

УДК 04.31

О. К. Тесленко, Г. О. Тарасенко, Я. М. Клятченко

## НЕРОЗДІЛЬНІ БЛОЧНІ 9-РОЗРЯДНІ ЗАВАДОСТІЙКІ КОДИ ДЛЯ ВИПРАВЛЕННЯ ОДНОРАЗОВОЇ ПОМИЛКИ

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ

**Анотація.** З бурхливим розвитком цифрових телекомунікаційних технологій актуалізується використання нових методів для підвищення швидкості та надійності передачі даних. До таких методів можна віднести кодування даних, що засновано на штучному введенні надмірності і дозволяє стороні одержувача не тільки виявляти спотворення переданих даних, а й формувати правильні значення. Роздільні коди (наприклад, коди Хемінга) поступаються нероздільним корегуючим кодам у швидкості. Але підвищення швидкості передачі даних для нероздільних кодів вимагає використання складних, багатомісних алгоритмів для пошуку максимальних кодів. Це збільшує час визначення завадостійких кодів в порівнянні з роздільними кодами, але підвищує швидкість передавання даних в експлуатації при однакових можливостях завадостійкості. Пропонується технологія, яка заснована на використанні 9-розрядних нероздільних блочних кодів та кодеків/декодерів для них, враховуючи, що загальна теорія їх побудови на даний момент відсутня. Враховуючи особливості реалізації спеціалізованих пристроїв на інтегральній технології, найбільш сприятливим є використання технології ПЛІС. Для практичної реалізації універсальних кодеків/декодерів 9-розрядних нероздільних кодів пропонується підхід із використанням ПЛІС, які можуть бути налаштовані на будь-який із цих кодів з будь-якими десятковими цифрами (нероздільні коди не мають інформаційних та перевірочних частин). Такі властивості нероздільних блочних кодів дозволяють збільшити швидкість передачі двійково-кодованих десяткових слів. Також, отримані результати можуть бути підґрунтям для розвитку теорії і практики застосування блочних нероздільних кодів при зростанні розміру блока та можуть бути обнадійливим фактором що стосується аналізу нероздільних кодів у випадку виправлення двох і більше помилок.

**Ключові слова:** корекція помилок, передача даних, булева функція, код Хемінга, нероздільні блочні коди.

**Abstract.** With the rapid development of digital telecommunication technologies, the usage of new methods to increase the speed and reliability of data transfer is becoming more relevant. Examples may include data encoding based on the artificial introduction of redundancy and enabling the receiving side not only to detect data distortions but also to form correct values. The separable codes (for example, the Hamming codes) are inferior to non-separable error correction codes in speed. But the increase in the speed of data transmission requires utilising complex, multipositional algorithms for searching maximal codes. This leads to an increase in the time of static proof codes determination compared to separable codes, although the speed of data transmission in operation is increased as well, having identical capacities for fault resistant. Technology is proposed that is based on using 9-bit non-separable codes and the encoders/decoders for them, factoring in the fact of absence of the generalised theory of building them. Factoring in the aspects of specialised devices implementation based on an integral technology, the usage of FPGA technology is the most opportune. For practical implementation of universal encoders/decoders for 9-bit non-separable codes, an approach is proposed utilising FPGA that can be configured to any of these codes with any decimal numbers (the non-separable codes do not contain informational and test parts). Such properties of non-separable codes enable increasing the transmission speed of BCD words. Additionally, the obtained results could be a basis for the further development of a theory and practice for employing the block inseparable codes when the block size increases and it is a reassuring factor as for the analysis of non-separable codes in case of two or more errors correction.

