Analysis and Synthesis of Single-Bit Adders for Multi-Bit Adders with Sequential Transfers †

Sakhbay Tynymbayev 1, Assel Mukasheva 2,*, Kuanyshbek Ibragimov 3, Adil Mukhamedgali 2, Gani Sergazin 4 and Teodor Iliev 5

1 International University of Information Technology, Almaty 050013, Kazakhstan; s.tynymbayev@iitu.edu.kz
2 School of Information Technology and Engineering, Kazakh-British Technical University, Almaty 050000, Kazakhstan; a.mukhamedgali@kbtu.kz
3 Kazakh National University named after Al-Farabi, Almaty 050040, Kazakhstan; kuanyshbek-ibragimov@mail.ru
4 Academy of Logistics and Transport, Almaty 050012, Kazakhstan; g.balbayev@gmail.com
5 Department of Telecommunications, University of Ruse, 7004 Ruse, Bulgaria; tiliev@uni-ruse.bg

* Correspondence: mukashevascience@gmail.com
† Presented at the International Conference on Electronics, Engineering Physics and Earth Science (EEPES’24), Kavala, Greece, 19–21 June 2024.

Abstract: This paper provides an analysis of existing single-digit binary adders from the point of view of their implementation on fans built based on metal-oxide-semiconductor field-effect transistor—MOSFETs. The synthesis of a single-digit adder with a conditional sum is carried out. The considered adders are compared in terms of speed and hardware complexity (by the number of MOSFETs). Adders perform arithmetic operations on numbers. In combination with other logical operations, adders are the core of the circuits of arithmetic logic devices that implement several different operations; they are an integral part of different processors. The most important parameters of adders are their hardware complexity and performance; therefore, many options for single-bit and multi-bit connectors with serial, parallel and combined transmissions have been developed. In the final part, a scheme of a multi-bit adder with consecutive transfers on adders with a conditional sum is given. An example of performing addition operations is given.

Keywords: single-bit adder; multi-bit adder with consecutive transfers; conditional sum adder; multiplexer

1. Introduction

Adders perform arithmetic addition and subtraction of numbers. Together with other logical operations, adders are the core of the circuits of arithmetic logic devices that implement a number of various operations (multiplication, division, etc.), which are an indispensable part of various processors.

Important parameters of adders are their hardware complexity and performance; therefore, many variants of single-bit and multi-bit adders with serial, parallel and combined transfers have been developed [1].

The listed adders are synthesized based on the truth table. The analytical expressions of the sum \( S_i \) and transfer functions \( C_i \) have the form:

\[
S_i = \pi_i \overline{b_i} C_{i-1} \lor \rho_i b_i C_{i-1} \lor a_i \overline{b_i} C_{i-1} \lor a_i b_i C_{i-1};
\]

\[
C_i = a_i b_i \lor a_i C_{i-1} \lor b_i \overline{C}_{i-1},
\]

where \( \pi_i \) and \( \rho_i \) are the \( i \)-th digits of the numbers \( A \) and \( B \); \( C_{i-1} \) — transfer from the junior category.
The formula can be reproduced on various sets of logic elements, such as AND-NOT, OR-NOT, “exclusive OR”, etc. At the same time, the hardware’s complexity and performance may be different.

2. Materials and Methods

The analysis of single-digit adders on elements is discussed below:

- two-stage logic AND-OR-NOT [1];
- two “exclusive OR” valves and OR-NOT and AND-NOT circuits [2];
- three-way valve “exclusive OR” and circuits AND-NOT;
- on valves “exclusive OR” and multiplexers, an adder with a conditional sum (conditional-sum addition CSA) is implemented. Before determining the hardware complexity (the number of MOSFETs) and performance (the number of logic elements on which the signal is delayed), the circuits of the analyzed adders are depicted as logic gates on the MOSFET. Consider the analysis of an adder built on the basis of the two-stage logic AND-OR-NOT [1,3–5].

The circuit of such an adder is shown in Figure 1, and Figure 2 shows the circuit of this adder on the logic circuits of a MOSFET [6–8].

