УДК 621.382:62-1/-9(045)

О. С. МЕЛЬНИК, В. О. КОЗАРЕВИЧ, А. О. РЕШЕТНИК

ПРОГРАМУВАННЯ МУЛЬТИПЛЕКСОРНИХ МІКРО- ТА НАНОСХЕМ

Застосування великих інтегральних схем (ВІС) в цифрових мікро- і наноелектронних пристроях дозволяє істотно поліпшити їх експлуатаційні можливості, в першу чергу підвищити надійність і швидкодію, понизити споживану потужність і габаритні розміри. Проте розробка ВІС є тривалим і дорогим процесом, який економічно виправданий тільки при досить великому обсязі випуску готових виробів. Підвищення спеціалізації ІС при поліпшенні вказаних вище показників вступає в протиріччя з їх універсальністю, що розширює їх необхідну номенклатуру і зменшує обсяг випуску кожного окремого типономіналу. Останнє веде до подорожчання продукції. Усунути вказане протиріччя між спеціалізацією і універсальністю можна шляхом розробки ВІС, алгоритми роботи яких можуть бути змінені за бажанням розробника конкретної апаратури, тобто, шляхом створення логічних схем, що настроюються або програмуються. Запропоновані автоматизовані методи програмування мультиплексорних мікро- та наносхем для відтворення логічних функцій декількох аргументів. На сучасних комп'ютерних системах було реалізовано моделювання та верифікацію запропонованих пристроїв.

Ключові слова: мікро- та наносхеми з програмованими структурами, мультиплексор, автоматизоване проектування.

Вступ. Застосування великих інтегральних схем (ВІС) в цифрових мікро- і наноелектронних пристроях дозволяє істотно поліпшити їх експлуатаційні можливості, в першу чергу підвищити надійність і швидкодію, понизити споживану потужність і габаритні розміри. Проте розробка ВІС є тривалим і дорогим процесом, який економічно виправданий тільки при досить великому обсязі випуску готових виробів.

Мета роботи. Підвищення спеціалізації при поліпшенні вказаних вище показників завжди вступає в протиріччя з їх універсальністю, що розширює необхідну номенклатуру ВІС і зменшує обсяг випуску кожного окремого типономіналу. Останнє веде до подорожчання продукції.

Усунути вказане протиріччя між спеціалізацією і універсальністю можна шляхом розробки ВІС, алгоритми роботи яких можуть бути змінені користувачем шляхом створення програмованих логічних схем. Тут під програмованістю розуміється не здатність реалізувати заданий алгоритм обробки вхідних кодів, змінюючи програму роботи, як це робить мікропроцесор, а можливість зміни внутрішньої структури ВІС так, щоб вона забезпечувала реалізацію заданих логічних функцій на апаратному рівні.

При виготовленні таких ВІС використовується єдиний технологічний процес, тому з точки зору виробника це універсальні вироби. Налаштування ж ВІС на заданий алгоритм роботи виконує безпосередньо користувач, з точки зору якого ця схема реалізує вузько спеціалізовані завдання. В результаті програмування в ВІС вносяться зворотні або незворотні зміни структури, які і приводять до набуття необхідних характеристик.

Методи програмування мультиплексорів. В якості мікро- та наносхем з програмованими структурами часто використовують мультиплексори, які окрім пристроїв комутації, можуть функціонувати як програмовані логічні елементи. Суть програмування мультиплексора полягає в тому, що його адресні входи використовуються як інформаційні і на них подаються аргументи відтворюваної функції, а інформаційні входи виконують роль програмованих.

© О. С. Мельник, В. О. Козаревич, А. О. Решетник, 2016

При цьому на програмованих входах можуть формуватися сигнали або логічних констант, або деякі допоміжні функції.

