on Gate Diffusion Input Technique," *Proc. – Int. SoC Design Conf. 2018 (ISOCC 2018)*, 2018, pp. 233–234, doi: 10.1109/ISOCC.2018.8649885.

[26] M. T. Moreira *et al.,* "NCL synthesis with conventional EDA tools: Technology mapping and optimization," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 6, pp. 1981–1993, 2018.

[27] R. A. *et al.*, "A comparison of asynchronous QDI templates using static logic," *8th IEEE Latin American Symp. on Circuits and Systems (LASCAS 2017 R9)*, 2017, pp. 1–4, doi: 10.1109/LASCAS.2017.7948103.

[28] G. E. Sobelman and K. Fant, "CMOS circuit design of threshold gates with hysteresis," *Proc. - IEEE Int. Symp. on Circuits and Systems 2*, 1998, pp. 61–64, doi: 10.1109/iscas.1998.706841.

[29] A. Caberos *et al.,* "Area-efficient CMOS implementation of NCL gates for XOR-AND/OR dominated circuits," *Asia Pacific Conf. on Postgraduate Research in Microelectronics and Electronics*, 2017, pp. 37–40, doi: 10.1109/PRIMEASIA.2017.8280358.

[30] C. P. Taylor , "Null Convention Logic Asynchronous Register Full Path Completion Feedback Loop Using Two Stage Voltage Divider," Dissertations, 2014. [online]. Available: https://corescholar.libraries.wright.edu/etd_all/1187.

TÓM TẮT

PHƯƠNG PHÁP MỚI ĐỂ CẢI THIỆN CÔNG SUẤT CHO CÁC BỘ CỘNG

Lê Thành Tới[1]*, Trương Trí Lạc[2]

[1]*Trường Đại học Công Thương Thành phố Hồ Chí Minh*

[2]*Công ty TNHH Một Thành Viên Uniquify Việt Nam*

*Email: *toilt@huit.edu.vn*

Bộ cộng là một trong những thành phần quan trọng tạo nên hệ thống tính toán. Việc tối ưu hóa công suất, tốc độ và độ trễ của bộ cộng đã góp phần vào việc tiết kiệm và sử dụng năng lượng hiệu quả. Bài báo này đề xuất một quy trình thiết kế mạch logic tổ hợp sử dụng phương pháp thiết kế bất đồng bộ dựa trên Null Convention Logic (NCL). Để minh họa cho quy trình được đề xuất, Carry Look Ahead (CLA) và Ripple Carry Adder (RCA) được chọn để thực hiện. Các bộ cộng này được tổng hợp bằng công cụ DC (Design Compiler) với các thư viện cell thông thường. Ngoài ra, chúng tôi cũng thực hiện so sánh kết quả tổng hợp của các bộ cộng được đề cập ở trên bằng cách sử dụng các kỹ thuật bất đồng bộ dựa trên NCL và kỹ thuật đồng bộ. Kết quả tổng hợp cho thấy công suất của các thiết kế bất đồng bộ dựa trên NCL giảm 62,88% (RCA) và 75,09% (CLA) so với các thiết kế đồng bộ tương ứng.

Từ khóa: RCA, CLA Adder, Null Convention Logic, Mạch tổ hợp NCL, Phương pháp bất đồng bộ.