

**О. С. МЕЛЬНИК, В. О. КОЗАРЕВИЧ**

### **МІКРО- ТА НАНОСХЕМИ З ПРОГРАМОВАНИМИ СТРУКТУРАМИ**

Поширення та впровадження великих інтегральних мікро- та наносхем (ВІС) в пристроях цифрової і аналогової електроніки суттєво покращує їх експлуатаційні характеристики і параметри, підвищуючи надійність, продуктивність, швидкодію та зменшуючи собівартість, енергоспоживання, вагові та габаритні показники. Водночас, проектування та виготовлення ВІС є довготривалим і дорогим процесом, до того ж економічно і технологічно виправданим за досить великих об'ємів виробництва. Зараз збільшення універсальності ВІС завжди супроводжується зменшенням їх спеціального застосування. Така суперечливість із запровадженням спеціалізованих і водночас універсальних ВІС з'ясовується на початкових етапах автоматизованого ієрархічного проектування. Для підвищення ефективності систем автоматизованого проектування (САПР) в роботі створені універсальні мікро- та наносхеми з програмованими структурами (МНПС). Однією з переваг програмованих логічних інтегральних схем (ПЛІС) перед ВІС є малий час виготовлення з наперед заданими характеристиками. При цьому береться стандартна мікро- або наносхема і за рахунок подання на певні входи спеціальних сигналів або відповідним з'єднанням виводів напрямлено змінюються її параметри. Ця перевага визначає основне призначення таких ПЛІС – заміну груп логічних ІС середнього і великого ступенів інтеграції. В якості простих ПЛІС можуть використовуватися мультиплексори. В статті наведені ефективні методи програмування мультиплексорних МНПС для реалізації функцій булевої та мажоритарної логіки. Отримані результати можна використовувати для репрограмування мультиплексорних функціональних блоків програмованих інтегральних схем. На сучасних САПР виконане співставне моделювання логічних МНПС, яке довело адекватність їх функціонування, переваги частотних і недоліки температурних характеристик наномультіплексорних схем.

**Ключові слова:** мікро- та наносхеми, цифрова та аналогова електроніка, програмовані логічні структури, автоматизоване проектування, мультиплексори.

**А. С. МЕЛЬНИК, В. А. КОЗАРЕВИЧ**

### **МИКРО- И НАНОСХЕМЫ С ПРОГРАММИРУЕМЫМИ СТРУКТУРАМИ**

Распространение и внедрение больших интегральных микро- и наносхем (БИС) в устройствах цифровой и аналоговой электроники существенно улучшает их эксплуатационные характеристики и параметры, повышая надежность, производительность, быстродействие и уменьшая себестоимость, энергопотребление, весовые и габаритные показатели. В то же время, проектирование и изготовление БИС является длительным и дорогостоящим процессом, к тому же экономически и технологически оправданными при достаточном количестве больших объемов производства. Сейчас увеличение универсальности БИС всегда сопровождается уменьшением их специального применения. Такая противоречивость с внедрением специализированных и одновременно универсальных БИС выясняется на начальных этапах автоматизированного иєрархического проектирования. Для повышения эффективности систем автоматизированного проектирования (САПР) в работе созданы универсальные микро- и наносхеми с программируемыми структурами (МНПС). Одним из преимуществ программируемых логических интегральных схем (ПЛІС) перед БИС является малое время изготовления с заданными характеристиками. При этом берется стандартная микро- или наносхема и за счёт подавания на определенные входы специальных сигналов или соответствующим соединением выводов направленно изменяются ее параметры. Это преимущество определяет основное предназначение таких ПЛІС – замену групп логических ІС средней и большой степеней интеграции. В качестве простых ПЛІС могут использоваться мультиплексоры. В статье приведены эффективные методы программирования мультиплексорных МНПС для реализации функций булевой и мажоритарной логіки. Полученные результаты можно использовать для репрограммирования мультиплексорных функциональных блоков программируемых интегральных схем. На современных САПР выполнено сопоставимое моделирование логических МНПС, доказавшее адекватность их функционирования, преимущества частотных и недостатки температурных характеристик наномультіплексорных схем.

**Ключевые слова:** микро- и наносхеми, цифровая и аналогова електроніка, программируемые логические структуры, автоматизированное проектирование, мультиплексоры.

**O. S. MELNYK, V. O. KOZAREVYCH**

### **MICRO- AND NANOCIRCUITS WITH PROGRAMMABLE STRUCTURES**

In this work, structural programming does not mean the creation of algorithms for processing multi-argument functions by changing the operating programs, as implemented by the microprocessor, but rather technological changes in the large-scale integrated circuits (LSIC) configurations in such a way as to synthesize logical functions at the structural-logical level. During the automated design and manufacture of programmable LSIC (FPGA or CPLD), the same technological cycle is used as for specialized circuits. Obviously, such LSICs are universal for technologists. However, programming of individual LSICs that implement given functions is performed by the user. Thus, the main advantage of universal (FPGA, CPLD) programmable LSIC over specialized ones is low cost, which is fundamentally important for small-scale production.

Currently, the increase in the universality of LSIC is always accompanied by a decrease in their special application. Such inconsistency is revealed in the initial stages of automated hierarchical design. Universal micro- and nanocircuits with programmable structures are used to increase the efficiency of CAD. One of the advantages of programmable logic integrated circuits (PLCs) over LSIC is a short manufacturing time with predetermined characteristics. At the same time, a standard micro- or nanocircuit is taken and its parameters are directly changed by applying special signals to certain inputs or connecting the outputs accordingly. This advantage determines the main purpose of such FGAs - the replacement of groups of logic ICs of medium and large degrees of integration. Multiplexers can be used as simple FGAs. The article introduces effective methods of multivariate programming of multiplexer MNPS for reproduction of Boolean and majority logic functions. The obtained results can be used for reprogramming multiplexer functional blocks of programmable integrated circuits. Comparative modeling of logical MNPS was performed on modern CAD systems, which proved the adequacy of their functioning, advantages of frequency and disadvantages of temperature characteristics of nanomultiplexer circuits.

**Key words:** micro- and nanocircuits, digital and analog electronics, programmable logical structures, computer-aided design, multiplexer.

**Вступ.** Під конфігуруванням (структурним програмуванням) мікро- та наномультіплексорів розуміють не здатність створювати алгоритми опрацювання вхідних багатозмінних функцій, змінюючи робочі програми, як, зазвичай, це реалізує мікроконтролер, а технологічні зміни внутрішніх конфігурацій електронних кіл таким чином, щоб вони забезпечували синтез необхідних функцій на структурно-логічному рівні.

Під час *автоматизованого проектування* та виготовлення *програмованих логічних інтегральних схем (ПЛІС)* на базі *замовних великих інтегральних схем (ВІС)* використовується той же технологічний цикл, що і для спеціалізованих схем. Вочевидь, для технологів такі ВІС є універсальними. Проте програмування окремих ВІС, які реалізують задані функції, виконує користувач. Таким чином, головною перевагою універсальних *мікро-* та *наносхем з програмованими структурами (МНПС)* перед спеціалізованими є низька собівартість, що принципово важливо при дрібносерійному виробництві [1, 2].