Наприклад, реалізуємо на мультиплексорі (4→1) логічну функцію двох аргументів Виключне АБО (додавання по модулю 2 чи нерівнозначності) має вигляд:

$$f_{\oplus} = x_1\bar{x}_0 \vee \bar{x}_1 x_0 = x_1 \oplus x_0. \quad (1)$$

Використовуючи рівняння вихідної функції цього мультиплексори, отримаємо:

$$f = D_3(x_1x_0) \vee D_2(x_1\bar{x}_0) \vee D_1(\bar{x}_1x_0) \vee D_0(\bar{x}_1\bar{x}_0). \quad (2)$$

Порівнюючи між собою функції (1) і (2), запишемо функцію програмування мультиплексора у вигляді:

$$f_{пр} = 0(x_1x_0) \vee 1(x_1\bar{x}_0) \vee 1(\bar{x}_1x_0) \vee 0(\bar{x}_1\bar{x}_0). \quad (3)$$

В результаті, реалізуємо цей алгоритм програмування у вигляді табл. 1 і схеми мультиплексора, побудованої на рис. 1.

Таблиця 1 – Таблиця програмування Логічної функції Виключне АБО

x_1	x_0	$f_{пр}$	D
0	0	0	$D_0 = 0$
0	1	1	$D_1 = 1$
1	0	1	$D_2 = 1$
1	1	0	$D_3 = 0$

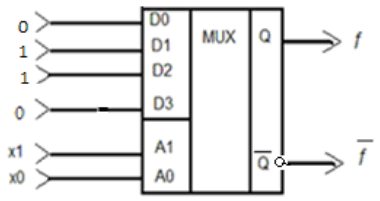


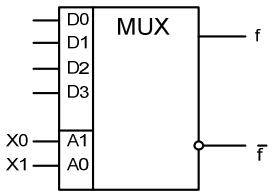
Рис. 1 – Реалізація логічної функції двох аргументів на мультиплексорі.

На мультиплексорі (4→1) можливе відтворення 16 функцій двох аргументів (табл. 2). Для цього на адресні входи A_1 та A_0 слід подавати бінарні аргументи x_1 та x_0, а інформаційні $D_3,...,D_0$ – програмувати логічними константами цих аргументів.

З розглянутого прикладу видно, що описаний метод обмежується реалізацією функцій чотирьох змінних, оскільки мультиплексори, що реально випускаються, мають не більше 16 інформаційних входів.

Таблиця 2 – Програмування 16 функцій двох аргументів на мультиплексорі (4→1)

$$f = D_3(x_1x_0) \vee D_2(x_1\bar{x}_0) \vee D_1(\bar{x}_1x_0) \vee D_0(\bar{x}_1\bar{x}_0).$$



x_1	x_0	f	$\bar{f}$
0	0	D_0	$\bar{D}_0$
0	1	D_1	$\bar{D}_1$
1	0	D_2	$\bar{D}_2$
1	1	D_3	$\bar{D}_3$

X_1	1	1	0	0	Нормальна диз'юнктивна форма	Логічна функція
X_0	1	0	1	0		
№	$D3$	$D2$	$D1$	$D0$		
0	0	0	0	0	$f_0 = 0$	Активний лог.0
1	0	0	0	1	$f_1 = \bar{x}_1\bar{x}_0 = \overline{x_1 \vee x_0}.$	АБО-НІ (стрілка Пірса)
2	0	0	1	0	$f_2 = \bar{x}_1x_0 = x_0 \leftarrow x_1$	Заборона
3	0	0	1	1	$f_3 = \bar{x}_1x_0 = \bar{x}_1\bar{x}_0 = \bar{x}_1(x_0 \vee \bar{x}_0) = \bar{x}_1$	Заперечення x_1
4	0	1	0	0	$f_4 = x_1\bar{x}_0 = x_1 \leftarrow x_0$	Зворотна заборона
5	0	1	0	1	$f_5 = x_1\bar{x}_0 \vee \bar{x}_1\bar{x}_0 = \bar{x}_0(x1 \vee \bar{x}_1) = \bar{x}_0$	Заперечення x_0
6	0	1	1	0	$f_6 = x_1\bar{x}_0 \vee \bar{x}_1x_0 = x_1 \oplus x_0$	Виключне АБО (нерівнозначність)
7	0	1	1	1	$f_7 = x_1\bar{x}_0 \vee \bar{x}_1x_0 \vee \bar{x}_1\bar{x}_0 = \overline{x_1x_0}$	І-НІ (штрих Шеффера)
8	1	0	0	0	$f_8 = x_1x_0$	І
9	1	0	0	1	$f_9 = x_1x_0 = \bar{x}_1\bar{x}_0 = \overline{x_1 \oplus x_0}$	Виключне АБО-НІ (рівнозначність)
10	1	0	1	0	$f_{10} = x_1x_0 \vee \bar{x}_1x_0 = x_0(x_1 \vee \bar{x}_1) = x_0$	Активний аргумент x_0

