UDC 621.518
DOI: 10.15587/1729-4061.2025.347658

DESIGN OF IMPROVED STRUCTURES FOR MULTI-BIT DATA ADDITION DEVICES IN BINARY CODES OF THE RADEMACHER THEORETICAL-NUMERICAL BASIS

Yaroslav Nykolaychuk
Doctor of Technical Sciences, Professor*
Igor Pitukh
PhD, Associate Professor*
Volodymyr Hryha
Corresponding author
PhD, Associate Professor
Department of Computer Engineering and Electronics
Vasyl Stefanyk Carpathian National University
Shevchenko str., 57, Ivano-Frankivsk, Ukraine, 76018
E-mail: volodymyr.hryha@cnu.edu.ua
*Department of Specialized Computer Systems
West Ukrainian National University
Lvivska str., 11, Ternopil, Ukraine, 46009

This study investigates the process of designing multi-bit adders based on an improved element base of their components in binary codes of Rademacher theoretical-numerical basis. The task addressed relates to the fact that the known element base of modern single-bit adders does not make it possible to achieve the minimax characteristics of sum formation and carry-save efficiency in their structures in 1 micro cycle.

The improved element base of single-bit adders has been built on the basis of the "Exclusive AND" logic element, which has low hardware complexity and high speed of output signal formation.

Improved architectural solutions for full and half single-bit binary adders with minimax characteristics of hardware and time complexity in structures of multi-bit adders of various types have been applied. The system characteristics of microelectronic structures of multi-bit adders of various types based on full and half single-bit binary combinational adders with direct, inverse, and paraphase inputs and outputs have been investigated.

As a result, it has been theoretically established and practically confirmed that multi-bit adders based on an improved element base have 2 times less hardware complexity and 6 times higher speed.

The structure of a cascaded n-bit fast adder has been designed, which has a speed 1.8 times higher than that of the known implementation.

Using the VHDL hardware description language and Vivado CAD, a 64-bit cascade adder was modelled and synthesized on FPGA, which has twice the speed of the known adder.

The designed multi-bit Rademacher basis adders are applicable to statistical, correlation, and entropy analysis tasks, video image processing, image recognition, as well as various artificial intelligence tasks

Keywords: cascade and pyramid adders, full and half adders, time and hardware complexity, FPGA

Received 03.10.2025
Received in revised form 21.11.2025
Accepted 19.12.2025
Published 30.12.2025

How to Cite: *Nykolaychuk, Y., Pitukh, I., Hryha, V. (2025). Design of improved structures for multi-bit data addition devices in binary codes of the Rademacher theoretical-numerical basis. Eastern-European Journal of Enterprise Technologies, 6 (2 (138)), 72–83. https://doi.org/10.15587/1729-4061.2025.347658*

1. Introduction

Representation of digital data in binary codes of the Rademacher number-theoretic basis is the fundamental basis for the formation, transformation, and digital processing of data in modern computer technology [1, 2].

The operation of adding and accumulating sums of binary codes is basic in the implementation of more complex functional operations of squaring, multiplication, division, taking the square root, and others [3–6]. The operation of data summation is also widely used in solving problems of processing RGB pixels of color images [6] and entropy analysis [4, 6]. Adding codes and accumulating sums of binary codes are components of algorithms for statistical, correlation, spectral, and entropy analysis. It should be noted that the summation and accumulation of binary data provides their integration, which makes it possible to significantly reduce the volume of data, change their correlation and spectral characteristics [6].

In addition, the arithmetic operation of addition exists in almost all signal processing and computing algorithms. This operation and the components that implement it are a weighting attribute that can significantly affect the performance, hardware and structural complexity of multi-bit binary adders and their components [3, 4].

Multi-bit binary adders are divided into the following classes: linear (cascade, chain) without structural branches; pyramidal adders; fast adders; and parallel prefix adders [3, 7].

Depending on the problem to be solved, there are three main types of half adders (HAs) and full binary adders (FAs) [6]:

– with single-phase direct input-outputs;
– with single-phase inverse input-outputs;
– with paraphase input-outputs.

Most often, single-phase binary half adders with direct input-outputs are used in multi-bit binary adders. Adders of this class are used as the lower digits of multi-bit combinational adders of binary codes.

Single-bit FAs and HAs are basic components in the construction of arithmetic-logical units (ALUs) for universal and specialized processors in personal computers and microcontrollers, as well as coprocessors and accelerators of computational operations [3, 5, 6, 8, 9]. Multi-bit fast adders are effectively used in processors for sorting digital data arrays of vector and scalar supercomputers, processors for digital processing of multi-bit codes during the implementation of data encryption algorithms, etc. The number of single-bit half and full binary adders in the structures of these devices can be 3–6 orders of magnitude.

Therefore, improving the architectural and structural solutions for the element base of addition devices will make it possible to increase the speed of performing the arithmetic operation of adding data represented by binary codes. Evaluating the system characteristics of the time and hardware complexity of single-bit and multi-bit adders will make it possible to optimize the structures of special processors that solve complex computational problems, as well as their microelectronic implementation on FPGA.

2. Literature review and problem statement

The classical structure of a half binary adder (HA) is described in [3]; it contains a 2-input logical element "AND" and a 2-input logical element "Exclusive OR". It should be noted that the internal structure of the logical element "Exclusive OR" consists of four logical elements "2AND-NOT" and has a time complexity of 3 micro cycles. It is shown that the hardware complexity of such a classical binary half adder is 5 gates, and the time complexity of obtaining the output signal of the sum is 3 micro cycles, and the output signal of the carry is 1 micro cycle. The reported results of research into the system characteristics of such a classical half binary adder are redundant and do not make it possible to maximize the performance of multi-bit binary adders in modern ALU processors. The reason for this is that the formation of the sum bit occurs much slower than the formation of the output through-carry, which causes a significant delay in signals of this type of half binary adders. For example, when using such HA structures in pyramidal multi-bit adders (for $n > 128$), their hardware complexity increases by 2–3 orders of magnitude while the time complexity increases several times.

This approach is used in [4, 5], in which their authors proposed an improved HA structure, which is built on the basis of logical elements "AND", "OR", "NOT". It is shown that the hardware complexity of such an improved binary HA is lower compared to the known one [3] and is 4 gates, while the time complexity of obtaining the sum and output carry are the same as the characteristics of a classical half adder (HA). The results of the studies show that the system characteristics of HA are partially improved and require further improvement of the structure.

In work [6], a more advanced structure of HA is proposed, which is built on the basis of logical elements "AND", "OR", "AND-NOT", "NOT". It is shown that the hardware complexity of such an improved binary HA is 4 gates, and the time complexity of obtaining the sum and the output carry is 2 micro cycles, which makes it possible to avoid unnecessary delays in multi-bit structures of adders.

In [8], the time complexity of HA at the output of the through-carry is 2 micro cycles, and at the output of the sum is 3 micro cycles, which respectively requires a delay of through-carries by $2n$-micro cycles, when used in structures of n-bit multi-bit adders.

But the unsolved part of the problem in [6, 8] is the impossibility of forming the sum and through-carry signals with a delay of 1 micro cycle, which requires further improvement of this type of adders.

From the above results, it can be concluded that the fundamental disadvantage of half binary adders is the low speed and high hardware complexity of the known structures [3–6, 8]. That is, the time complexity of HA at the output of the through-carry is 1 or 2 micro cycles, and at the output of the sum is 2 or 3 micro cycles.

In work [9], the classical structure of a full binary adder is presented, which contains two 2-input logical elements "AND", a 2-input logical element "OR" and two 2-input logical elements "Exclusive OR". It is shown that the hardware complexity of such a classical binary full adder is 11 gates, and the time complexity of obtaining the sum is 6 micro cycles, and the output carry is 2 or 5 micro cycles. The results of research on such a classical full binary adder have high hardware and time complexity, which significantly affects the performance of multi-bit binary adders in modern ALU processors. Another disadvantage of such an adder is the ambiguity of the formation of the through-carry and high structural complexity.