![Functional diagram of the adder](image1)

![Logic circuit on a MOSFET](image2)

Figure 1. Functional diagram of the adder.

Figure 2 shows its logic circuit on a MOSFET.

The analytical formula of the analyzed adder has the form:

\[
\begin{align*}
C_i &= (a_i b_i \vee a_i C_{i-1} \vee b_i C_{i-1}); \\
S_i &= C_{j-1} (a_i \vee b_i \vee C_{j-1}) \vee a_i b_i C_{i-1}.
\end{align*}
\] (2)

Figure 1 shows the functional diagram of the adder constructed according to Formula (2). Figure 2 shows its logic circuit on a MOSFET.

Figure 2 shows that the adder consists of six AND-NOT circuits with two inputs each. To implement them, \(6 \times 4 = 24\) transistors will be required. It has nine circuits of NOT \((2 \times 9 = 18)\) transistors and a gate AND-NOT for 3 inputs (6 transistors), a gate OR-NOT for 3 inputs (6 transistors) and a gate OR-NOT for 4 inputs (8 transistors). In total, 62 MOSFETs will be required to implement the adder.

The time of formation of the transfer \(t_{ci} = 4 \tau_{le}\).

The time of formation of the sum \(S_i = 7 \tau_{le}\).
The second single-digit adder \([2,3,9]\), which is subject to analysis, refers to the following formula:

\[
S_i = C_{i-1} \oplus a_i \oplus b_i
C_i = a_i b_i a_{i-1} b_{i-1}.
\] (3)

The functional scheme of the \(C_i\) transfer is shown in Figure 3, which consists of three two-input and one three-input AND-NOT circuit. To implement them, we will need \((4 \times 3) + 6 = 18\) transistors. The time of transfer formation \(t_{\text{ci}} = 2 \tau_{\text{le}}\).

\[
S_i = \overline{C_i} \lor a_i \lor \overline{b_i};
S_i = S_{i}^1 C_i \lor S_{i}^1 \overline{C_i}.\] (4)
The truth table for the sum of $S_i$ variables. In our case, $a_i$, $b_i$ and $C_{i-1}$ come in as input variables. Then, the truth table for $S_i$ transistors. The second adder can be built on the basis of an adder modulo two of three input AND-NOT circuits, which will require $(4 \times 2) + (6 \times 4) = 30$ transistors.

Table 1. The truth table for the sum of $S_i$. 

<table>
<thead>
<tr>
<th>$C_{i-1}$</th>
<th>$a_i$</th>
<th>$b_i$</th>
<th>$S_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

From this table we have:

$$ S_i = C_{i-1} \overline{a} b_i \lor C_{i-1} a \overline{b}_i \lor C_{i-1} \overline{a} b_i \lor C_{i-1} a \overline{b}_i. $$  \hspace{1cm} \text{(5)}

To calculate $S_i$, the Formula (5) is transformed as follows:

$$ S_i = \overline{(C_{i-1} \overline{a} b_i)(C_{i-1} a \overline{b}_i)}(C_{i-1} \overline{a} b_i)(C_{i-1} a \overline{b}_i). $$  \hspace{1cm} \text{(6)}

The functional scheme for the sum of $S_i$ constructed according to the Formula (6) is shown in Figure 5.

According to Figure 5, it is not difficult to calculate that 38 MOSFETs will be required for the implementation of the $S_i$ adder.

According to Figure 3, to calculate $C_i$, 18 transistors will be required for a total of 56 transistors. For the formation of $S_i$, a delay on 3 logical elements will be required, i.e., $t_{si} = 3 \tau_{le}$.

Recently, the construction of so-called conditional sum adders (conditional-sum addition, CSA) has been vigorously discussed [10–14].

The adder with a conditional sum is also built according to the truth table. Table 2 shows a modified table of the truth of the adder.
According to Figure 5, it is not difficult to calculate that 38 MOSFETs will be required for the implementation of the Si adder.

To calculate $C_i$, 18 transistors will be required for a total of 56 transistors. For the formation of $S_i$, a delay on 3 logical elements will be required, i.e., $t_{si} = 3\tau_{le}$.