Продовження таблиці 2						
11	1	0	1	1	$f_{11} = x_1x_0 \vee \bar{x}_1x_0 \vee \bar{x}_1\bar{x}_0 = \bar{x}_1 \vee x_0 = x_1 \to x_0$	Імплікація
12	1	1	0	0	$f_{12} = x_1x_0 \vee x_1\bar{x}_0 = x_1(x_0 \vee \bar{x}_0) = x_1$	Активний аргумент x_1
13	1	1	0	1	$f_{13} = x_1x_0 \vee \bar{x}_1x_0 \vee \bar{x}_1\bar{x}_0 = x_1 \vee \bar{x}_0 = x_0 \to x_1$	Зворотна імплікація
14	1	1	1	0	$f_{14} = x_1x_0 \vee x_1\bar{x}_0 \vee \bar{x}_1\bar{x}_0 = x_0 \vee x_1$	АБО
15	1	1	1	1	$f_{15} = x_1x_0 \vee x_1\bar{x}_0 \vee \bar{x}_1x_0 \vee \bar{x}_1\bar{x}_0 = 1$	Активна логічна 1

На виходах мультиплексора запрограмовано по 8 функцій:

f	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
$\bar{f}$	f_{15}	f_{14}	f_{13}	f_{12}	f_{11}	f_{10}	f_9	f_8

При необхідності реалізації логічної функції великого числа вхідних змінних використовують структуру *мультиплексорного дерева*. Проте при невеликому числі аргументів цю задачу можна вирішити методом вибору сигналів налаштування не з множини $\{1,0\}$, як це було зроблено вище, а з множини $\{1,0,x_i\}$, де x_i – один з аргументів відтворюваної функції. В цьому випадку вдається на мультиплексорі без додаткових апаратних витрат реалізувати логічну функцію, число аргументів якої більше числа його адресних входів.

Приклад моделювання. Реалізуємо на мультиплексорних мікро- та наносхемах (4→1) логічну функцію мажоритарного вибору 2 з 3 аргументів. На виході мажоритарного елемента (МЕ) логічний сигнал співпадає з більшістю сигналів на непарній кількості входів [1]. На рис. 2 показане умовне позначення тривходового МЕ, а в табл. 3 побудована відповідна таблиця істинності.

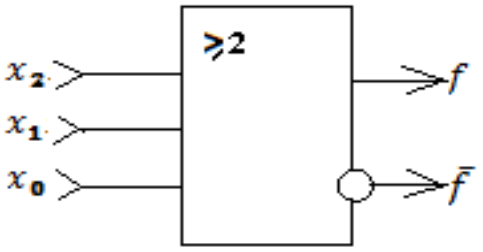


Рис. 2 – Тривходовий МЕ.

Таблиця 3 – Таблиця істинності МЕ

x_2	x_1	x_0	f	$\bar{f}$
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

Диз'юнктивну нормальну форму (ДНФ) мажоритарної функції f отримують з її таблиці істинності:

$$f = maj(x_2, x_1, x_0) = \bar{x}_2x_1x_0 \vee x_2\bar{x}_1x_0 \vee x_2x_1\bar{x}_0 \vee x_2x_1x_0. \quad (4)$$

Виконавши дужкові перетворення рівняння (4) шляхом триразового логічного додавання останнього доданку знаходять довершену форму мажоритарної функції наступного типу:

$$f = maj(x_2, x_1, x_0) = x_2x_1(\bar{x}_0 \vee x_0) \vee x_2x_0(\bar{x}_1 \vee x_1) \vee x_1x_0(\bar{x}_2 \vee x_2) = x_2x_1 \vee x_2x_0 \vee x_1x_0. \quad (5)$$

Мажоритарна функція мінімальної форми (5) є найбільш поширеною в наноелектроніці. Інверсну мажоритарну функцію називають *міноритарною*:

$$\bar{f} = maj(x_2, x_1, x_0) = min(x_2, x_1, x_0) = \bar{x}_2\bar{x}_1 \vee \bar{x}_2\bar{x}_0 \vee \bar{x}_1\bar{x}_0. \quad (6)$$

Якщо один з входів МЕ запрограмувати сигналом $x_2 = 0$ чи $x_2 = 1$, то як видно з табл. 3, на прямому виході МЕ реалізуються логічні функції помноження I ($f_I = x_1x_0$) чи додавання АБО ($f_{АБО} = x_1 \vee x_0$).

Перетворимо повну мажоритарну функцію (4) для програмування мультиплексора (4→1) при використанні в якості адресних аргументів x_1 і x_0. Виконаємо спрощені дужкові перетворення цієї функції:

$$f = maj(\overline{x_2, x_1, x_0}) = x_1x_0(\bar{x}_2 \vee x_2) \vee x_2x_1\bar{x}_0 \vee x_2\bar{x}_1x_0. \quad (7)$$

Порівнюючи вихідну функцію мультиплексора (2) з перетвореною МЕ (7), запишемо функцію програмування мультиплексора (4→1) для реалізації мажоритарної функції:

$$f_{пр}(x_1, x_0) = 1(x_1x_0) \vee x_2(x_1\bar{x}_0) \vee x_2(\bar{x}_1x_0) \vee 0(\bar{x}_1\bar{x}_0). \quad (8)$$

Отже, згідно рівнянню (8) на інформаційні входи мультиплексора слід подавати наступні сигнали:

$$D_3 = 1, \quad D_2 = D_1 = x_2 \text{ та } D_0 = 0.$$

Рішення (8) не є єдино існуючим. На адресні входи мультиплексора можливо подавати довільні набори аргументів, наприклад, (x_2x_0) або (x_2x_1), а на інформаційні входи D_3, – залишкові аргументи та логічні сигнали з

множини $\{1,0\}$. В першому випадку функція програмування матиме вигляд:

$$f_{пр}(x_2,x_0)=1(x_2x_0)\vee x_1(\overline{x}_2\overline{x}_0)\vee x_1(\overline{x}_2x_0)\vee 0(\overline{x}_2\overline{x}_0),$$

а в другому:

$$f_{пр}(x_2,x_1)=1(x_2x_1)\vee x_0(x_2\overline{x}_1)\vee x_0(x_2\overline{x}_1)\vee 0(\overline{x}_2\overline{x}_1).$$

Таким чином, практичне програмування мажоритарної функції (3) на базі мультиплексорів (4→1) свідчить, що при любій комбінації адресних аргументів на виході реалізується однотипні функції програмування.

Результати комп'ютерного моделювання. На рис. 3 побудована функціональна схема запрограмованого мультиплексора в графічному редакторі системи автоматизованого проектування (САПР) MAX+PLUS II, яка є інтегрованим середовищем для розробки цифрових пристроїв на базі програмованих логічних схем (ПЛІС) фірми ALTERA [2]. Вона складається з восьми мікросхем: трьох інверторів НІ, чотирьох тривходових логічних елементів I і одного чотиривходового логічного елемента АБО, має два адресних входи, чотири інформаційних і один комплементарний вихід.