**Key words:** error-correcting code, data transmission, boolean functions, Hamming code, non-separable block codes.

**DOI:** <https://doi.org/10.31649/1999-9941-2023-56-1-30-34>.

### Вступ

Теоретична швидкість блочних завадостійких кодів визначається як  $R = (\log_2 N_k)/n$ , де  $N_k$  – кількість кодослів у коді,  $n$  – їх розрядність,  $d$  – найменша відстань Хемінга між будь-якими кодословами коду (у випадку виправлення одноразової помилки  $d=3$ ) [1]. Значного розвитку набула теорія та практика побудови блочних роздільних кодів, де існує чіткий розподіл між інформаційними та додатковими бітами. Прикладами таких кодів є коди Хемінга, Боуза-Чоудхурі-Хоквінгема та ін. Перевагою у використанні таких кодів є те, що можна аналітично визначити дозволені комбінації, а також те, що можна порівняно нескладно реалізувати такі кодеки та декодери. У нероздільних блочних кодах надлишковість вводиться шляхом знаходження множини дозволених кодів, в якій між кожною парою кодових слів відстань за Хемінгом не менша ніж  $d$  [2]. В той же час значно менше вивчені нероздільні коди, де не існує розділу на інформаційні та додаткові біти. Перевага нероздільних кодів може полягати в збільшенні швидкості завдяки збільшенню  $N_k$  при одній і тій же розрядності. Так, наприклад, для 8-розрядного коду Хемінга  $N_k = 16$ . В [3] теоретично доведено, що для 8-розрядного нероздільного коду  $N_k = 20$ . Недоліком нероздільних кодів є складність їх побудови, оскільки визначення  $N_k$  пов'язано з пошуком максимальної кліки в графі. Така задача в загальному випадку відноситься до NP – повних задач [4].

### Актуальність

В роботі [5] наведено 8-розрядний нероздільний блочний код на максимальну кількість 20 кодослів в коді, запропонована реалізація на ПЛІС кодеків та декодерів такого коду у випадку передачі послідовності із упакованих двійково-десяткових чисел. Показано, що швидкість передачі всього потоку збільшується на 25% порівняно з 8-розрядними кодами Хемінга. Можна очікувати, що при використанні 9-розрядного нероздільного коду швидкість передачі потоку ще можна збільшити.

### Мета

Метою даної роботи є визначення ефективних рішень із використанням 9-розрядного нероздільного коду для підвищення швидкості передачі потоку.

### Задачі

1. Визначення блочних нероздільних 9-розрядних кодів.
2. Реалізація кодеків/декодерів для цих послідовностей.
3. Оцінка результатів з точки зору підвищення швидкості передачі даних.

### Розв'язання задач

Один із можливих методів визначення таких кодів базується на тому факті, що максимальна кількість кодослів 9-розрядного коду для виправлення однієї помилки дорівнює 40 [5]. Його можна побудувати із двох 8-розрядних нероздільних кодів. Дійсно, зафіксувавши один із розрядів кожного кодослова, ми розіб'ємо 9-розрядний код на два 8-розрядні коди – з 0 або 1 у вибраному розряді. Кожен з цих утворених кодів не може мати більше ніж 20 кодослів [4]. При цьому дистанція між ними не менша за 2. Дистанцію між двома кодами  $C_1(n, N_1, d)$  та  $C_2(n, N_2, d)$  будемо визначати як найменшу відстань між будь-якими кодословами двох кодів. Для пошуку можна скористатись результатами визначення всіх 8-розрядних нероздільних кодів, наведеними в [6]. Загальна кількість різних максимальних 8-розрядних кодів значна – 403200. Перебір всіх можливих пар можна значно скоротити, оскільки в різних кодах є кодослова, що збігаються, тому відстань між ними буде дорівнювати нулю. Наприклад, більша частина всіх 8-розрядних кодів мають в своєму складі нульове кодослово.

Експериментальним шляхом було визначено декілька пар з відстанню між кодами, що дорівнює 2. Приклад однієї із таких пар наведений в табл. 1.

Таблиця 1 – Варіанти кодослів

0	7	25	30	42	53	75	84	108	115	140	147	161	175	182	184	197	218	226	253
13	18	35	60	70	72	81	95	101	122	138	149	164	187	195	220	233	238	240	247

Побудова 9-розрядного коду полягає в установці дев'ятого розряду в кодословах одного із 8-розрядних кодів в 0 і в 1 – в іншому.