Recently, the construction of so-called conditional sum adders (conditional-sum addition CSA) has been vigorously discussed [10–14].

The adder with a conditional sum is also built according to the truth table. Table 2 shows a modified table of the truth of the adder.

<table>
<thead>
<tr>
<th>$C_{i-1}$</th>
<th>$a_i$</th>
<th>$b_i$</th>
<th>$S_{i0}$</th>
<th>$S_{i1}$</th>
<th>$C_i^0$</th>
<th>$C_i^1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>-</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>-</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>-</td>
<td>0</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>-</td>
<td>0</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>1</td>
</tr>
</tbody>
</table>

From the first part of Table 2, where $C_{i-1} = 0$, we have

$$S_{i0} = a_ib_i \lor \overline{a_i}\overline{b_i}; \quad (7)$$

$$C_{i}^0 = a_ib_i. \quad (8)$$

From the second part of Table 2, where $C_{i-1} = 0$, we have

$$S_{i1} = \overline{a_i}\overline{b_i} \lor a_i b_i, \quad (9)$$

$$C_{i1} = \overline{a_i}b_i \lor a_i\overline{b_i} \lor a_ib_i = a_i \lor b_i \quad (10)$$

From this table it is also easy to see that

$$S_i = \overline{S_{i0}} \quad (11)$$

To construct the sum of $S_{i0}$ on the fans AND-NOT, OR-NOT and NOT, Formula (7) must be reduced to the form:

$$S_{i0} = (a_i \lor b_i)\left(\overline{a_i}b_i\right) = (a_i \lor b_i)\left(\overline{a_i}b_i\right) = (a_i \lor b_i)\left(\overline{a_i} \lor b_i\right) \quad (12)$$
According to the Formulas (8), (10), (11) and (12) it is possible to put the functional sum of the adder on the conditional sum. At the same time, the functional circuit must be supplemented with multiplexers that are controlled by transferring from the lower digit to a selection of one of $C_i^0$ and $C_i^1$ and one of $S_i^0$ and $S_i^1$, forming a transfer to the next digit $C_{i+1}$ and the value of the sum $S_i$.

Taking into account the above, the functional scheme of the adder has the form as the scheme shown in Figure 6.

![Figure 6](image-url)

**Figure 6.** Functional diagram of the conditional sum adder.

The adder consists of two circuits, OR-NOT$_1$ and AND-NOT$_2$, three inverters, INV$_1$, INV$_2$ and INV$_3$, and a MUX$_1$ and MUX$_2$ multiplexer. At the inputs of the circuits OR-NOT$_1$ and AND-NOT$_2$, are the i-th digits $a_i$ and $b_i$ of the numbers A and B. At the output of the valve AND-NOT$_2$, the sum of $S_i^0$ is formed, which is fed to the input “0” of the MUX$_1$ multiplexer and to the input of the inverter INV$_2$, and at its output, the sum of $S_i$ is formed, which enters the input “1” MUX$_1$ multiplexer. From the output of the inverter INV$_1$, transfer $C_i^0 = \overline{a_i} \overline{b_i} = a_i \overline{b_i}$ enters the input “0” of the MUX$_2$ multiplexer. From the output of the inverter INV$_3$ transfer:

$$C_i^1 = \overline{a_i} \vee \overline{b_i} = a_i \lor b_i$$

(13)

It is fed to the input “1” of the MUX$_2$ multiplexer. The control inputs of the multiplexers are transferred from the lower-order $C_{i-1}$.

The adder works as follows. After the bits $a_i$ and $b_i$ are fed to the inputs of the gates OR-NOT$_1$ and AND-NOT$_1$, the value of $S_i^0$ is formed at the output of the circuit OR-NOT$_2$, and the value of the sum $S_i^1$ is formed at the output of the inverter INV$_2$. At the same time, the transfer value $C_i^0$ is formed at the output of the inverter INV$_1$ and the transfer value $C_i^1$ is formed at the output of the inverter INV$_3$.

After feeding the transfers $C_i^0$ and $C_i^1$ and the sums $S_i^0$ and $S_i^1$ at the input of the corresponding multiplexers by transferring $C_{i-1}$, which is fed to the control inputs of the multiplexers simultaneously, the value of the sum Si is formed at the output of the MUX$_1$ multiplexer and at the same time, the transfer $C_1$ is formed at the output of the MUX$_2$ multiplexer, which is fed to the inputs of the multiplexers of the next highest digit.
3. Result and Discussion

Figure 7 shows the functional diagram of the MUX₁ multiplexer on the MOSFET and the MUX₁ operation Table 3.

![Figure 7. Functional diagram of MUX₁ multiplexer on the MOSFET.](image)

**Table 3. Modified adder truth table.**

<table>
<thead>
<tr>
<th>C_{i-1}</th>
<th>S_{i0}</th>
<th>S_{i0}</th>
<th>S_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 8 shows the functional diagram of the MUX₂ multiplexer and its operation table (Figure 8).

![Figure 8. MUX₂ multiplexer function diagram.](image)

**Table 4. MUX₂ multiplexer truth table.**

<table>
<thead>
<tr>
<th>C_{i-1}</th>
<th>a_i</th>
<th>b_i</th>
<th>C_{i0}</th>
<th>C_{i1}</th>
<th>C_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

To build a CSA adder, an AND-NOT gate is required for two inputs (4 transistors), two OR-NOT gates for 2 inputs (2 × 4 = 8 transistors), 3 inverters INV₁/INV₃
(3 × 2 = 6 transistors) and 2 multiplexers (2 × 6 = 12 transistors). Everything will be required for a total of 30 transistors (4 + 8 + 6 + 12 = 30 transistors). The $S_i$ formation time is $4 \tau_{le}$ and the delay time on multiplexers is $2 \tau_{le}$.

Table 4 shows an example of a truth table for MUX2, which is shown in Figure 8.

Table 4. MUX2 multiplexer truth table.

<table>
<thead>
<tr>
<th>$C_{i-1}$</th>
<th>$a_i$</th>
<th>$b_i$</th>
<th>$C_i^0$</th>
<th>$C_i^1$</th>
<th>$C_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

The order of operation of the bit adder with a conditional sum is given in Table 5.

Table 5. The order of operation of the adder with a conditional sum.

<table>
<thead>
<tr>
<th>№</th>
<th>$C_{i-1}$</th>
<th>$a_i$</th>
<th>$b_i$</th>
<th>$S_i^0$</th>
<th>$S_i^1$</th>
<th>$C_i^0$</th>
<th>$C_i^1$</th>
<th>$S_i$</th>
<th>$C_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

From Figure 8, it is not difficult to calculate the number of transistors. To form the transfers of $C_i^0$ and $C_i^1$ and the sums of $S_i^0$ and $S_i^1$, 18 transistors are required. To build two multiplexers, 12 transistors are required.

Total $V_{tr} = 18 + 12 = 30$ transistors. (14)

The maximum delay time for the formation of $S_i^0$ and $S_i^1 = 4 \tau_{le}$, the delay time on parallel functioning multiplexers is $-2 \tau_{le}$. A summary table of the main parameters for the considered adders is given in Table 6.

Table 6. The order.

<table>
<thead>
<tr>
<th>Adders</th>
<th>Number of Transistors (N)</th>
<th>Time of Transfer Formation</th>
<th>Multiplexer Delay</th>
<th>Time of Sum Formation on n-bit Adder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adder on two-stage logic (CM-1)</td>
<td>62</td>
<td>$4 \tau_{le}$</td>
<td>-</td>
<td>$7 \tau_{le}$</td>
</tr>
<tr>
<td>Adder on two schemes “Excluding OR” (CM-3)</td>
<td>50</td>
<td>$2 \tau_{le}$</td>
<td>-</td>
<td>$6 \tau_{le}$</td>
</tr>
<tr>
<td>Adder on a three-input circuit “Excluding OR” (CM-3)</td>
<td>56</td>
<td>$2 \tau_{le}$</td>
<td>-</td>
<td>$3 \tau_{le}$</td>
</tr>
<tr>
<td>Conditional sum adder (CSA) (CM-4)</td>
<td>30</td>
<td>$2 \tau_{le}$</td>
<td>$2 \tau_{le}$</td>
<td>$4 \tau_{le} + \tau_{MUX}$</td>
</tr>
</tbody>
</table>

Table 6 shows that for the construction of a single-digit adder, the minimum number of transistors has an adder with a conditional sum. The same adder has a minimal delay in
the formation of inter-bit transfers. An example of adding numbers to four-digit (N = 4) adders with a conditional sum is shown in Figure 9.