**Аналіз останніх досліджень і публікацій.** Науковці Харківського університету радіоелектроніки [3] та американського університету Клівленда [4] першими запропонували багатоструктурні мікро- та наносистеми на базі універсальних функціонально повних модулів, як один з найбільш перспективних напрямів розвитку електроніки.

Останнім часом дослідження і результати автоматизованого ієрархічного проектування програмованих мікро- та наносхем достатньо повно описані у вітчизняних та закордонних публікаціях [1 – 14]. В роботі [2] досліджені сучасні напрямки створення і розвитку нанорозмірних репрограмованих пристроїв. Однак ця робота не відповідає на питання автоматизованого моделювання і верифікації проектів. Існує багато технічних задач, в яких в рамках неавтоматизованого програмування мікро- та наносхем [5 – 8] вдається реалізувати 16 дво- та 256 триаргументних функцій на базі мультиплексорів. Проте в якості універсального логічного мікроелемента такі реалізації заздалегідь носять надлишковий характер. Крім того, припускається, що такі схеми є *відмовостійкими*. Вирішенню цих проблем були присвячені роботи авторів цієї статті [1, 9], але практично не була ще досягнута мета автоматизованого проектування МНПС.

В роботах [10 – 12] дослідники запропонували ефективні методи конфігурування мікро- та нанопристроїв для відтворення широкого спектру функцій алгебри логіки комбінаційних та послідовностних пристроїв. В публікації [10] представлено алгоритм автоматизованого програмування мікросхем *великого ступеня інтеграції*, який частково відтворений в даній статті для наноелектронних приладів з програмованими структурами.

В монографіях *Kilts S.* [13] та *Groul I.* [14] обґрунтовуються моделі схем та алгоритми їх програмування, але не доведена можливість багатоваріантних підходів спрощення та адекватних їх реалізацій.

В цій статті запропоновані багатоваріантні процедури контролю результатів програмування мікро- та наносхем. Питання *синхронізації* та *впливу кріотемператури* на працездатність нанопристроїв ще недостатньо досліджені [1, 2, 5], тому вони стимулюють подальший пошук додаткових підходів для підвищення якості МНПС.

**Метою роботи** є запровадження автоматизованих методів моделювання та проектування мультиплексорних мікро- та наносхем з програмованими структурами для реалізації *логічних функцій булевої та мажоритарної логіки*. Експлуатаційні характеристики МНПС заздалегідь гірші, ніж у стандартних мікросхем.

**Програмування мультиплексорних мікро- та наноструктур.** Для використання мультиплексорів в якості універсального логічного наноелемента на його адресні входи комутують сигнали деяких аргументів відтвореної функції, а інформаційні входи виконують роль програмованих наноструктур. Для цифрового пристрою з двома вхідними змінними  $x_1$ ,  $x_0$  можливі чотири варіанти комбінацій аргументів: 00, 01, 10, 11 і 10 та 16-ть різних вихідних функцій (табл. 1).

Таблиця 1 – Логічні функції двох аргументів

$x_1$	$x_2$	$f_0$	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	$f_8$	$f_9$	$f_{10}$	$f_{11}$	$f_{12}$	$f_{13}$	$f_{14}$	$f_{15}$
0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Кожна функція  $f_i$  ( $i = 0, \dots, 15$ ) є результатом виконання однієї з операцій над аргументами  $x_1$ ,  $x_0$  і являє собою чотирирозрядні двійкові числа від 0000 до 1111. Зверху в табл. 1 показано мультиплексор ( $4 \rightarrow 1$ ), на входи якого подають сигнали програмування  $D_3, \dots, D_0$  для відтворення відповідної функції  $f_i$  [9].

Шість тривіальних функцій, серед яких  $f_3 = \overline{x_1}$ ,  $f_5 = \overline{x_0}$ ,  $f_{10} = x_0$ ,  $f_{12} = x_1$  залежать тільки від одного аргументу, а нульова  $f_0 = 0$  і одинична  $f_{15} = 1$  взагалі є константами. Решта десять функцій  $f_i(x_1, x_0)$  є двоаргу-

ментними. Деякі з них мають декілька назв. Наприклад,  $f_1 = \overline{x_1 \vee x_0}$  зветься АБО-НІ, додавання (диз'юнкція) із запереченням, *стрілка Пірса*;  $f_2 = \overline{x_1}x_0$  чи  $f_4 = x_1\overline{x_0}$  – заборона  $x_1$  чи  $x_0$ ,  $f_3 = \overline{x_1}$  чи  $f_5 = \overline{x_0}$  – заперечення  $x_1$  чи  $x_0$ ;  $f_6 = x_1x_0 \vee \overline{x_1}\overline{x_0} = x_1 \oplus x_0$  – нерівнозначність, Виключне АБО, додавання за модулем 2;  $f_7 = \overline{x_1x_0}$  – І-НІ, заперечення множення (*кон'юнкції*), *итрих Шеффера*;  $f_8 = x_1x_0$  – І, множення (добуток, кон'юнкція);  $f_9 = x_1x_0 \vee \overline{x_1}\overline{x_0} = x_1 \equiv x_0 = \overline{f_6}$  – Виключне АБО-НІ, рівнозначність (еквівалентність);  $f_{10} = x_0$  чи  $f_{12} = x_1$  – тотожність  $x_0$  чи  $x_1$ ;  $f_{11} = \overline{x_1}x_0$  чи  $f_{13} = x_1\overline{x_0}$  – пряма чи зворотна (заперечна) *імплікація* та  $f_{14} = x_1 \vee x_0$  – АБО, додавання (диз'юнкція).

За потреби реалізації багатоаргументних логічних функцій, зазвичай, використовують структуру мультиплексорного дерева [6]. Однак, якщо число вхідних змінних не суттєво перевищує число адресних входів мікро- чи наномультимплексора, то такі задачі вирішують вибором сигналів репрограмування МНПС не з множини  $\{0, 1\}$ , як це показано в табл. 1, а з множини  $\{x_i, 0, 1\}$ , де  $x_i$  – один з аргументів відтворюваної функції. В цьому випадку вдається на МНПС без додаткових апаратних витрат реалізувати логічну функцію, число аргументів якої більше числа адресних входів.

На рис. 1 наведені умовне позначення (а) мультиплексора ( $2 \rightarrow 1$ ), його еквівалентні МНПС на базі логічних мікроелементів (б) та наноелектронних мажоритарних елементів [3] (в) і таблиця істинності (г).