Із результату по табл.1 можна отримати наступні два максимальні 9-розрядні коди на 40 кодослів (табл. 2 та табл. 3).

Таблиця 2 – Перший 9-розрядний код  $C(9,40,3)$

0	7	25	30	42	53	75	84	108	115	140	147	161	175	182	184	197	218	226	253
---	---	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Таблиця 2 (продовження)

269	274	291	316	326	328	337	351	357	378	394	405	420	443	451	476	489	494	496	503
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Таблиця 3 – Другий 9-розрядний код  $C(9,40,3)$

13	18	35	60	70	72	81	95	101	122	138	149	164	187	195	220	233	238	240	247
----	----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Таблиця 3 (продовження)

256	263	281	286	298	309	331	340	364	371	396	403	417	431	438	440	453	474	482	509
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Для реалізації кодеків та декодерів для кодів  $C(9,40,3)$  розглянемо наступне. Як і в [5], будемо розглядати передавання потоку з упакованих двійково-десяткових чисел. Позначимо через  $j$  номер чергового байту послідовності двійково-десяткових чисел ( $j \in \{1, 2, \dots\}$ ). Сорок кодових слів обумовлюють встановлення порядку для кодування двійково-десяткових цифр та двох бітів наступної цифри, як наведено в таблиці 4.

Таблиця 4 – Порядок кодування

Цифра $2_j$				Цифра $2_{j-1}$			
Біт 3	Біт 2	Біт 1	Біт 0	Біт 3	Біт 2	Біт 1	Біт 0

Кодування залишку попереднього байту та наступного байту згідно з табл. 5.

Таблиця 5 – Порядок кодування залишку

Цифра $2_{j+1}$				Цифра $2_j$			
Біт 3	Біт 2	Біт 1	Біт 0	Біт 3	Біт 2	Біт 1	Біт 0

Кодек може бути реалізований як комбінаційна схема (КС1) на шість входів і дев'ять виходів, для чого знадобиться лише 9 блоків LUT FPGA [7]. Кожен з блоків LUT можна налаштувати на реалізацію булевої функції від 6 змінних, які створюються згідно з налаштуваннями, як показано в табл.6.

Таблиця 6 – Налаштування ПЛІС для реалізації кодека C(9,40,3)

In	00	01	02	03	04	05	06	07	08	09
Out	0	7	25	30	42	53	75	84	108	115
In	10	11	12	13	14	15	16	17	18	19
Out	140	147	161	175	182	184	197	218	226	253
In	20	21	22	23	24	25	26	27	28	29
Out	269	274	291	316	326	328	337	351	357	378
In	30	31	32	33	34	35	36	37	38	39
Out	394	405	420	443	451	476	489	494	496	503

На інших входах булеві функції невизначені, і їм можуть бути задані будь-які значення з метою оптимізації по інших критеріях. Реалізований таким чином кодек можна використати для кодування як згідно табл. 4, так і табл. 5. Формування окремого кодека по табл. 5 є недоцільним.

Декодер, що може бути реалізовано згідно з табл. 4, виглядатиме як комбінаційна схема (КС2) на дев'ять входів та сім виходів. Із них 6 виходів-для значень розрядів результату декодування. Вони визначаються по табл. 4, якщо входи і виходи поміняти місцями та за умови відсутності помилок. Таким порядком визначаються булеві функції лише на 40 із 512 можливих вхідних даних. На інших  $Nk \cdot n = 360$  вхідних даних булеві функції визначаються як результат виправлення одноразових помилок. Приклад такого визначення наведено в табл. 7.