Let \( A = a_3a_2a_1 = 1 1 0 0_{12} \)
\[ B = b_3b_2b_1 = 0 1 0 1_{12} \]
\[ C_0 = 1 0 0 1_{12} \]
\[ C_0 = 1_{12} \]

(15)

Figure 9. Functional diagram of a multi-bit adder based on a single-bit conditional sum adder.

Using Table 3, let us consider the operation of adding numbers A and B:

1. \( a_0 = 0 \quad b_0 = 1 \quad C_0 = 1 \)
   According to the line 7 \( S_0^0 = 1 \quad S_0^1 = 0 \quad C_0^0 = 0 \quad C_0^1 = 1 \)
   At the same time \( S_0 = 0 \) and \( C_1 = 1 \)
2. \( a_1 = 0 \quad b_1 = 1 \quad C_1 = 1 \)
   According to the line 5 \( S_1^0 = 0 \quad S_1^1 = 1 \quad C_1^0 = 0 \quad C_1^1 = 0 \)
   At the same time \( S_1 = 1 \) and \( C_2 = 0 \)
3. \( a_2 = 1 \quad b_2 = 1 \quad C_2 = 0 \)
   According to the line 4 \( S_2^0 = 0 \quad S_2^1 = 1 \quad C_2^0 = 1 \quad C_2^1 = 1 \)
   At the same time \( S_2 = 0 \) and \( C_2 = 1 \)
4. \( a_3 = 1 \quad b_3 = 0 \quad C_2 = 1 \)
   According to the line 7 \( S_3^0 = 1 \quad S_3^1 = 0 \quad C_3^0 = 0 \quad C_3^1 = 1 \)
   At the same time \( S_3 = 0 \) and \( C_4 = S_5 = 1 \)

\( S_{A+B} = 10010_{12} = 18_{10} \)
For N bit adders, \( T_{sm} = T_{sm} = n\tau_{MUX} + 4\tau_{l7} \).
By \( \tau_{MUX} = 2\tau_{l7} \)
\[ T_{sm} = (2n + 4)\tau_{l7}. \] (16)

4. Conclusions

This paper analyzes adders built on the basis of the two-stage logic AND-OR-NOT and on the two- and three-input “exclusive OR” circuits, synthesizing a conditional sum adder. The main parameters (the number of MOSFETs and the speed of adders) are determined. It is shown that from the point of view of the number of transistors and the time of formation of inter-bit transfers, the conditional sum adder is advantageous. The efficiency of the
adder is shown by an example. The final part shows a functional diagram of a four-digit adder with consecutive transfers.

**Author Contributions:** Conceptualization, S.T., A.M. (Assel Mukasheva), G.S., T.I., K.I. and T.I.; methodology, S.T., A.M. (Assel Mukasheva) and K.I.; software, K.I. and A.M. (Assel Mukasheva); validation, A.M. (Adil Mukhamedgali), G.S., T.I. and K.I.; formal analysis, S.T. and T.I.; investigation, S.T., A.M. (Adil Mukhamedgali), G.S. and T.I.; resources, S.T.; data curation, K.I. and A.M. (Adil Mukhamedgali); writing—original draft preparation, T.I. and G.S.; writing—review and editing, K.I., A.M. (Adil Mukhamedgali), G.S. and T.I.; visualization, K.I. and A.M. (Adil Mukhamedgali); supervision, T.I.; project administration, A.M. (Adil Mukhamedgali). All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data are not publicly available.

**Conflicts of Interest:** The authors declare no conflicts of interest.

**References**


**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.