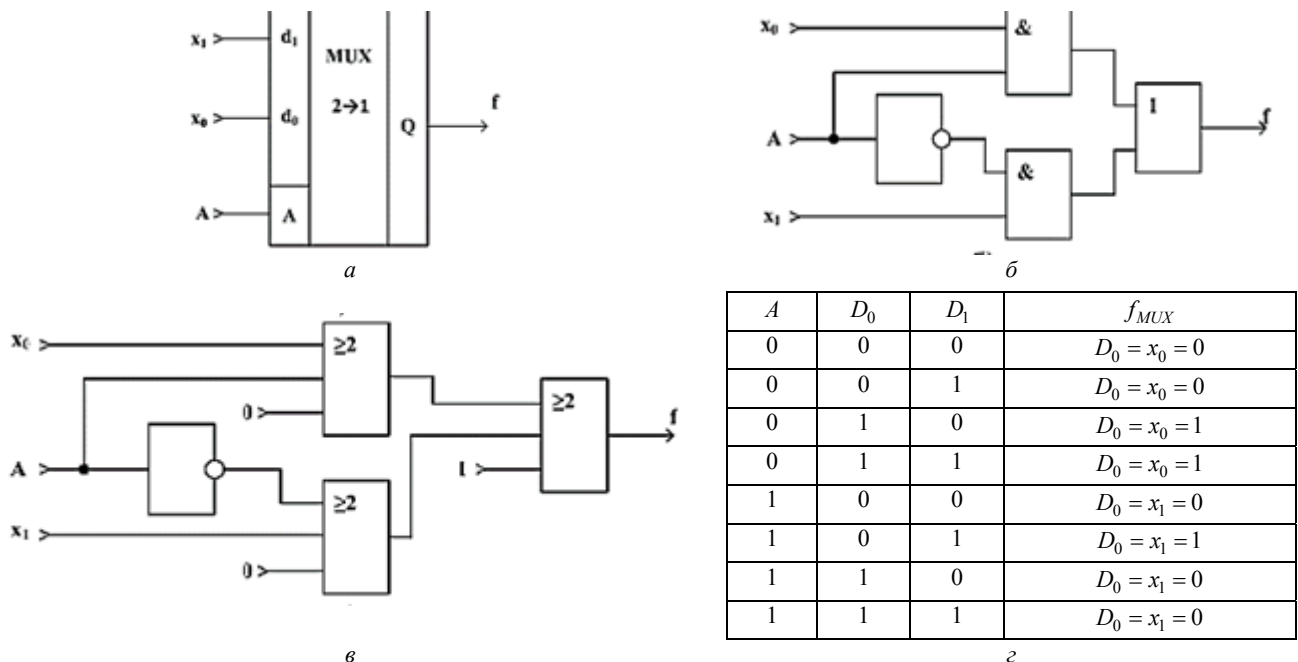


Рис. 1 – а – Двовходовий мультиплексор; б – мікросхема мультиплексора; в – наносхеми мультиплексора; г – таблиця істинності.

З таблиці на рис. 1, г після дужкових перетворень отримують функцію алгебри логіки мультиплексора ( $2 \rightarrow 1$ ):

$$f_{MUX} = \overline{AD_0}D_1 \vee \overline{AD_1}D_0 \vee AD_0D_1 = D_1A \vee D_0\overline{A}. \tag{1}$$

Досить зрозуміло виконується програмування шести тривіальних функцій  $f_0, f_3, f_5, f_{10}, f_{12}$  та  $f_{15}$  з табл. 1 на одноадресних мультиплексорах ( $2 \rightarrow 1$ ). Наприклад, для реалізації функції заперечення  $f_3 = \overline{x_1}$  слід порівняти таблиці істинності самої функції (табл. 1) та мультиплексора (рис. 1, г), з яких створюють тотожність:

$$f_3 = \overline{x_1} = f_{MUX} = D_1A \vee D_0\overline{A}. \tag{2}$$

Згідно цьому рівнянню можливі чотири варіанти програмування адресного входу МНПС  $A = \overline{x_1}, x_1, 0, 1$  та відповідно, інформаційних входів  $D_1$  і  $D_0$ : 1)  $A = \overline{x_1}$ , тоді  $\overline{x_1} = D_1\overline{x_1} \vee D_0x$ , звідки для виконання цієї тотожності (2)  $D_1 = \overline{x_1}$  або 1, а  $D_0 = \overline{x_1}$  або 0; 2)  $A = x_1, D_1 = \overline{x_1}$  або 0,  $D_0 = \overline{x_1}$  або 1; 3)  $A = 0, D_1 = \overline{x_1}$  або 1,  $D_0 = \overline{x_1}$ ; 4)  $A = 1, D_1 = \overline{x_1}, D_0 = \overline{x_1}$  або 0.

Отже, практичне програмування функції  $f_3$  на базі двовходових МНПС (рис. 1) свідчить, що при будь-якій комбінації адресних аргументів на виході реалізуються однотипні функції програмування. У такий самий спосіб

програмуються наступні одноаргументні функції  $f_5$ ,  $f_{10}$  та  $f_{12}$ .

Для відтворення ще десяти двоаргументних функцій  $f_1$ ,  $f_2$ ,  $f_4$ ,  $f_6$ ,  $f_8$ ,  $f_9$ ,  $f_{11}$ ,  $f_{13}$  та  $f_{14}$  з табл. 1 необхідно виконати 4 перетворення, щоб отримати диз'юнктивні нормальні форми (ДНФ). Так, використовуючи закон двойності (інверсії) де Моргана для функції АБО-НІ  $f_1 = \overline{x_1 \vee x_0}$ , отримують  $f_1 = \overline{x_1 x_0}$  та реалізують алгоритм програмування (2):

$$f_1 = \overline{x_1 x_0} = D_1 A \vee D_0 \overline{A}. \quad (3)$$

Тоді при чотириразовому програмуванні єдиного адресного входу константами аргументів  $A = \overline{x_1}, x_1, \overline{x_0}, x_0$  слід подати на два інформаційні входи наступні комбінації сигналів:

$$A = \overline{x_1}, D_1 = \overline{x_0}, D_0 = \overline{x_1} \text{ або } 0; \quad A = x_1, D_1 = \overline{x_1} \text{ або } 0, \quad D_0 = x_0;$$

$$A = \overline{x_0}, D_1 = x_1, D_0 = x_0 \text{ або } 0; \quad A = x_0, D_1 = x_0 \text{ або } 0, \quad D_0 = x_1.$$

Далі для отримання ДНФ функції додавання  $f_{11}$ ,  $f_{13}$  та  $f_{14}$  (табл. 1) слід помножити неповні доданки на одиночні суми відсутніх аргументів  $(x_i \vee \overline{x_i})$  чи  $(x_i \vee 1)$ . Вочевидь, для реалізації функції додавання АБО  $f_{14} = x_1 \vee x_0$  можливі наступні чотири однотипні дужкові перетворення:

$$f_{14} = x_1 (x_0 \vee \overline{x_0}) \vee x_0 (x_1 \vee \overline{x_1}) = \begin{cases} 1) x_1 \overline{x_0} \vee x_0 (x_1 \vee x_1 \vee \overline{x_1}) = x_1 \overline{x_0} \vee x_0, \\ 2) \overline{x_1} x_0 \vee (x_0 \vee x_0 \vee \overline{x_0}) = x_1 x_0 \vee x_1; \\ 3) x_1 (x_0 \vee \overline{x_0}) \vee x_0 (x_1 \vee 1) = 3) x_1 \overline{x_0} \vee x_0 (x_1 \vee x_1 \vee 1) = x_1 \overline{x_0} \vee x_0; \\ 4) x_1 (x_0 \vee 1) \vee x_0 (x_1 \vee \overline{x_1}) = 4) \overline{x_1} x_0 \vee x_1 (x_0 \vee x_0 \vee 1) = \overline{x_1} x_0 \vee x_1. \end{cases} \quad (4)$$

Таким чином, як і два попередні перетворення (2) та (3), реалізується алгоритм чотириваріантного репрограмування МНПС для синтезу функції  $f_{14}$ :

$$A = \overline{x_1}, D_1 = x_0, D_0 = \overline{x_1} \text{ або } 1; \quad A = x_1, D_1 = x_1 \text{ або } 1, \quad D_0 = x_0;$$

$$A = \overline{x_0}, D_1 = x_1, D_0 = x_0 \text{ або } 1; \quad A = x_0, D_1 = x_0 \text{ або } 1, \quad D_0 = x_1.$$

Для мультиплексорної реалізації функції І-НІ отримують її ДНФ, виконуючи, по-перше, перетворення де Моргана, а по-друге – дужкові:

$$f_7 = \overline{x_1 x_0} = \overline{x_1} \vee \overline{x_0} = \overline{x_1} (x_0 \vee \overline{x_0}) \vee \overline{x_0} (x_1 \vee \overline{x_1}).$$

Дві функції заборони  $f_2 = \overline{x_1} x_0$  та  $f_4 = x_1 \overline{x_0}$ , одна функція перемноження І  $f_8 = x_1 x_0$ , як і модулярні функції  $f_6 = x_1 \oplus x_0$  та  $f_9 = \overline{x_1} \oplus \overline{x_0}$ , записані у ДНФ. Тому їх програмування є подібним до попередніх випадків, наприклад, як (3) для  $f_1$ .

**Результати і обговорення.** З розглянутих вище прикладів отримують підсумкову таблицю програмування МНПС мультиплексорів ( $2 \rightarrow 1$ ) для реалізації 16 двоаргументних функцій (табл. 2). Верхня її частина практично повністю повторює табл. 1, але з адресацією, а в нижній, власне, зібрані 4 варіанти вибору адресних сигналів  $A$  з множини  $\{\overline{x_1}, x_1, \overline{x_0}, x_0, 0, 1\}$  для кожної з відтворюваних функцій.

Таблиця 2 – Логічні функції двох аргументів для МНПС мультиплексорів ( $2 \rightarrow 1$ )

$x_1 \backslash x_0$	$x_0$	A	$f_0 = 0$	$f_1 = \overline{x_1} \vee \overline{x_0}$	$f_2 = \overline{x_1} x_0$	$f_3 = \overline{x_1}$	$f_4 = x_1 \overline{x_0}$	$f_5 = \overline{x_0}$	$f_6 = x_1 \oplus x_0$	$f_7 = \overline{x_1} \overline{x_0}$	$f_8 = x_1 x_0$	$f_9 = \overline{x_1} \oplus \overline{x_0}$	$f_{10} = x_0$	$f_{11} = \overline{x_1} \vee x_0$	$f_{12} = x_1$	$f_{13} = x_1 \vee \overline{x_0}$	$f_{14} = x_1 \vee x_0$	$f_{15} = 1$
0/0	0/0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0/1	0/1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
1/0	1/0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1/1	1/1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1

Продовження таблиці 2

1	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
Варіанти програмування	1	$D_0$	0	$\bar{x}_1(0)$	$\bar{x}_1(0)$	$\bar{x}_1(0)$	$x_0$	$\bar{x}_1(0)$	$\bar{x}_1(0)$	$\bar{x}_0$	$x_0$	$x_0$	$x_0(1)$	$x_0$	$x_1(1)$	$x_1(1)$	$x_1(1)$	1
		$D_1$	0(1)	$\bar{x}_0$	$x_0$	$\bar{x}_1(1)$	$x_1(0)$	$\bar{x}_0(1)$	$x_0$	$\bar{x}_1(1)$	$x_1(0)$	$\bar{x}_0$	$x_0(0)$	$\bar{x}_1(1)$	$x_1(0)$	$\bar{x}_0$	$x_0$	0(1)
		$A$	0	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_0$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_0$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_1$
	2	$D_0$	0(1)	$\bar{x}_0$	$x_0$	$\bar{x}_1(1)$	$x_1(0)$	$\bar{x}_0(1)$	$x_0$	$\bar{x}_1(1)$	$x_1(0)$	$\bar{x}_0$	$x_0(0)$	$\bar{x}_1(1)$	$x_1(0)$	$\bar{x}_0$	$x_0$	0(1)
		$D_1$	0	$\bar{x}_1(0)$	$\bar{x}_1(0)$	$\bar{x}_1(0)$	$\bar{x}_0$	$\bar{x}_0(0)$	$\bar{x}_0$	$\bar{x}_0$	$x_0$	$x_0$	$x_0(1)$	$x_0$	$x_1(1)$	$x_1(1)$	$x_1(1)$	1
		$A$	1	$x_1$	$x_1$	$x_1$	$x_1$	$x_0$	$x_1$	$x_1$	$x_1$	$x_1$	$x_0$	$x_1$	$x_1$	$x_1$	$x_1$	1
	3	$D_0$	$\bar{x}(0)$	$\bar{x}_1(0)$	$\bar{x}_1$	$\bar{x}_1$	$\bar{x}_0(0)$	$\bar{x}_0$	$\bar{x}_1$	$\bar{x}_1$	$x_1$	$x_1$	$x_0$	$x_0(1)$	$x_1$	$x_1$	$x_0(1)$	$x(1)$
		$D_1$	$x(0)$	$\bar{x}_1$	$x_0(0)$	$\bar{x}_1(0)$	$x_1$	$\bar{x}_0(0)$	$x_1$	$\bar{x}_0(1)$	$x_0(0)$	$\bar{x}_1$	$x_0(0)$	$\bar{x}_1$	$x_1(0)$	$\bar{x}_0(1)$	$x_1$	$\bar{x}(1)$
		$A$	$\bar{x}$	$\bar{x}_0$	$\bar{x}_0$	0	$\bar{x}_0$	0	$\bar{x}_0$	$\bar{x}_0$	$\bar{x}_0$	$\bar{x}_0$	0	$\bar{x}_0$	0	$\bar{x}_0$	$\bar{x}_0$	$\bar{x}$
	4	$D_0$	$x(0)$	$\bar{x}_1$	$x_0(0)$	$\bar{x}_1(0)$	$x_1$	$\bar{x}_0(0)$	$x_1$	$\bar{x}_0(1)$	$x_0(0)$	$\bar{x}_1$	$x_0(0)$	$\bar{x}_1$	$x_1(0)$	$x_0(1)$	$x_1$	$\bar{x}(1)$
		$D_1$	$\bar{x}(0)$	$\bar{x}_1(0)$	$\bar{x}_1$	$\bar{x}_1$	$x_0(0)$	$\bar{x}_0$	$\bar{x}_1$	$x_1$	$x_1$	$x_1$	$x_0$	$x_0(1)$	$x_1$	$x_1$	$x_0(1)$	$x(1)$
		$A$	$x$	$x_0$	$x_0$	1	$x_0$	1	$x_0$	$x_0$	$x_0$	$x_0$	1	$x_0$	1	$x_0$	$x_0$	$x$

Згідно рис. 1, б двохдодова МНПС побудована на площі проектування САПР Мікро-сар 11 [15] (рис. 2) з логічних елементів (а) та з окремого елемента в одному корпусі в макрос (б).

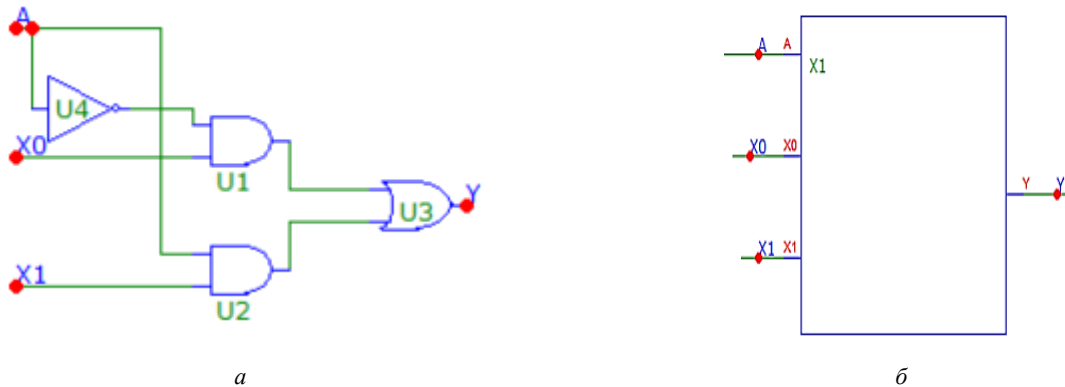


Рис. 2 – Двохдодова МНПС: а – на мікроелементах; б – в корпусі макрос.

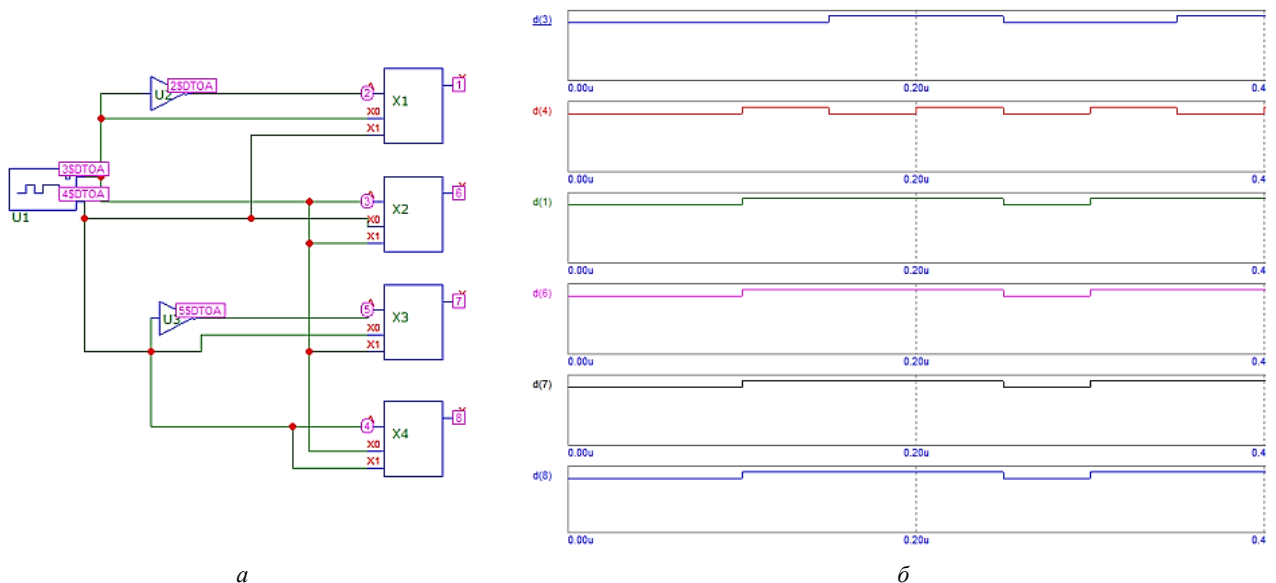


Рис. 3 – а – Автоматизоване програмування МНПС; б – часові реалізації запрограмованої функції АБО.

Рис. 3 ілюструє схему комп'ютерної реалізації функції АБО на чотирьох МНПС (а), які запрограмовані згідно перетворенням (4) і стовпцю  $f_{14}$  з табл. 2 та її результати моделювання (б).

Генератор  $U_1$  відтворює *меандри сигналів* двох аргументів  $x_1, x_0$ , які комутують входи кожної мультиплексорної МНПС ( $x_1, \dots, x_4$ ) для програмування функції  $f_{14}$ . На двох верхніх часових діаграмах  $d_3, d_4$  показані сигнали  $x_1, x_0$ , а чотири нижні повністю співпадаючі осцилограми  $d_1, d_5, d_7$  і  $d_8$  підтверджують програмування і реалізації функції АБО на МНПС.

Розвиток наноелектроніки, в свою чергу, йде в бік зменшення розмірів, енерговитрат, підвищення швидкості і ступеня інтеграції. Одноелектронні наносхеми з'явилися за принципових обмежень, що виникли на шляху розвитку мікросхем [5, 6]. Гнучке програмування функцій і наноструктур почало суттєво переважати програмування мікросхем (ПЛІС) за рахунок структурних переваг і меншого надлишка. Тому вкрай необхідним є впровадження нових методів моделювання ПНПС.

Квантові *коміркові автомати (КА)* – це обчислювальна парадигма, згідно з якою інформація представляється певною конфігурацією електронів в комірці КА, що формується з однієї чи двох окремих молекул [16, 17].

Прилади на КА складаються з нанорозмірних комірок-діелектриків, які мають чотири квантові напівпровідникові точки, розташовані у кутках, і два рухомі електрони. Порядком їх розміщення залежить лише від скінченного набору можливих значень в межах певної комірки. Окрема комірка забезпечує тунельний перехід електронів крізь потенціальні бар'єри. Переходи контролюються локальним електричним полем, величина якого збільшується для перешкодження руху електронів, або зменшується для його спонукання. Стани поляризації виникають, коли величина потенціального бар'єру зростає і зберігає свій рівень, щоб мінімізувати енергетичний рівень комірки. Набір станів  $Q$  скінченний і типовий:  $Q = \{0, 1\}$ . Ймовірність перебування комірки в одному із станів поляризації може бути співвіднесена з густиною заряду кожної окремої квантової точки і визначається формулою:

$$P = \frac{(\rho_1 + \rho_3) - (\rho_2 + \rho_4)}{(\rho_1 + \rho_3) + (\rho_2 + \rho_4)} = \pm 1, \quad (5)$$

де  $\rho_i$  – густина електричного заряду кожної квантової точки комірки.

На рис. 4 зображена базова комірка КА, два способи її розміщення в просторі і поляризації електронів.

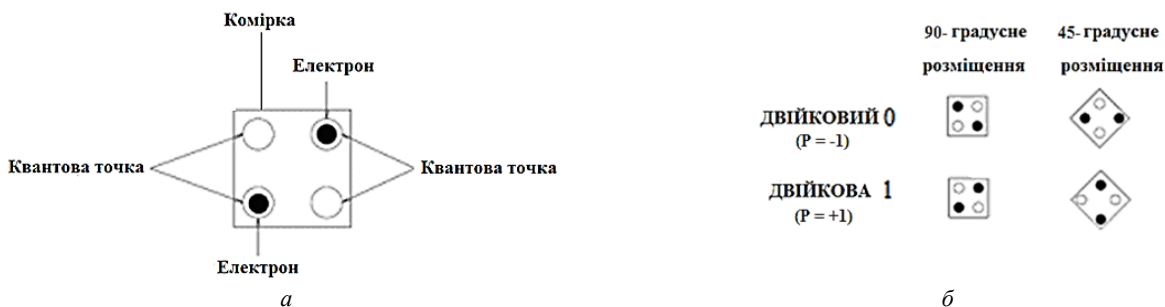


Рис. 4 – *a* – Окрема комірка квантового автомата; *б* – два способи розміщення КА в просторі і поляризації ( $P = \pm 1$ ).

Таким чином, бінарна інформація передається без руху зарядів. Тобто нема протікання струму між комірками. Це основна причина, чому структури КА споживають надмізерну кількість енергії  $10^{-23}$  Дж.

За допомогою КА можуть бути сконструйовані різні елементи для виконання арифметичних і логічних операцій. Базовими логічними елементами в теорії КА є *тривходовий мажоритарний елемент (МЕ)* та *інвертор* (рис. 5).

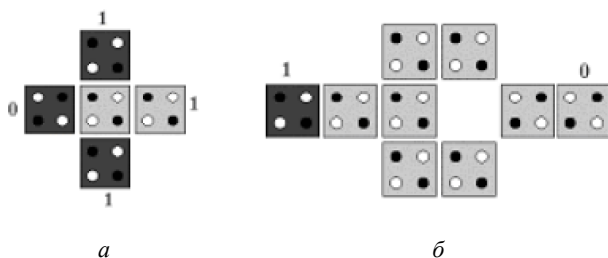


Рис. 5 – *a* – Мажоритарний елемент; *б* – інвертор на базі квантових автоматів.

Вихідна комірка МЕ буде мати поляризацію більшості вхідних комірок. Булевий вираз для мажоритарної функції [3, 16]:

$$\text{maj}(x_2, x_1, x_0) = x_2 x_1 \vee x_2 x_0 \vee x_1 x_0, \quad (6)$$

де  $x_2, x_1$  і  $x_0$  – входи. Фіксація поляризації одного з входів МЕ, як логічний 0 чи логічна 1, дає змогу отримати елементи І чи АБО, відповідно (6) отримують  $\text{maj}(x_2, x_1, 0) = x_2 \cdot x_1$  чи  $\text{maj}(x_2, x_1, 1) = x_2 \vee x_1$ . Такі комірки можуть бути створені в технологічному процесі, що усуває необхідність підтримувати постій-

ний струм через схему.

Створена раніше МНПС двовходового наномультимплексора (рис. 1, *в*) побудована на проектному планшеті САПР одноелектронних наносхем QCADesigner [18] на рис. 6, *a*. Вона складається з інвертора адресного сигналу  $A$ , двох вхідних МЕ, структури яких запрограмовані нульовими поляризаціями – 1, 00 для виконання операції кон'юнкції  $x_1 A$  і  $x_0 \bar{A}$ , та вихідного МЕ, структурно запрограмованого одиночною поляризацією 1, 00 для від-



творення диз'юнктивної вихідної логічної функції  $F = f_{MUX}$  МНПС (1). На рис. 6, б наведені результати моделювання її часових діаграм, які повністю співпадають з таблицею істинності (рис. 1, з). Однак на осцилограмі вихідної функції  $F$  існує затримка на один такт аргумента  $x_1$  через необхідність використання асинхронних моделей тактових сигналів на ME  $DD_2$  (для  $x_1$ ) та  $DD_4$  (для  $F$ ), які мають різні кольорові забарвлення.

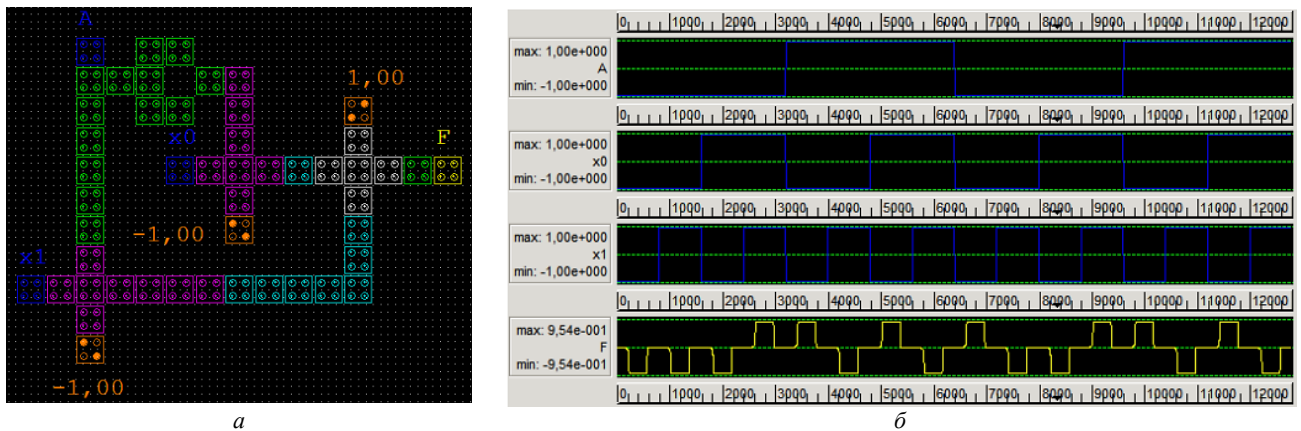


Рис. 6 – а – Автоматизоване програмування наносхеми МНПС; б – відповідні осцилограми.

Автоматизоване проектування одноелектронної МНПС виконується з структурним налаштуванням (рис. 7, а порівняно з рис. 6, а) і обранням програмування за 4-м варіантом для функції  $f_{14}$  АБО (табл. 2):

$$A = x_0, x_1(D_1) = x_0 \text{ та } x_0(D_0 = x_1).$$

Верифікація відповідності часового моделювання (рис. 7, б) та таблиць істинності рис. 1, з і 2 підтверджує адекватність налаштованої МНПС.

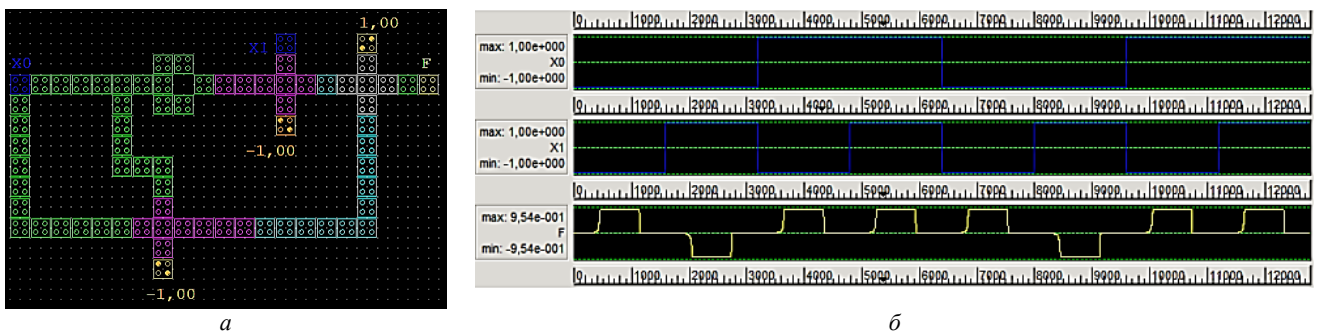


Рис. 7 – Автоматизоване проектування МНПС на КА: а – одноаргументна наносхема; б – результати часового моделювання.

МНПС збирається з 70 комірок КА розміром  $(18 \times 18)$  нм кожна, з 4-ма квантовими точками діаметром 5 нм і відстанню між центрами 20 нм, загальний розмір наносхеми складає  $(128 \times 212)$  нм.

Окрім того, порівняння результатів моделювання осцилограм мікро- (рис. 3, б) та нанорозмірних (рис. 7, б) мультиплексорів дозволяє зробити висновок, що вони повністю співпадають і достовірно відтворюють запрограмовану логічну функцію АБО.

**Висновки.** Оскільки розвиток мікроелектроніки на базі комплементарних транзисторів із структурою метал-окисел-напівпровідник (КМОПТ) лімітований квантово-технологічними і малорозмірними ефектами [5], то в статті надається перевага саме застосуванню наноелектронних програмованих структур для реалізації логічних функцій. Розроблені алгоритми (табл. 2) автоматизованої модифікації МНПС (рис. 3 і 7) підтвердили адекватність розроблених моделей. Проте, недоліком одноелектронних схем є надкритичний температурний діапазон (від 0 до 4 К) працездатності [16, 17], що, ймовірно, пов'язано з впливом явищ термогенерації фонових електронів. Наносхеми мають переваги у швидкодії, продуктивності, відмовостійкості та малорозмірності, тому надалі є найбільш перспективною компонентною базою електроніки.

#### Список літератури

1. Мельник О. С., Миколушко А. М. Репрограмовані мультиплексорні наносхеми // Вісник НТУ «ХПІ». Серія : Математичне моделювання в техніці та технологіях. – Харків : НТУ «ХПІ», 2019. – № 8 (1333). – С. 224 – 231.
2. Коротун А. В., Карандась Я. В., Погосов В. В. Нарис сучасних напрямків у нанотехнологіях. – Ужгород : ФОП Сабов А.М., 2019. – 392 с.
3. Пакулов Н. И. Мажоритарный принцип построения надежных узлов и устройств ЦВМ. – М. : Сов. Радио, 1974. – 194 с.
4. Chu P. P. FPGA prototyping by VHDL examples Xilinx SpartanTM-3 Version. – WILEY & SONS, 2008. – 528 p.
5. Etienne Sicard, Sonia Delmas Bendhia. Advanced CMOS cell design. – Компания McGraw-Hill, 2007. – 385 p.
6. Матвієнко М. П. Проектування цифрових пристроїв. – Київ : Ліра-К., 2018. – 358 с.

7. Roy S. *Advanced Digital System Design: FPGA and ASIC Implementation*. – Ane Books, 2021. – 418 p.
8. Рябенкий В. М., Жушков В. Я., Гулий В. Д. Цифрова схемотехніка: навчальний посібник. – Л.: Новий Світ, 2009. – 736 с.
9. Melnyk O. S., Yurchuk I. I. Nanodevices with Programmable Logic // *International scientific journal «Electronics and control systems»*. – 2019. – v. 4. – № 62. – P. 47 – 52. DOI:10.18372/1990-5548.62.14383.
10. Harris D., Harris S. *Digital Design and Computer Architecture*. – Morgan K., 2017. – 307 p.
11. Бруно Ф. Программирование FPGA для начинающих // Пер. с англ. под редакцией Романова. – М.: ДМК Пресс, 2022. – 320 с.
12. Сочков А. Г. *Електроніка і мікросхемотехніка*. – Київ: Каравела, 2009. – 416 с.
13. Kiltz S. *Advanced FPGA Design: Architecture, Implementation, and Optimization*. – Wiley, 2016. – 537 p.
14. Grout Ian. *Digital systems design with FPGAs and CPLDs*. // Library of Congress Cataloging-in-Publication Data, 2008. – 763 p.
15. Micro-Cap 11 Full CD Version 11.2.0.3. – Режим доступу: <https://archive.org/details/micro-cap-11-full-cd-version-11.2.0.3>. – Дата звертання: 02 вересня 2023.
16. Lent C. S., Tougaw P. D. A device architecture for computing with quantum dots // *Proc. of the IEEE*. – 1997. – vol. 85. – № 5. – pp. 541 – 557. DOI: 10.1109/5.573740.
17. Мельник О. С., Козаревич В. О., Пилипенко Р. І. Комп'ютерне проектування арифметичних нанопристроїв // Вісник НТУ «ХПІ». Серія: Математичне моделювання в техніці та технологіях. – Харків: НТУ «ХПІ», 2014. – № 39 (1082). – С. 126 – 134.
18. Walus K., Dysart T.J., Jullien G.A., Budiman R.A. QCADesiner: A Rapid Design and Simulation Tool for Quantum-Dot Cellular Automata // *IEEE Transactions on Nanotechnology* – 2004. – v. 3. – № 1. – pp. 26–31. DOI:10.1109/TNANO.2003.820815.

## References (transliterated)

1. Mel'nyk O. S., Mykolushko A. M. Reprogramovani mul'tipleksorni nanoskhemy [Reprogrammed multiplexer nanocircuits]. *Visnyk NTU «KhPI». Seriya: Matematychnе modelyuvannya v tekhnitsi ta tekhnologiyakh* [Bulletin of the National Technical University "KhPI". Series: Mathematical modeling in engineering and technology]. Kharkiv, NTU «KhPI» Publ., 2019, no. 8 (1333), pp. 224–231.
2. Korotun A. B., Karandas' Ya V., Pogosov V. V. *Narys suchasnykh napryamkiv u nanotekhnologiyakh* [Outline of modern trends in nanotechnology]. Uzhgorod, FOP Sabov A.M., 2019. 392 p.
3. Pakulov N. I. *Mazhoritarnyy printsyp postroeniya nadyiozhykhk uzlov i ustroystv TSMV* [The majority principle of building reliable nodes and digital computers]. Moscow, Sov. Radio Publ., 1974. 194 p.
4. Chu P. P. *FPGA prototyping by VHDL examples Xilinx SpartanTM-3 Version*. WILEY & SONS, 2008. 528 p.
5. Etienne Sicard, Sonia Delmas Bendhia. *Advanced CMOS cell design*. Kompaniua McGraw-Hill, 2007. 385 p.
6. Matviyenko M. P. *Proektuvannya tsyfrovyykh prystroyiv* [Design of digital devices]. Kyiv, Lira-K. Publ., 2018. 358 p.
7. Roy S. *Advanced Digital System Design: FPGA and ASIC Implementation*. Ane Books, 2021. 418 p.
8. Ryaben'kyu V. M., Zhuykov V. Ya., Gulyu V. D. *Tsyfrova skhemotekhnika: navchal'nyy posibnyk* [Digital circuit engineering: Textbook]. Lviv, Novyy Svit Publ., 2009. 736 p.
9. Melnyk O. S., Yurchuk I. I. Nanodevices with Programmable Logic. *International scientific journal «Electronics and control systems»*. 2019, vol. 4, no. 62, pp. 47–52. DOI:10.18372/1990-5548.62.14383.
10. Harris D., Harris S. *Digital Design and Computer Architecture*. Morgan K., 2017. 307 p.
11. Bruno F. *Programmirovaniye FPGA dlya nachinayushikh* [FPGA programming for beginners]. Per. s angl. Ed. Pomanov. Moscow, DMK Press Publ., 2022. 320 p.
12. Sochkov A. G. *Elektronika i mikroskhemotekhnika* [Electronics and engineering of microcircuits]. Kyiv, Karavela Publ., 2009. 416 p.
13. Kiltz S. *Advanced FPGA Design: Architecture, Implementation, and Optimization*. Wiley, 2016. 537 p.
14. Grout Ian. *Digital systems design with FPGAs and CPLDs*. Library of Congress Cataloging-in-Publication Data, 2008. 763 p.
15. Micro-Cap 11 Full CD Version 11.2.0.3. Available at: <https://archive.org/details/micro-cap-11-full-cd-version-11.2.0.3>. (accessed 2 September 2023).
16. Lent C. S., Tougaw P. D. A device architecture for computing with quantum dots. *Proc. of the IEEE*. 1997, vol. 85, no. 5, pp. 541–557. DOI: 10.1109/5.573740.
17. Mel'nyk O. S., Kozarevych V. O., Pylypenko R. I. Komp'yuterne proektuvannya aryfmetychnykh nanoprystroyiv [Computer design of arithmetic nanodevices]. *Visnyk NTU «KhPI». Seriya: Matematychnе modelyuvannya v tekhnitsi ta tekhnologiyakh* [Bulletin of the National Technical University "KhPI". Series: Mathematical modeling in engineering and technology]. Kharkiv, NTU «KhPI» Publ., 2014, no. 39 (1082), pp. 126–134.
18. Walus K., Dysart T.J., Jullien G.A., Budiman R.A. QCADesiner: A Rapid Design and Simulation Tool for Quantum-Dot Cellular Automata. *IEEE Transactions on Nanotechnology*. 2004, vol. 3, no. 1, pp. 26–31. DOI:10.1109/TNANO.2003.820815.

Надійшла (received) 17.09.2023

## Відомості про авторів / Сведения об авторах / Information about authors

**Мельник Олександр Степанович** – кандидат технічних наук, доцент, доцент кафедри електроніки, робототехніки і технологій моніторингу та інтернету речей, Національний авіаційний університет, м. Київ; тел.: (067) 213-03-08; e-mail: [oleksandr.melnyk@npp.nau.edu.ua](mailto:oleksandr.melnyk@npp.nau.edu.ua).

**Мельник Александр Степанович** – кандидат технических наук, доцент, доцент кафедры электроники, робототехники и технологий мониторинга и интернета, Национальный авиационный университет, г. Киев; тел.: (067) 213-03-08; e-mail: [oleksandr.melnyk@npp.nau.edu.ua](mailto:oleksandr.melnyk@npp.nau.edu.ua).

**Melnyk Oleksandr Stepanovych** – Candidate of Technical Sciences, Associate Professor, Associate Professor at the Department of Electronics, Robotics and Monitoring of Technologies and Internet of Things, National Aviation University, Kyiv; tel.: (067) 213-03-08; e-mail: [oleksandr.melnyk@npp.nau.edu.ua](mailto:oleksandr.melnyk@npp.nau.edu.ua).

**Козаревич Вікторія Олександрівна** – старший викладач кафедри електроніки, робототехніки і технологій моніторингу та інтернету речей, Національний авіаційний університет, м. Київ; тел.: (063) 310-30-41; e-mail: [viktoriia.kozarevych@npp.nau.edu.ua](mailto:viktoriia.kozarevych@npp.nau.edu.ua).

**Козаревич Викторія Александровна** – старший преподаватель кафедры электроники, робототехники и технологий мониторинга и интернета, Национальный авиационный университет, г. Киев; тел.: (063) 310-30-41; e-mail: [viktoriia.kozarevych@npp.nau.edu.ua](mailto:viktoriia.kozarevych@npp.nau.edu.ua).

**Kozarevych Viktoriia Oleksandrivna** – Senior Lecturer at the Department of Electronics, Robotics and Monitoring of Technologies and Internet of Things, National Aviation University, Kyiv; tel.: (063) 310-30-41; e-mail: [viktoriia.kozarevych@npp.nau.edu.ua](mailto:viktoriia.kozarevych@npp.nau.edu.ua).