Таблиця 7 – Декодування значень в результаті одноразової помилки

Input: значення в результаті одноразової помилки в кодослові 0									Output
256	128	64	32	16	8	4	2	1	00
Input: значення в результаті одноразової помилки в кодослові 7									
263	135	71	39	23	15	3	5	6	01
...									
Input: значення в результаті одноразової помилки в кодослові 269									Output
13	397	333	301	285	261	265	271	268	29
...									
Input: значення в результаті одноразової помилки в кодослові 496									Output
240	368	432	464	480	504	500	498	241	38
Input: значення в результаті одноразової помилки в кодослові 503									Output
247	375	439	471	487	511	499	501	502	39

Решта вхідних значень можуть з'явитись лише в результаті двох і більше помилок. На сьомому виході КС2 з'являється ознака багатократних помилок при передаванні даних по каналу зв'язку. Булева функція від 9 змінних на сьомому виході приймає значення 0 на вхідних даних згідно з порядком в табл. 6 і табл. 7. На інших наборах вхідних даних ця функція приймає значення 1 як ознаку багатократних спотворень, які виявляє декодер. Для побудови декодера згідно з табл.5 ця функція додатково приймає значення 1 і на вхідних даних, заданих відповідно до останнього рядка табл. 6.

Реалізація декодера на ПЛІС дещо складніша, ніж реалізація кодека. Використовуючи розклад Шеннона, будь-яку булеву функцію від 9 змінних можна реалізувати не більше ніж на трьох слайсах (slices) сучасних ПЛІС FPGA. На двох слайсах реалізуються дві будь-які функції від 8 змінних, а на третьому –

функція від 9 змінних. Таким чином, знадобиться не більше ніж 21 слайс, що значно менше за їх загальну кількість в одній мікросхемі.

Далі виконаємо оцінку швидкості передавання даних. Теоретично перевага у швидкості запропонованого 9-розрядного нероздільного коду перед 9-розрядним кодом Хемінга з п'ятьма інформаційними та чотирма додатковими розрядами очевидна при передаванні будь-яких даних. Дійсно, показник теоретичної швидкості для 9-розрядного нероздільного коду  $R=(\log 2Nk)/n = (\log 240)/9$  – не менше 0.59, а 9-розрядного коду Хемінга – не більше 0.56. Але у випадку передавання послідовностей десяткових чисел ця перевага збільшується.

Зауважимо, що показник теоретичної швидкості для нероздільного 8-розрядного коду [5]- 0.54.

Нехай необхідно передавати потік із  $s$  десяткових чисел або  $4s$  біт. Для забезпечення стійкості від одиничних спотворень послідовність  $4s$  біт розбивається на блоки по 6 біт, які потім кодуєм нероздільного 9-ти розрядного коду перетворюються у 9-бітні блоки. Об'єм даних для передавання –  $(4s/6)*9 = 6s$  біт.

У випадку 9-розрядного коду Хемінга маємо –  $(4s/5)*9 = 7,2s$  біт. Отже, об'єм переданих даних збільшується у 1,2 рази.

### Висновки

1. Таким чином, збільшення швидкості передачі даних для нероздільних кодів вимагає використання трудомістких алгоритмів пошуку максимальних кодів. Це збільшує час визначення завадостійких кодів в порівнянні з роздільними кодами, але підвищує швидкість передавання даних в експлуатації при однакових можливостях завадостійкості.

2. Ефективне застосування ПЛІС-технологій дозволяє значно скоротити витрати на практичну реалізацію кодеків та декодерів, що є надійним підґрунтям застосування таких нероздільних кодів. Порівняно незначні за кількістю витрати елементів дозволяють отримати прийнятні результати для засобів передачі інформації на ПЛІС, що входять до складних пристроїв обробки інформації.

3. Одержані результати є підґрунтям для розвитку теорії і практики застосування блочних нероздільних кодів при зростанні розміру блока. Подальші дослідження стосуються аналізу нероздільних кодів у випадку виправлення двох і більше помилок.