In [5, 7], an improved structure of a single-bit combinational full adder based on logical elements "AND-NOT" and multi-input elements "OR-NOT" was proposed. It was shown that the hardware complexity of such a binary full adder is 11 gates, and the time complexity of such an adder at the sum output is 5 micro cycles, and at the through-carry output – 3 micro cycles. The obtained system characteristics of such a classical full binary adder have a large hardware and time complexity, which significantly affects the performance of multi-bit binary adders in modern ALU processors. Another disadvantage of such an adder is the ambiguity of the formation of the through-carry and the large structural complexity. The presented structure of such an adder has limited functional capabilities for their implementation on crystals, which is due to the presence of multi-input logical elements "OR-NOT". Another disadvantage of such an FA structure is the presence of only direct inputs and outputs, as well as significant structural complexity.

At the same time, the problem of forming the sum signal and the through-carry signal in a full binary adder [7, 9] for the minimum possible number of micro cycles remains unsolved. All this gives grounds to argue that the structures of HAs and FAs still require deeper improvement.

An option for overcoming such difficulties associated with high hardware complexity and low time complexity of forming the sum and through-carry of HA and FA is to study their structural and technological capabilities [8].

In [10], a structure of the adder with an accelerated carry is proposed, which is built on the basis of m-bit adders and 2-input multiplexers. The disadvantage of such an adder with an accelerated carry is low speed and high hardware complexity. The research results indicate that the low performance of such an adder with an accelerated carry is due to the delay of the through-carry in a single-bit full adder with direct inputs and outputs of the carry and the delay of signals in a multiplexer with a single-phase direct switching signal. The high hardware complexity of such an adder is due to the high hardware complexity of $(k-1)$ pairs of m-bit adders and the presence of m-bit and 2-bit multiplexers in each $(k-1)$-bit of the device.

Analysis of known structures of FAs and HAs gives grounds to assert that the task to design improved structures for such adders with minimax characteristics of hardware and

time complexity remains to be solved. It is the implementation of highly efficient hardware algorithms for data summation on a modern element base and the improvement of their structures that could make it possible to obtain high-performance components of modern ALU processors, hardware accelerators, and coprocessors.

3. The aim and objectives of the study

The purpose of our research is to design multi-bit binary adders of various types on an improved element base with minimax characteristics of hardware and time complexity. This will make it possible to significantly reduce the cost of hardware implementation and speed of multi-bit components of vector and scalar processors in supercomputers, in particular arithmetic and logical devices (ALUs), coprocessors, and hardware accelerators for performing operations on binary numbers.

To achieve the goal, it is necessary to solve the following tasks:

– to apply the proposed improved structures of full and half binary adders with minimax characteristics as part of multi-bit binary adders of various types and determine their system characteristics in terms of time and hardware complexity;

– to design the structure of a cascade n-bit adder with the accelerated carry for data summation in the Rademacher number-theoretic basis and determine its system characteristics;

– to model and synthesize a multi-bit cascade-type adder on FPGA using Active-HDL and Vivado automated design systems.

4. The study materials and methods

The object of our study is the process of designing multi-bit adders based on the improved element base of their components in binary codes of the Rademacher number-theoretic basis.

The principal hypothesis assumes that the speed of multi-bit adders could be increased by improving the element base of full and half binary adders at the level of logical carry. The main limitation of the known structures of HAs and FAs at the logical level is a significant delay in the formation of sum and through-carry signals. This drawback can be improved by using the proposed logic element "Exclusive AND" as a known "XOR" component [11]. In addition to increasing the speed of multi-bit adders, it is also expected to reduce their hardware complexity due to a smaller number of logical elements required for the implementation of HAs and FAs.

To design improved structures of single-bit adders and an adder circuit with an accelerated carry, circuit design methods were used [12]. When studying the system characteristics of half and complete single-bit binary adders, the theory of complexity [13] and methods of comparative analysis [14] were applied. Analytical methods were used to determine the hardware complexity (number of gates) and time complexity (number of micro cycles) of single-bit FAs and HAs, adder circuits with an accelerated carry, and various types of multi-bit adders built on their basis [15].

To verify the correct functioning of the designed devices, a method of computer modeling (simulation) was used. The modeling of the designed multi-bit adder structures was performed in the Active-HDL integrated environment by the Aldec company (USA) at the functional level. As input data of multi-bit adders, integers with a fixed decimal point ($n = 64$) were used. To measure the actual indicators of system characteristics (actual indicators of hardware complexity and type code) of multi-bit adders, the method of experimental synthesis was used. The synthesis of multi-bit adders of various types was carried out on FPGAs from Xilinx (USA), Artix7 family, in the Vivado Design Suite environment.

5. Results of investigating multi-bit data summation devices in the Rademacher basis

5. 1. Applying the improved HA and FA structures in multi-bit adders of various types

The boundary estimate of the minimum time complexity, which ensures the maximum speed of single-bit adders, is the simultaneous formation of direct and inverse output signals $(S_i, \overline{S_i}, C_{out}, \overline{C_{out}})$ of the sum and carry with a minimum time delay of signals by 1 micro cycle. Such speed parameters determine the relevance of solving the scientific and applied problem for the synthesis and improvement of the structures of components of complex devices and special processors of computing equipment.

Fig. 1 shows the structure of an improved half binary adder built on the logic element "Exclusive AND" [15].

The improvement of such a single-bit half adder was achieved by simplifying the structure and microelectronic implementation of the "Exclusive OR" logic element on the "AND-NOT" and "OR" logic elements, the outputs of which are combined and implement the "Exclusive AND" logic element. This allowed us to reduce the hardware complexity to three logic elements, i.e., by 2–3 times, and increase the speed of the sum signal operation and through-carry per 1 micro cycle, i.e., by 3 times compared to known implementations of HAs [3].

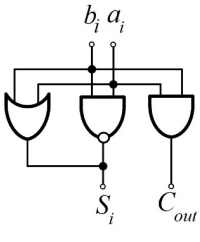

Fig. 1. Structure of the improved half binary adder based on the "Exclusive AND" logic element

Such a structure of a half binary adder can be used as a component of multi-bit binary adders of the pyramidal type.

Fig. 2 shows the structure of an 8-bit pyramidal type adder.

The hardware complexity of this type of pyramidal multi-bit adders is calculated from the following formula

$$A_{PA} = \frac{m^2 + m}{2} \times A_{HA}, \quad (1)$$

where A_{HA} is the hardware complexity of a half binary adder, m-bit input data.

For example, when $m = 8$ using classical HAs in the structure of a pyramidal 8-bit adder: $A_1 = \frac{8^2 + 8}{2} \times 5 = 180$ (gates).

When using improved HAs in the structure of a pyramidal 8-bit adder: $A_2 = \frac{8^2 + 8}{2} \times 3 = 108$ (gates).

That is, the hardware complexity of an 8-bit pyramidal multi-bit adder will be 1.7 times less with the use of improved HAs.

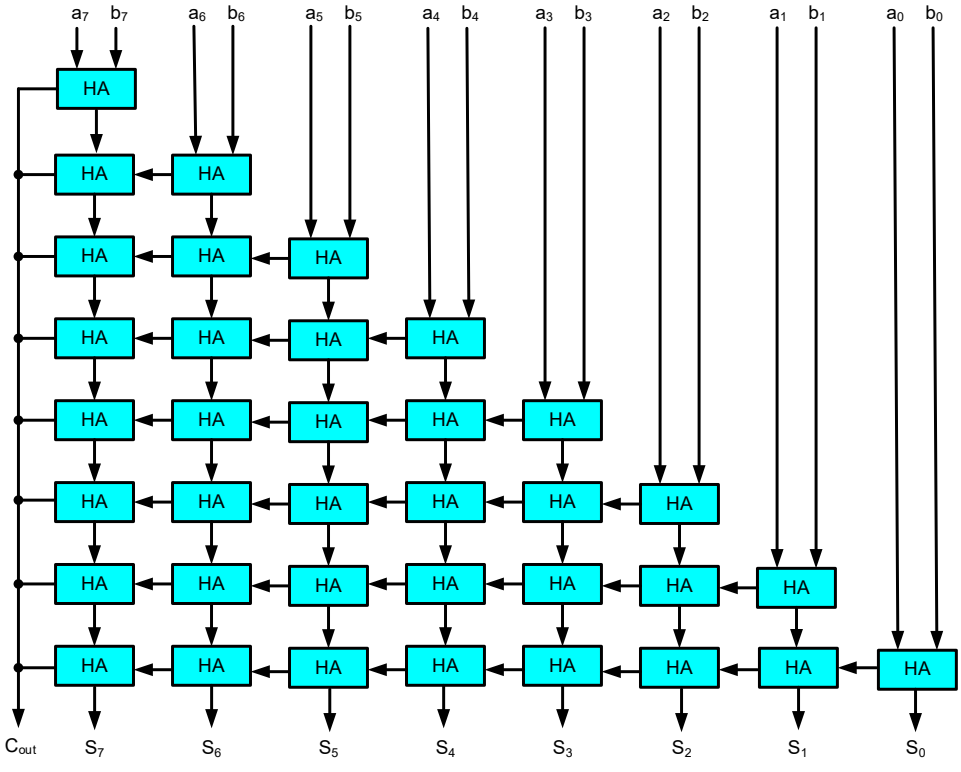

Fig. 2. Structure of an 8-bit pyramidal adder based on half adders

The time complexity of this type of pyramid adder is calculated from the following formula

$$t_{PA} = m \times t_{HA}, \quad (2)$$

where t_{HA} – time complexity of a half binary adder, m – input data bit size.

For example, at $m = 8$ with the use of classical HAs in the structure of a pyramidal 8-bit adder: $t_1 = 8 \times 3 = 24$ (micro cycles).

In the case of using an improved HA in the structure of a pyramidal 8-bit adder: $t_1 = 8 \times 1 = 8$ (micro cycles).

That is, the time complexity of an 8-bit pyramidal adder will be 3 times greater with the use of improved HAs.

Table 1 gives results for the hardware and time complexity of the pyramidal adder with different input data bit size.

The calculation of the hardware and time complexity of a pyramidal adder built on the known (classical) HA structure, which is reported in [6], was performed using formulae (1) and (2).

Our results for the hardware and time complexity of a pyramidal adder show that the use of the improved HA gives significant advantages in terms of its speed.

Fig. 3 shows dependence plot of the hardware complexity of pyramidal adders on their bit size when using classical and improved HAs.

The plot shows the linear dependence of the number of gates in a pyramidal adder on the values of its bit size (from $n = 8$ to $n = 256$).

Fig. 4 shows dependence plot of the time complexity of pyramidal adders on their bit size when using classical and improved HAs.

The plot shows the linear dependence of the number of micro cycles of the pyramidal adder on the values of its bit size (from $n = 8$ to $n = 128$).

Table 1

Results of calculating the hardware and time complexity of a pyramidal adder using classical and improved HAs

Bit capacity of the pyramid adder	Hardware complexity (known HAs), number of gates	Hardware complexity (improved HAs), number of gates	Time complexity (known HAs), number of micro cycles	Time complexity (improved HAs), number of micro cycles
8	180	108	24	8
16	680	408	48	16
32	2640	1584	96	32
64	10400	6240	192	64
128	41280	24768	384	128
256	164480	98688	768	256
512	656640	393984	1536	512
1024	2624000	1574400	3072	1024
2048	10490880	6294528	6144	2048

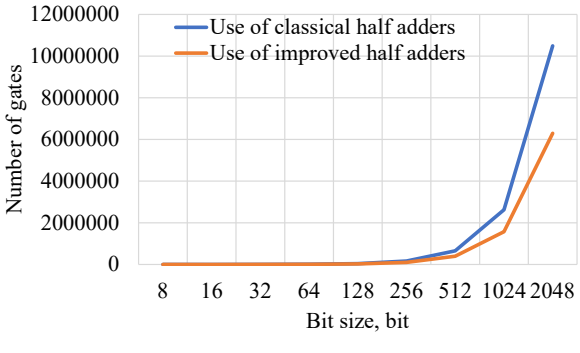

Fig. 3. Dependence plot of the hardware complexity of pyramidal multi-bit adders on their bit size when using different structures of half adders

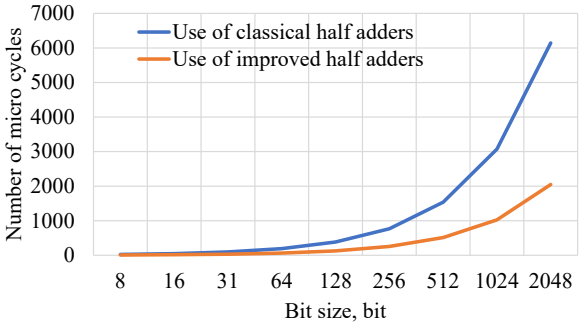

Fig. 4. Dependence plot of the time complexity of pyramidal multi-bit adders on their bit size when using different structures of half adders

The resulting plots in Fig. 3, 4 are constructed on the basis of the calculated values of the hardware and time complexity, which are given in Table 1.

In work [16], the authors proposed an improved structure of a full binary adder, which contains 6 gates (Fig. 5).

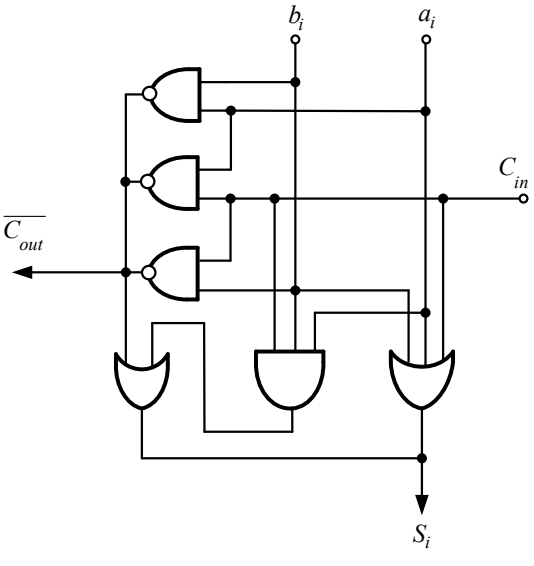

Fig. 5. Structure of the improved full binary adder with inverted carry-through output

The full binary adder contains information inputs a_i and b_i; 3 logical elements "AND-NOT"; logical element "AND"; 2 logical elements "OR"; input of the through-carry C_{in}; output of the sum S_i and inverse output $\overline{C_{out}}$ of the through-carry.

When applying to a_i, b_i and C_{in} inputs the following combinations of signals (000; 001; 010; 100) at the outputs of the logical elements "AND-NOT" and, accordingly, at the inverse output of the through-carry $\overline{C_{out}}$ of the adder, a bit "0" is formed. In the case of applying to the a_i, b_i and C_{in} inputs the following combinations of signals (011; 101; 110; 111) at the outputs of the logical elements "AND-NOT" and, accordingly, at the inverse output of the through-carry $\overline{C_{out}}$ of the adder, a bit "1" is formed. The received bit "0" or "1" from the output $\overline{C_{out}}$ is fed to the first input of the second logical element "OR", the output of which forms the corresponding bit "1" or "0" depending on the presence of a bit "1" at the output of the logical element "AND", when all its inputs have signals "1". Bit "0" at the output of the sum S_i is formed if all inputs of the first logical element "OR" have signals "0".

Such a structure of a full binary adder can be used as a component of multi-bit binary adders of the cascade type with output inverse carries.

Fig. 6 shows the structure of an 8-bit adder of the cascade type with output inverse carries.

The hardware complexity of a multi-bit cascade adder is calculated from the following formula

$$A_{KA} = n \times A_{FA}, \tag{3}$$

where A_{FA} is the hardware complexity of the full binary adder, n is the number of bits of input data.

For example, when $n = 8$ using classical FAs in the structure of an 8-bit cascade adder: $A_3 = 8 \times 11 = 88$ (gates).

When using an improved FA in the structure of an 8-bit cascade adder: $A_4 = 8 \times 6 = 48$ (gates).

That is, the hardware complexity of an 8-bit cascade adder will be 1.8 times less with the use of the improved FA.

The time complexity of a multi-bit cascade adder is calculated from the following formula

$$t_{KA} = n \times t_{FA}, \tag{4}$$

where t_{FA} is the time complexity of the full binary adder, n is the input data bit depth.

For example, when $n = 8$ using classical HAs in the structure of an 8-bit cascade adder: $t_3 = 8 \times 6 = 48$ (micro cycles).

In the case of using an improved FA in the structure of an 8-bit cascade adder: $t_4 = 8 \times 2 = 16$ (micro cycles).

That is, the time complexity of an 8-bit cascade adder will be 3 times greater with the use of the improved FA.

Table 2 gives results for the hardware and time complexity of the cascade adder with different input data bit depth.

The calculation of the hardware and time complexity of the cascade adder built on the known (classical) FA structure, which is reported in [5], was performed using formulae (3) and (4).

Our results for the hardware and time complexity of the cascade adder show that the use of the improved FA gives significant advantages in terms of its speed.

Fig. 7 shows dependence plot of the hardware complexity of a cascade adder on its bit size when using classical and improved HAs.

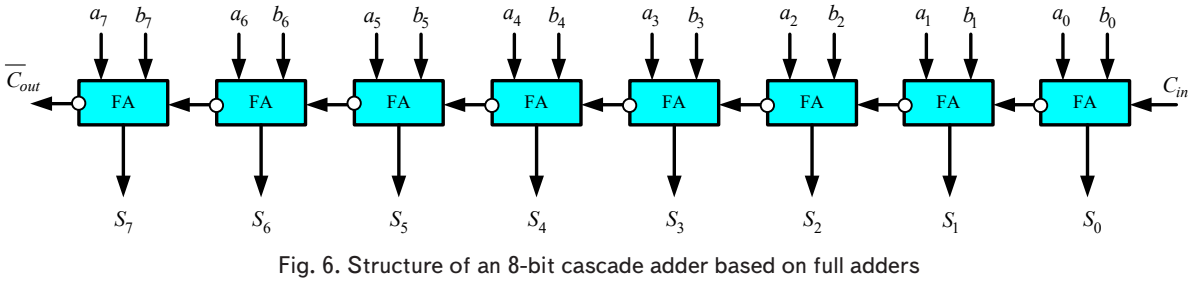

Fig. 6. Structure of an 8-bit cascade adder based on full adders

Table 2

Results of calculating the hardware and time complexity of a cascade adder using classical and improved FAs

Bit capacity of the cascade adder	Hardware complexity (known FAs), number of gates	Hardware complexity (improved FAs), number of gates	Time complexity (known FAs), number of micro cycles	Time complexity (improved FAs), number of micro cycles
8	88	48	48	16
16	176	96	96	32
32	352	192	192	64
64	704	384	384	128
128	1408	768	768	256
256	2816	1536	1536	512
512	5632	3072	3072	1024
1024	11264	6144	6144	2048
2048	22528	12288	12288	4096

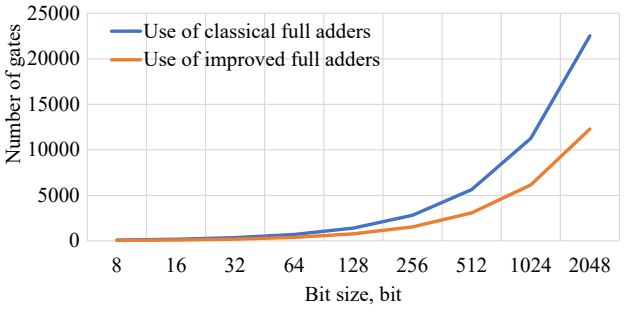

Fig. 7. Dependence plot of the hardware complexity of a cascade adder on its bit size when using different structures of full adders

The plot shows the linear dependence of the number of gates in a cascade adder on the values of its bit size (from $n = 8$ to $n = 128$).

Fig. 8 shows dependence plot of the time complexity of a cascade adder on its bit size when using classical and improved FAs.

The plot shows the linear dependence of the number of micro cycles of a cascade adder on the values of its bit size (from $n = 8$ to $n = 128$).

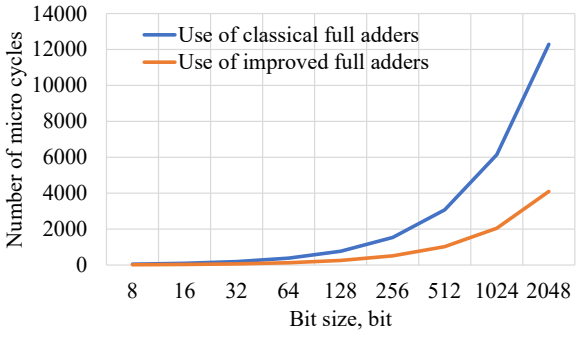

Fig. 8. Dependence plot of the time complexity of a cascade adder on its bit size when using different structures of full adders

The resulting plots in Fig. 7 and Fig. 8 are constructed on the basis of the calculated values of the hardware and time complexity, which are given in Table 2.

Fig. 9 proposes the structure of FA with paraphase inputs $(a_i, \overline{a_i}, b_i, \overline{b_i})$, and an inverse input of the input carry $(\overline{C_{in}})$, which forms the output signals of the sum $(\overline{S_i})$ and through carry $(\overline{C_{out}})$ in 1 micro cycle [17].

Such an adder contains input paraphase information channels and inverse input of the through-carry $(a_i, \overline{a_i}, b_i, \overline{b_i}, \overline{C_{in}})$; two logical elements "Exclusive AND", 4 logical elements "OR"; inverse outputs of the sum $(\overline{S_i})$ and through-carry $(\overline{C_{out}})$.

When applying to the inputs of the logical elements of the signals of the input paraphase channels and the input signal of the through-carry $(\overline{C_{in}})$, the inverse signals of the sum $(\overline{S})$ and through-carry $(\overline{C_{out}})$ are formed at the outputs of the device in 1 micro cycle.

The use of such a structure of FA makes it possible to maximize the speed of multi-bit adders of various types and matrix multipliers of multi-bit binary codes.

Fig. 10 shows the structure of an 8-bit cascade-type adder with paraphase inputs and inverse outputs of the sum and through-carry.

For example, at $n = 8$ using classical HAs in the structure of an 8-bit cascade adder with paraphase inputs and inverse sum and through-carry outputs: $t_5 = 8 \times 6 = 48$ (micro cycles).

In the case of using an improved FA (Fig. 9) in the structure of an 8-bit cascade adder with paraphase inputs and inverse sum and through-carry outputs: $t_6 = 8 \times 1 = 8$ (micro cycles).

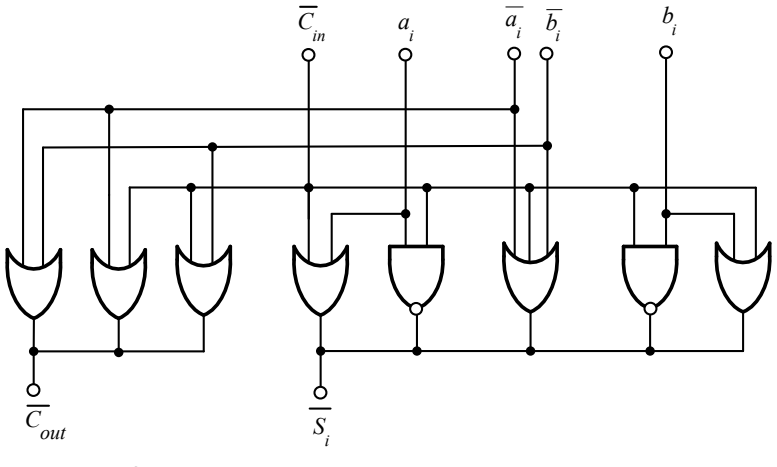

Fig. 9. Structure of an 8-valve single-bit adder with minimax characteristics with direct, inverse, and paraphase inputs and outputs

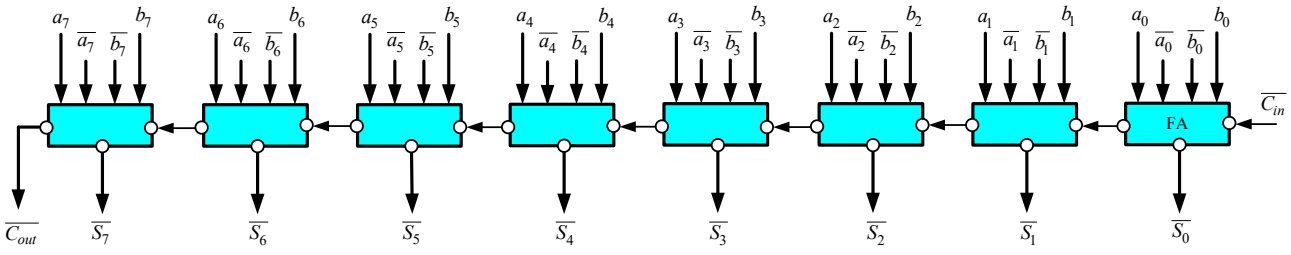

Fig. 10. Structure of an 8-bit cascade adder with paraphase inputs and inverted sum and carry-through outputs

That is, the time complexity of an 8-bit cascade adder with paraphase inputs and inverse sum and through-carry outputs will be 6 times greater with the use of such an improved FA.

Table 3 gives results for the time complexity of a cascade adder with paraphase inputs and inverse sum and through-carry outputs for different input data bits.

The calculation of the hardware and time complexity of such a cascade adder built on the known structure of FA with paraphase inputs and inverted outputs, which is reported in [9], was performed using formula (4).

Table 3

Results of calculating the time complexity of a cascade adder with paraphase inputs and inverse sum and carry-through outputs using classical and improved FAs

Bit depth of the cascade adder	Time complexity (known FAs), number of micro cycles	Time complexity (advanced FAs), number of micro cycles
8	48	8
16	96	16
32	192	32
64	384	64
128	768	128
256	1536	256
512	3072	512
1024	6144	1024
2048	12288	2048

The results in Table 3 for calculating the time complexity values of such a cascade adder show that the use of an improved FA gives significant advantages in terms of its speed.

Fig. 11 shows dependence plot of the time complexity of a cascade adder with paraphase inputs and inverted sum and through-carry outputs on its bit depth when using classical and improved FAs.

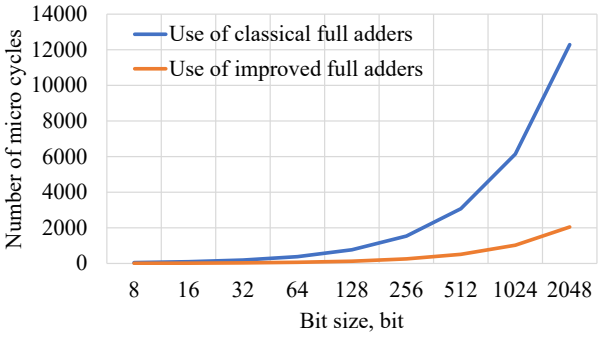

Fig. 11. Time complexity plot of a cascade adder with paraphase inputs and inverted sum and carry-through outputs

According to the above data (Fig. 11), it was found that the time complexity of the cascade adder, which contains improved full adders, has a linear dependence, which significantly affects the speed of the device.

5. 2. Design of the structure for a cascade n-bit fast adder and examination of its system characteristics

Fig. 12 proposes the structure of an improved cascade fast adder [18].

The fast adder includes: 1 – input $4n$-bit bus; 2 – $k = n / m$, m-bit adders with paraphase information inputs and inverse input/outputs of through-carry; 3 – output $(n + 1)$ – bit bus (Fig. 12).

The fast-carry adder (FCA) operates according to the following method: the input paraphase n-bit binary codes $(\overline{a}_{n-1},a_{n-1},...,\overline{a}_0,a_0)$ and $(\overline{b}_{n-1},b_{n-1},...,\overline{b}_0,b_0)$ of the input $4n$-bit bus 1 simultaneously arrive at the corresponding first information inputs of all m-bit adders 2. The second through-carry input of the first m-bit adder is connected to a logical '1', which for the accelerated-carry inverse logic module is the inverse input of the logical '0'. Fig. 13 shows the internal structure of a 4-bit adder with an accelerated-carry logic module.

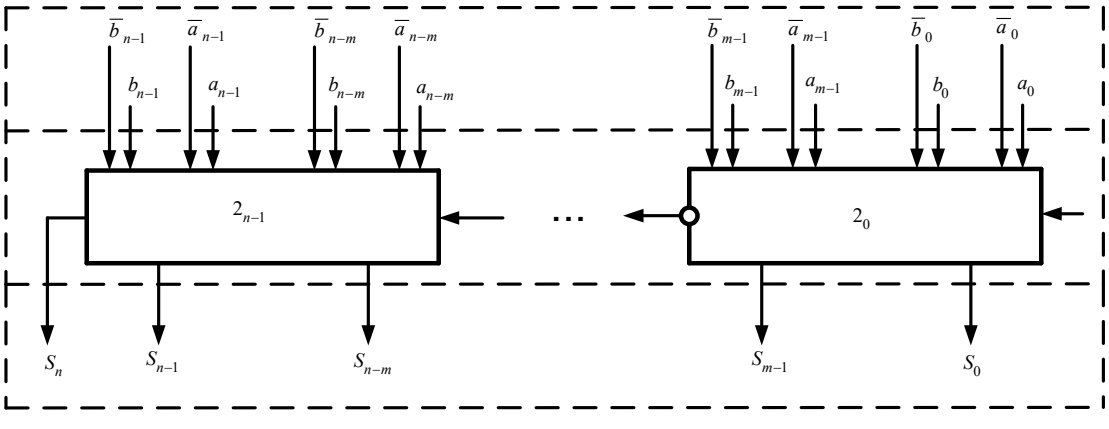

Fig. 12. Improved structure of a cascade n-bit fast adder

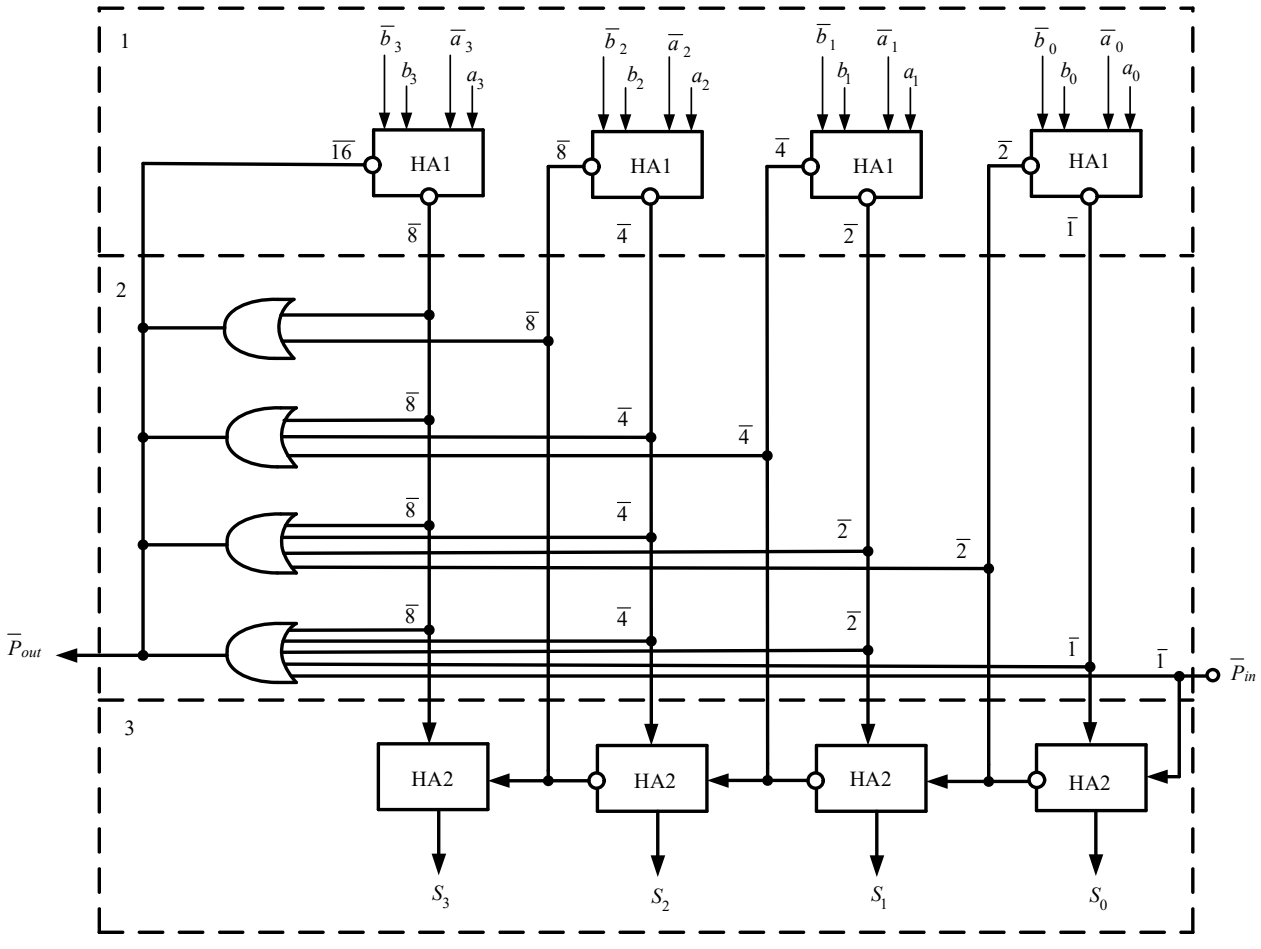

Fig. 13. Structure of a 4-bit adder with an accelerated-carry logic module

The logic module of the accelerated through-carry includes: n – half single-bit adders with paraphase inputs and inverted outputs; m-half single-bit adders with inverted inputs and direct sum outputs and m-logical elements "OR". The outputs of the logical elements "OR" are connected to each other and to the output of the through-carry, and the inputs are respectively connected to the inverted outputs of the half adders with paraphase inputs.

As a result, with a signal delay of 2 micro cycles at the output of the first m-bit adder, the output inverted signal of the through-carry is formed. At the same time, the output code of the 4-bit sum with a signal delay of m-micro cycles is formed at the information outputs of the first m-bit adder. The signals of accelerated through-carries are formed at the outputs of each subsequent m-bit adder with a delay of 1 micro cycle. The delay of the output n-th bit of the k-th m-bit adder, which additionally contains an output inverter and converts the inverted signal into a direct one, is formed with a signal delay of 2 micro cycles.

The hardware complexity of the improved cascade fast adder is calculated from the following formula

$$A_{CA} = k \times \left(m \times A_{HA1,2} + A_{LE}\right), \quad (5)$$

where k is the number of m-bit logic modules of accelerated carry, A_{HA1} is the hardware complexity of the first half binary adder; A_{HA} is the hardware complexity of the second half binary adder; m-bit of the logic module of accelerated carry.

For example, for $n = 32$ and $m = 4$, the hardware complexity of the improved fast adder will be: $FCA = 8 \times (8 \times 3 + 4)3 = 224$ (gates).

When using the known accelerated carry scheme [7] in the structure of a 32-bit cascade adder based on full binary adders: $A_{CA1} = 8 \times 52 = 416$ (gates).

That is, the hardware complexity of the proposed 32-bit cascade fast adder will be 1.9 times smaller compared to the known adder.

The time complexity of the improved cascade fast adder is calculated from the following formula

$$t_{CA} = k + m, \quad (6)$$

where k is the number of m-bit logic modules of accelerated carry, m-bit size of the logic module of accelerated carry.

For example, for $n = 32$ and $m = 4$, the time complexity of the improved fast adder will be: $t_{CA} = 8 + 4 = 12$ (micro cycles).

When using the known accelerated carry scheme [9] in the structure of a 32-bit cascade adder based on full binary adders: $t_{CA1} = 8 + 14 = 22$ (micro cycles).

That is, the time complexity of the proposed 32-bit cascade fast adder will be 1.8 times greater in comparison with the known adder of this type.

Table 4 gives results for the hardware and time complexity of a cascade fast adder for different input data bits.

The calculation of the hardware and time complexity of a cascade fast adder built on the known structure of FA with

accelerated carry, which is reported in [10], was performed using formulae (5) and (6).

Table 4

Results of calculating the hardware and time complexity of a cascade fast adder when using known and improved FCAs

Bit capacity of a cascade fast adder	Hardware complexity (known FCA scheme), number of gates	Hardware complexity (improved FCA scheme), number of gates	Time complexity (known FCA scheme), number of micro cycles	Time complexity (improved FCA scheme), number of micro cycles
8	104	56	10	6
16	208	112	14	8
32	416	224	22	12
64	832	448	38	20
128	1664	896	70	36
256	3328	1792	134	68
512	6656	3584	262	132
1024	13312	7168	518	230
2048	26624	14336	1030	516

Our results for the hardware and time complexity of a cascade fast adder show that the use of the improved FCA scheme gives advantages in terms of its speed.

Fig. 14 shows dependence plot of the hardware complexity of the improved cascade fast adder on its bit size when compared with a known adder of this type.

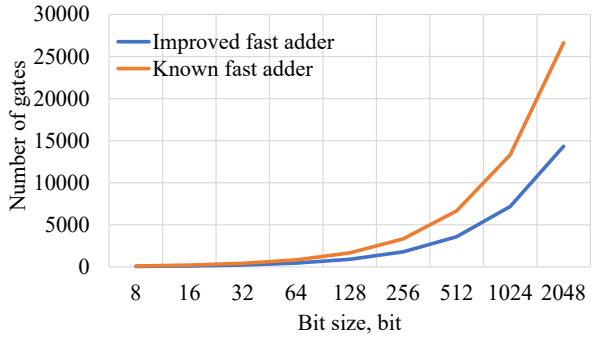

Fig. 14. Dependence plot of the hardware complexity of an improved cascade fast adder on its bit size when compared with a known adder

Fig. 15 shows dependence plot of the time complexity of an improved cascade fast adder on its bit size when compared with a known adder of this type.

The time complexity of the proposed fast adder has a partially exponential dependence, which significantly increases the speed of such a device compared to the known one [10].

The resulting plots in Fig. 14, 15 are constructed on the basis of the calculated values of hardware and time complexity, which are given in Table 4.

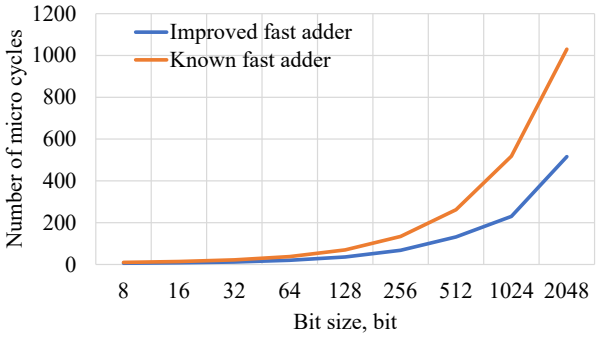

Fig. 15. Dependence plot of the time complexity of an improved cascade adder on its bit size when compared with a known adder

5. 3. Modeling and synthesis of a multi-bit cascade-type adder on programmable logic integrated circuits

The design and modeling of the operation of a multi-bit cascade adder was carried out using the VHDL hardware description language in the Active-HDL integrated environment [19].

Fig. 16 shows a diagram of the functional simulation of a 64-bit cascade adder built on the basis of improved full single-bit adders.

This diagram shows the supply of 64-bit input data (a and b) and the generation of the sum result at the 64-bit output S.

The synthesis of the designed VHDL model of the multi-bit cascade adder was carried out on the Xilinx-AMD FPGA (Artix-7 family, xc7a50tcsg324-1 crystal).

Fig. 17 shows the implementation of a 64-bit cascade adder on the xc7a50tcsg324-1 crystal from the Artix-7 family.

During the implementation of the 64-bit cascade adder design on the FPGA element base, look-up tables (LUTs) were used, which are elements of the configurable logic block (SLICE) of the selected crystal.

Table 5 gives synthesis results for the implemented 64-bit pyramidal and cascade adders on the Xilinx FPGA from the Artix-7 family using classical full and half adders.

Table 6 gives results from the synthesis of implemented 64-bit pyramidal and cascade adders on Xilinx FPGAs of the Artix-7 family using improved full and half adders.

According to the synthesis results (Tables 5, 6), the minimum hardware costs in terms of quantity (LUT) are a multi-bit cascade adder built on the basis of an improved full single-bit adder with an inverse output of a through-carry. The best performance is achieved by a pyramidal adder based on HA and a cascade adder based on paraphase FAs.

Table 7 gives synthesis results for the implemented known and improved 64-bit fast adder on the Xilinx FPGA from the Artix-7 family.

Analysis of our results (Table 7) reveals that the improved fast adder during synthesis on FPGA (crystal xc7a50tcsg324-1) occupies 77 (LUTs) while the operating frequency of such an adder is 374 MHz. The obtained system characteristics improve the speed and hardware complexity of such an adder by approximately 2 times compared to the known implementation.

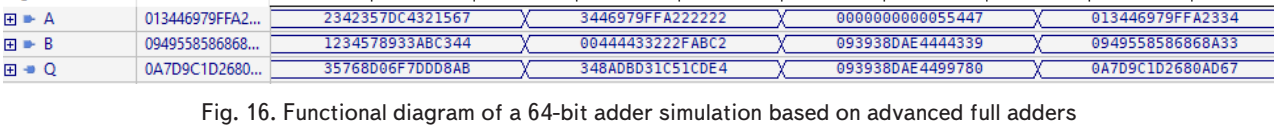

Fig. 16. Functional diagram of a 64-bit adder simulation based on advanced full adders

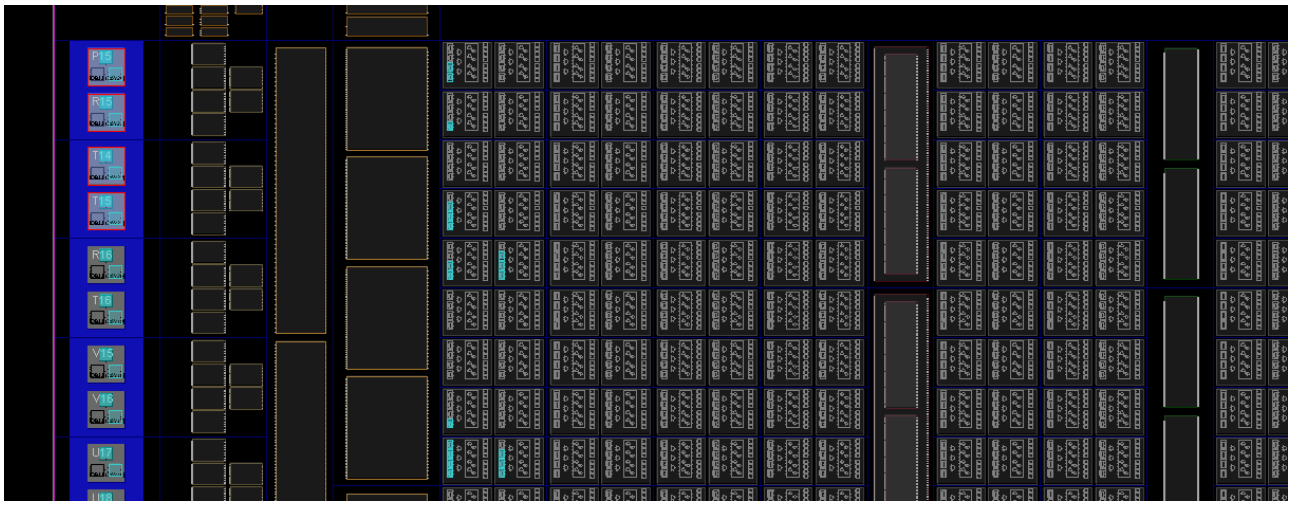

Fig. 17. Internal part of the xc7a50tcsg324-1 crystal when implementing a 64-bit cascade adder

Table 5

Results of synthesizing 64-bit pyramidal and cascade adders on FPGA using known full and half adders

No.	Multi-bit adder type	Number of look-up tables (LUTs)	Clock frequency, (MHz)
1	Pyramidal adder based on HA	421	115
2	Cascade adder based on FA	194	62
3	Cascade adder based on paraphase FAs	241	57

Table 6

Results of synthesizing 64-bit pyramidal and cascade adders on FPGA using improved full and half adders

No.	Multi-bit adder type	Number of look-up tables (LUTs)	Clock frequency, (MHz)
1	Pyramidal adder based on HA	245	334
2	Cascade adder based on FA	108	172
3	Cascade adder based on paraphase FAs	145	325

Table 7

Results of synthesizing a well-known and the improved 64-bit fast adder on FPGA

No.	Fast-carry adder type	Number of look-up tables (LUTs)	Clock frequency, (MHz)
1	Well-known fast-carry adder	144	210
2	Improved fast-carry adder	77	374

6. Discussion of the results of investigating multi-bit binary adders of various types

The use of an improved half binary adder (Fig. 1) in the structure of a multi-bit binary adder of the pyramidal type allowed us to improve its system characteristics (Table 1). In particular, when calculating the hardware complexity of such a binary adder using formula (1), it was possible to reduce the hardware complexity by approximately 1.7 times compared to known implementations [3, 6]. Analysis of the dependence of the hardware complexity of the pyramidal adder on its bit depth (Fig. 3) reveals that with a bit depth of $n = 512$ and more, a significant saving of logical elements is observed, which is measured in hundreds of thousands.

When calculating the time complexity of such a binary adder using formula (2), it was possible to increase its speed by approximately 3 times compared to known implementations [3, 6]. Analysis of the dependence of the time complexity of the pyramidal adder on its bit size (Fig. 4) reveals that with a bit size of $n = 512$ and more, the use of an improved element base provides an advantage in terms of the speed of such a device. In particular, the time spent on performing the addition operation is reduced by 1000 micro cycles compared to known implementations.

The obtained practical results regarding the synthesis of a multi-bit pyramidal adder with an improved element base on FPGA (Table 6) in comparison with known implementations on FPGA (Table 5) confirmed the results of our theoretical calculations. This is explained by the approximation of the internal structure of the FPGA to the structures of multi-bit adders at the level of register carry.

The use of an improved full binary adder (Fig. 5) in the structure of a multi-bit binary adder of the cascade type allowed us to improve its system characteristics (Table 2). In particular, when calculating the hardware complexity of such a binary adder using formula (3), it was possible to reduce the hardware complexity by approximately 1.8 times compared with known implementations [5, 9]. Analysis of the dependence of the hardware complexity of the cascade adder on its bit size (Fig. 7) reveals that with a bit size of $n = 256$ and more, a significant saving of logical elements is observed, which is measured in thousands of gates. When calculating the time complexity of such a binary adder using formula (4), it was possible to increase its speed by approximately 3 times compared to known implementations [5, 9]. Analysis of the dependence of the time complexity of the cascade adder on its bit size (Fig. 8) reveals that with a bit size of $n = 256$ and more, a significant advantage is observed in terms of the speed of such a device. In particular, multi-bit cascade adders on an improved element base have a speed that is 1000 micro cycles lower compared to known implementations [9].

In the case of using an improved full binary adder (Fig. 9) in the structure of a multi-bit cascade adder with paraphase

inputs and inverse sum and carry-through outputs, it was possible to increase its speed by approximately 6 times compared to known implementations (Table 3).

Analysis of the dependence of the time complexity of such a cascade adder on its bit depth (Fig. 11) reveals that its speed as it increases has a linear dependence. In the case of using classical full single-bit adders in the structure of such an adder, their speed as it increases has an exponential dependence [7].

Our practical results from the synthesis of a multi-bit cascade adder with improved full single-bit adders on FPGA (Table 6) in comparison with known implementations on FPGA (Table 5) confirmed the results of our theoretical calculations.

When calculating the hardware and time complexity of the proposed fast adder (Fig. 12) using formula (5) and formula (6), it was possible to reduce its hardware complexity by 1.9 times, and its speed by 1.8 times compared to known implementations (Table 4). Analysis of the dependence of hardware and time complexity on the bit size of such adders (Fig. 14, 15) reveals that the system characteristics have an exponential dependence.

Our practical results from the synthesis of fast adders on FPGA (Table 7) have confirmed the results of theoretical calculations.

The limitations of these results are in the approximation at the gate level to the minimax system characteristics of hardware and time complexity.

The disadvantages of our research include the fact that not all system characteristics of the structures were taken into account; in further research, we plan to advance the analysis and study of the structural and functional complexity of such multi-bit adders.

Future studies imply further microelectronic research into single-bit adder structures based on semiconductor technologies [7, 10], which make it possible to significantly enhance electrical characteristics.

7. Conclusions

1. The use of various types of improved structures of full and half binary adders built on the basis of the "Exclusive AND" logic element in multi-bit adders has been justified. This allowed us to significantly accelerate the formation of sum and through-carry signals in the minimum time, as well as reduce the number of gates in such adders. It was established that the speed of multi-bit adders using improved HAs and FAs increases by 6 times compared to the classical element base.

2. The structure of a cascade n-bit fast adder for data summation in the Rademacher number-theoretic basis has been designed and its system characteristics have been determined. It was established that the time complexity of the improved fast adder is 1.8 times higher compared to known implementations.

3. A 64-bit structure of a multi-bit adder of cascade type for summing Rademacher basis numbers has been designed by using the hardware description language VHDL; its modeling and synthesis on FPGA were carried out. When synthesizing multi-bit adders on FPGA, it was found that the use of improved FAs and HAs makes it possible to reduce the hardware complexity of such adders by approximately 2 times and increase their speed by 6 times compared to known implementations. The use of improved components in the structures of multi-bit adders allows for their effective application as part of vector and scalar processors in supercomputers, in particular arithmetic and logical devices (FPGAs), coprocessors, as well as hardware accelerators for performing operations on binary numbers.

Conflicts of interest

The authors declare that they have no conflicts of interest in relation to the current study, including financial, personal, authorship, or any other, that could affect the study, as well as the results reported in this paper.

Funding

The study was conducted without financial support.

Data availability

All data are available, either in numerical or graphical form, in the main text of the manuscript.

Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technologies when creating the current work.

Authors' contributions

Yaroslav Nykolaychuk: Conceptualization, Methodology, Supervision. **Igor Pitukh**: Formal analysis, Investigation. **Volodymyr Hryha**: Software, Validation, Visualization.

References

1. Heath, R. W. (2017). Introduction to Wireless Digital Communication: A Signal Processing Perspective. Pearson, 464. Available at: https://www.pearson.com/en-us/subject-catalog/p/introduction-to-wireless-digital-communication-a-signal-processingperspective/P200000009126/9780134431796
2. Allemang, R., Avitabile, P. (Eds.) (2022). Handbook of Experimental Structural Dynamics. Springer. https://doi.org/10.1007/978-1-4614-4547-0
3. Anand Kumar, A. (2016). Fundamentals of Digital Circuits. PHI. Available at: https://books.google.co.in/books?id=CyavDAAAQBAJ&printsec=copyright#v=onepage&q&f=false
4. Kusuma, H S., Kalpana, H. M., Ravi, H. K (2025). Design and Comparative Analysis of Half and Full Adders Using CMOS and Pass Transistor Logic Styles. 2025 3rd International Conference on Smart Systems for Applications in Electrical Sciences (ICSSES), 1–6. https://doi.org/10.1109/icsses64899.2025.11010034

5. Kumar, P., Bhandari, N. S., Bhargav, L., Rathi, R.,Yadav, S. C. (2017). Design of low power and area efficient half adder using pass transistor and comparison of various performance parameters. 2017 International Conference on Computing, Communication and Automation (ICCCA), 1477–1482. https://doi.org/10.1109/ccaa.2017.8230033
6. Nykolaichuka, Ya. M. (Ed.) (2017). Spetsializovani kompiuterni tekhnolohiyi v informatytsi. Ternopil: Beskydy, 913. Available at: https://scs.wunu.edu.ua/?p=353
7. John, V., Sam, S., Radha, S., Paul, P. S., Samuel, J. (2020). Design of a power-efficient Kogge–Stone adder by exploring new OR gate in 45nm CMOS process. Circuit World, 46 (4), 257–269. https://doi.org/10.1108/cw-12-2018-0104
8. Yadav, R., Sura, A., Dahiya, S. (2021). Half adder design using various technologies and comparison of. International Journal of Engineering Applied Sciences and Technology, 6 (2). https://doi.org/10.33564/ijeast.2021.v06i02.016
9. Jhamb, Dr. M., Kumar, M., Vishal (2020). Full Adder for Low Power Applications. International Journal of Innovative Technology and Exploring Engineering, 9 (4), 2612–2687. https://doi.org/10.35940/ijitee.d1846.029420
10. Hameed, A. S., Kathem, M. J. (2021). High speed modified carry save adder using a structure of multiplexers. International Journal of Electrical and Computer Engineering (IJECE), 11 (2), 1591. https://doi.org/10.11591/ijece.v11i2.pp1591-1598
11. Davletova, A. Ya., Nykolaichuk, Ya. M. (2017). Pat. No. 115861 UA. Odnorozriadnyi nepovni sumator. No. u201612463; declareted: 07.12.2016, published: 25.04.2017. Available at: https://sis.nipo.gov.ua/uk/search/detail/805118/
12. Hryha, V., Dzundza, B., Melnychuk, S., Manuliak, I., Terletsky, A., Deichakivskyi, M. (2023). Design of various operating devices for sorting binary data. Eastern-European Journal of Enterprise Technologies, 4 (4 (124)), 6–18. https://doi.org/10.15587/1729-4061.2023.285997
13. Dunets, R., Gryga, V. (2015). Spatio-temporal synthesis of transformation matrix of reverse fast cosine transformation. The Experience of Designing and Application of CAD Systems in Microelectronics, 45–49. https://doi.org/10.1109/cadsm.2015.7230792
14. Gryga, V., Kolosov, I., Danyluk, O. (2016). The development of a fast iterative algorithm structure of cosine transform. 2016 13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET), 506–509. https://doi.org/10.1109/tcset.2016.7452100
15. Gryga, V., Dzundza, B., Dadiak, I., Nykolaichuk, Y. (2018). Research and implementation of hardware algorithms for multiplying binary numbers. 2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET), 1277–1281. https://doi.org/10.1109/tcset.2018.8336427
16. Nykolaichuk, Ya. M., Hryha, V. M., Nykolaichuk, L. M., Pitukh, I. R. (2021). Pat. No. 147277 UA. Povnyi odnorozriadnyi sumator. No. a202006409; declareted: 05.10.2020, published: 06.04.2022. Available at: https://sis.nipo.gov.ua/en/search/detail/1457390/
17. Nykolaichuk, Ya. M. (2023). Pat. No. 153527 UA. Povnyi odnorozriadnyi sumator. No. u202203361; declareted: 13.09.2022, published: 19.07.2023. Available at: https://sis.nipo.gov.ua/en/search/detail/1748918/
18. Nykolaichuk, Ya. M., Vozna, N. Ya., Hryha, V. M., Volynskyi, O. I. (2023). Pat. No. 150331 UA. Sumator z pryskorenym perenosom. No. u202104275; declareted: 21.07.2021, published: 02.02.2022. Available at: https://sis.nipo.gov.ua/en/search/detail/1676002/
19. Li, H., Ye, W. (2016). Efficient implementation of FPGA based on Vivado High Level Synthesis. 2016 2nd IEEE International Conference on Computer and Communications (ICCC), 2810–2813. https://doi.org/10.1109/compcomm.2016.7925210