Часовий аналіз вхідних і вихідних сигналів результатів програмування мажоритарної функції (8) наведений у вікні редактора діаграм на рис. 4. Він повністю відповідає таблиці істинності МЕ (табл. 3).

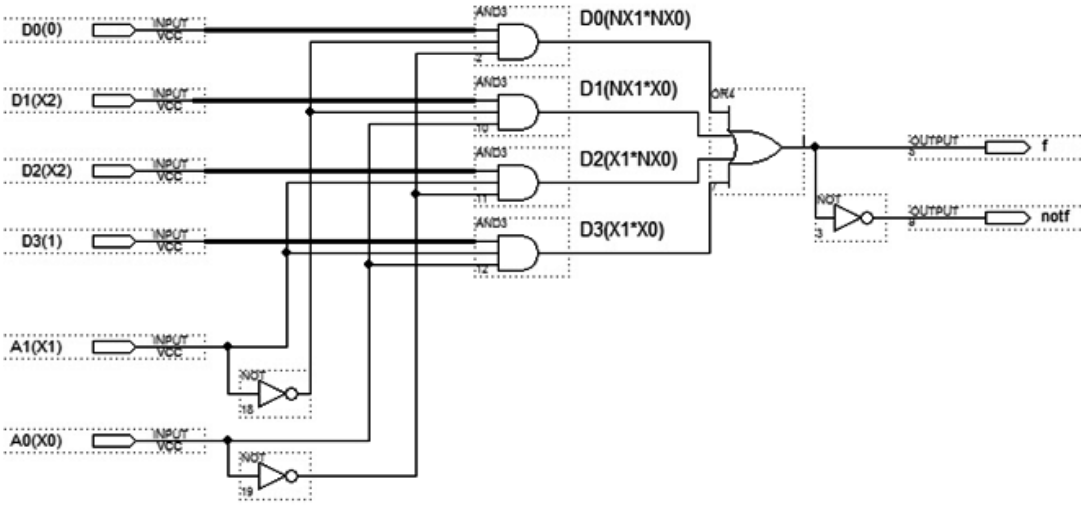


Рис. 3 – Мікросхема запрограмованого мультиплексора.

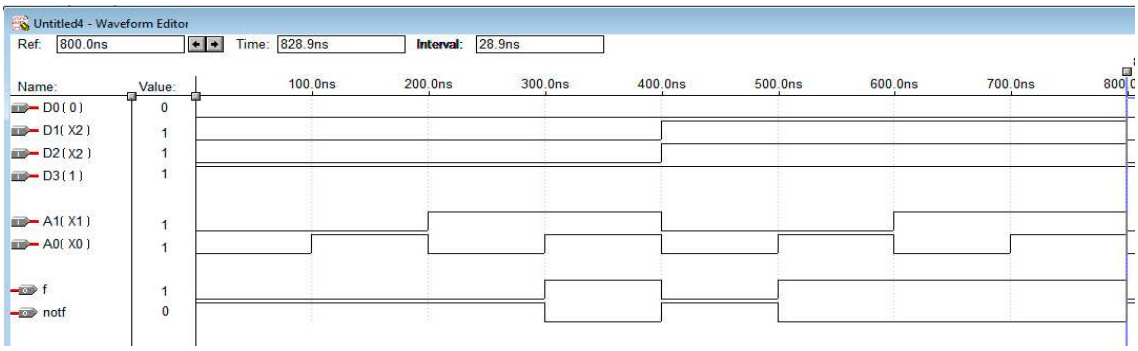


Рис. 4 – Часові діаграми запрограмованого мультиплексора.

Далі синтезуємо одноелектронну наносхему мультиплексора (4→1) на квантових коміркових автоматах (КА). Базова діелектрична комірка КА розміром (50×50) нм містить чотири напівпровідникові чи металеві квантові точки (острівці), які геометрично розміщені по куткам квадрату, як показано на рис. 5. Комірки використовуються для побудови нанопровідників, мажоритарних елементів, інверторів, запам'ятовуючих пристроїв, тощо [3].

Принцип дії КА, заснований на квантовому тунелюванні та кулонівській взаємодії електронів, дозволяє створювати пристрої, які можуть працювати як комутуючі транзистори, але з меншими розмірами та меншою споживаною потужністю. Однорідна архітектура КА може бути сформована за допомогою електронної літографії.

Кожна комірка може містити два надлишкові електрони, які в змозі тунелювати між квантовими точками. Коли потенціал діелектричних перетинок між точками високий, електрони будуть локалізовані та не зможуть тунелювати, а коли потенціал низький електрони зможуть вільно тунелювати. Кулонівське відштовхування між електронами примусить їх розташуватися на діаметрально протилежних кутках з двома можливими конфігура-

ціями (рис. 5). Ці два стабільні стани можуть бути представлені як поляризації комірок $P = -1$ та $P = +1$. Рівень поляризації $P = -1$ відповідає логічному нулю (рис. 5, *а*), а $P = +1$ відповідає логічній одиниці (рис. 5, *б*). Якщо в масиві КА, отриманому після розміщення декількох комірок в ряд, змінити поляризацію першої комірки в визначений стан, то друга комірка прийме той же стан по *принципу найменшої енергії*. Далі по *принципу доміно* всі комірки масиву набудуть цей же рівень поляризації.

Рис. 5 – Базова комірка в двох можливих логічних станах.

Поляризація комірки P визначається співвідношеннями:

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

де ρ_1 і ρ_3, ρ_2 і ρ_4 – щільності зарядів електронів у діагонально розташованих квантових точках комірки.

Таким чином, бінарна інформація передається без руху зарядів. Тобто нема протікання струму між комірками. Це – основна причина, чому структури КА споживають надмізерну кількість енергії $\sim 10^{-22}$ Дж.

На рис. 6 приведена наносхема мультиплексора на 3 інверторах і 11 МЕ, яка запрограмована для реалізації мажоритарної функції (6).

На рис. 7, *а* побудована наносхема на КА запрограмованого мультиплексора (4→1), яка реалізує функцію мажоритарного вибору, а на рис. 7, *б* – часові діаграми, отримані в результаті моделювання на САПР QCA Designer [4]. Сигнали вихідної функції f повністю співпадають з табличними значеннями для МЕ (табл. 3).

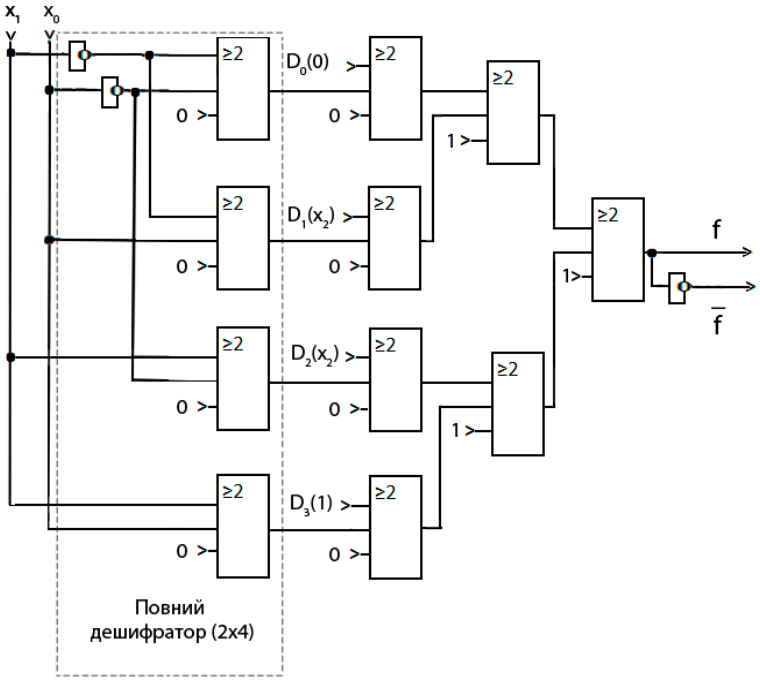
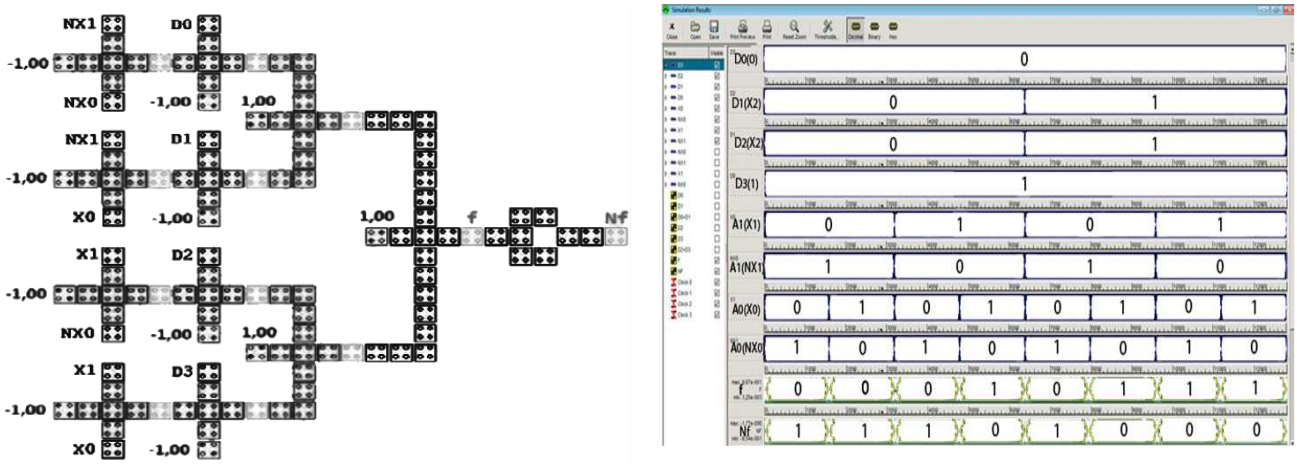


Рис. 6 – Наносхема запрограмованого мультиплексора.

Запрограмована таким чином наносхема мультиплексора (4→1) базується на 114 квантових коміркових автоматах, розмір яких складає (18×18) нм, відстань між їх центрами дорівнює 20 нм, з чотирма квантовими острівцями діаметром по 5 нм. Загальна площа наносхеми мультиплексора дорівнює $(454{,}8 \times 362{,}81)$ нм^2 $\approx$ $\approx 0{,}16$ мкм^2.

Висновки. В роботі розроблені мікро- та наносхеми запрограмованих мультиплексорів для реалізації логічних функцій. Програмування мультиплексора виконувалось за допомогою САПР MAX+PLUS II та САПР QCA Designer. Часові діаграми запрограмованих мультиплексорів в різних середовищах співпали з таблицею істиності.



а *б*

Рис. 7 – Наносхема мультиплексора: *а* – на квантових коміркових автоматах; *б* – її часові характеристики.

Термін "наноелектроніка" логічно пов'язаний з терміном "мікроелектроніка" і відображає перехід сучасної напівпровідникової електроніки від компонентів з характерним розміром в мікроній і субмікроній областях до комірок з розмірами в нанометровій області.

Принципово нова особливість наноелектроніки пов’язана з тим, що для елементів таких розмірів починають переважати квантові ефекти. З’являється нова номенклатура властивостей, відкриваються нові перспективи їх використання. Якщо при переході від мікро- до наноелектроніки квантові ефекти є паразитними, наприклад, роботі класичного транзистора при зменшенні розмірів починає заважати тунелювання носіїв заряду, то електроніка, яка використовує квантові ефекти, – це основа нової, так званої *наногетероструктурної електроніки*.

Список літератури: 1. *Пакулов Н. И.* Мажоритарный принцип построения надежных узлов и устройств ЦВМ. – М. : Сов. радио – 1974, 194 с. **2.** *Стешенко В. Б.* ПЛИС фирмы ALTERA: элементная база, система проектирования и языки описания аппаратуры. – М. : Изд. дом. «Додека – XXI», 2007. – 576 с. **3.** *Мельник О. С., Козаревич В. О., Пилипенко Р. І.* Комп’ютерне проектування арифметичних нанопристроїв // Вісник НТУ «ХПІ». Серія: Математичне моделювання в техніці та технологіях. – Х. : НТУ «ХПІ», 2014 – № 39 (1082). – С. 126 – 134. **4.** *Walus K.* QCADesiner: A CAD Tool for an Emerging Nano-Technology / K. Walus // Micronet Annual Workshop. – Toronto, Canada, 2003, pp. 292 – 294.

References: 1. Pakulov, N. I. *Mazhoritarnyy printsip postroeniya nadyozhnykh uzlov i ustroystv TSVM* [Majority principle of constructing reliable nodes and DC devices]. Moscow, Sov. radio Pabl., 1974. 174 p. **2.** Steshenko, V. B. *PLIS firmy ALTERA: elementnaya baza, systema proektirovaniya i yazyki opisaniya apparatury* [PLIS by ALTERA: circuit technology, design system, and hardware description language]. Moscow, Izd. dom. "Dodeka – XXI" Publ., 2007. 576 p. **3.** Me'lnyk, O. S., Kozarevych, V. O. and Pylypenko, R. I. Komp"yuterne proektyvannya aryfmetychnykh nanoprystroyiv [Computer-aided design of arithmetic nanodevices]. *Visnyk NTU "KhPI" Seriya: Matematychne modelyuvannya v texnici ta texnologiyakh* [Bulletin of National Technical University «KhPI» Series: Mathematical modeling in engineering and technologies]. Kharkiv, NTU «KhPI» Publ., 2014. no. 39 (1082), pp. 126–134. **4**. Walus, K. *QCADesiner: A CAD Tool for an Emerging Nano-Technology*. Micronet Annual Workshop. Toronto, Canada, 2003, pp. 292–294.

Надійшла (received) 24.03.2015

Відомості про авторів / Сведения об авторах / Information about authors

Мельник Олександр Степанович – кандидат технічних наук, доцент, Національний авіаційний університет, м. Київ; тел: (093) 994-52-51; e-mail: melnyk.ols@gmail.com.

Мельник Александр Степанович – кандидат технических наук, доцент, Национальный авиационный университет, г. Киев; тел.: (093) 994-52-51; e-mail: melnyk.ols@gmail.com.

Melnyk Olexandr Stepanovych – Candidate of Engineering Sciences, Associate Professor, National Aviation University, Kyiv; tel.: (093) 994-52-51; e-mail: melnyk.ols@gmail.com.

Козаревич Вікторія Олександрівна – асистент, Національний авіаційний університет, м. Київ; e-mail: st-viktoria@yandex.ru.

Козаревич Виктория Александровна – ассистент, Национальный авиационный университет, г. Киев; e-mail: st-viktoria@yandex.ru.

Kozarevych Viktoriia Oleksandrivna – Assistant Professor, National Aviation University, Kyiv; e-mail: st-viktoria@yandex.ru.

Решетник Анна Олександрівна – студентка, Національний авіаційний університет, м. Київ; e-mail: reshetnik.1994@bk.ru.

Решетник Анна Александровна – студентка, Национальный авиационный университет, г. Киев; e-mail: reshetnik.1994@bk.ru.

Reshetnyk Anna Oleksandrivna – student, National Aviation University, Kyiv; e-mail: reshetnik.1994@bk.ru.