### Список літератури

- [1] Richard E. Blahut, *Theory and practice of error control codes*. Addison-Wesley Pub. Co. edition, in English, 1983.
- [2] Elwyn R. Berlekamp (2014), *Algebraic Coding Theory*, World Scientific Publishing (revised edition).
- [3] Maryna S. Viazovska, "The sphere packing problem in dimension 8", *Annals of Mathematics SECOND SERIES*, vol. 185, no. 3 (May, 2017), pp. 991-1015.
- [4] Johnston, H. C. (1976), "Cliques of a graph—variations on the Bron–Kerbosch algorithm", *International Journal of Parallel Programming*, 5 (3): 209–238.
- [5] Y. Klyatchenko, O. Tarasenko-Klyatchenko, G. Tarasenko, O. Teslenko, "The Problems and Advantages of Using Non-separable Block Codes", *Lecture Notes on Data Engineering and Communications Technologies this link is disabled*, Springer, April 2022, 134, pp. 271–278. [https://doi.org/10.1007/978-3-031-04812-8\\_23](https://doi.org/10.1007/978-3-031-04812-8_23).
- [6] Звіт про науково-дослідну роботу МЕТОДИ ОЦІНКИ ТА ЗАБЕЗПЕЧЕННЯ НЕОБХІДНОГО РІВНЯ ТЕХНІЧНОЇ БЕЗПЕКИ РОБОТИ СПЕЦІАЛІЗОВАНИХ БАГАТОПРОЦЕСОРНИХ СИСТЕМ УПРАВЛІННЯ. Науковий керівник Романкевич О. М. № держреєстрації 0115U000323, 2017, 158 с.
- [7] Summary of Virtex-6 FPGA Features. Virtex-6 Family Overview. XILINX DS150 (v2.5) August 20, 2015.

Стаття надійшла: 06.02.2023.

### References

- [1] Richard E. Blahut, *Theory and practice of error control codes*. Addison-Wesley Pub. Co. edition, in English, 1983.
- [2] Elwyn R. Berlekamp (2014), *Algebraic Coding Theory*, World Scientific Publishing (revised edition).
- [3] Maryna S. Viazovska, "The sphere packing problem in dimension 8", *Annals of Mathematics SECOND SERIES*, vol. 185, no. 3 (May, 2017), pp. 991-1015.
- [4] Johnston, H. C. (1976), "Cliques of a graph—variations on the Bron–Kerbosch algorithm", *International Journal of Parallel Programming*, 5 (3): 209–238.
- [5] Y. Klyatchenko, O. Tarasenko-Klyatchenko, G. Tarasenko, O. Teslenko, "The Problems and Advantages of Using Non-separable Block Codes", *Lecture Notes on Data Engineering and Communications Technologies this link is disabled*, Springer, April 2022, 134, pp. 271–278. [https://doi.org/10.1007/978-3-031-04812-8\\_23](https://doi.org/10.1007/978-3-031-04812-8_23).

- [6] Report on research work METHODS FOR EVALUATION AND PROVISION OF THE REQUIRED LEVEL OF TECHNICAL SAFETY OF SPECIALISED MULTIPROCESSOR CONTROL SYSTEMS. Supervisor: Romankevich O.M. State registration number 0115U000323, 2017, 158 p. [in Ukrainian].
- [7] Summary of Virtex-6 FPGA Features. Virtex-6 Family Overview. XILINX DS150 (v2.5) August 20, 2015.

#### **Відомості про авторів**

**Тесленко Олександр Кирилович** – кандидат технічних наук, доцент кафедри системного програмування і спеціалізованих комп'ютерних систем.

**Тарасенко Георгій Олегович** – аспірант кафедри системного програмування і спеціалізованих комп'ютерних систем.

**Клятченко Ярослав Михайлович** – кандидат технічних наук, доцент кафедри системного програмування і спеціалізованих комп'ютерних систем.

O. K. Teslenko, G. O. Tarasenko, Ya. M. Klyatchenko

## **NON-SEPARABLE BLOCK 9-BIT FAULT RESISTANT SINGLE ERROR CORRECTION CODES**

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv