

Реалізація алгоритму недвійкових первинних кодів за допомогою Google Sheets

УДК 004.056

Діана Желізняк¹, Наталія Загородна², Кирил Шеханін³*Тернопільський національний технічний університет імені Івана Пулюя,**¹dianajeliznyk@gmail.com, ²zagorodna_n@ntu.edu.ua,**³kyryl.shekhanin@outlook.com*

У сучасних системах передачі та зберігання інформації ключову роль відіграють методи кодування, які забезпечують ефективність, завадостійкість та оптимальне використання наявних ресурсів. Традиційно найбільш поширеними є двійкові коди, що базуються на алфавіті з двох символів – «0» та «1». Однак у багатьох випадках застосування недвійкових первинних кодів, які використовують алфавіт потужністю більше двох ($q > 2$), відкриває додаткові можливості для підвищення інформаційної щільності та швидкості обробки даних. Недвійкові первинні коди є системами кодування, де кожен символ може нести більше одного біта інформації, що дозволяє скоротити довжину кодових комбінацій при передаванні того ж обсягу повідомлень.

Теоретичні засади недвійкового кодування базуються на потужності алфавіту q , довжині кодової комбінації n та загальній кількості можливих комбінацій, яка обчислюється за формулою $N_0 = q^n$. Наприклад, для трійкового коду ($q = 3$) або четвіркового ($q = 4$) кількість доступних кодових слів зростає експоненційно порівняно з двійковим представленням тієї ж довжини. Важливою характеристикою є кодова відстань, зокрема відстань Хеммінга, яка визначає здатність коду виявляти та виправляти помилки, що виникають під час передачі даних через канали зв'язку із завадами. Недвійкові первинні коди класифікуються за можливістю виявлення помилок (безнадмірні та надмірні), за структурою (блокові та неперервні), а також за рівномірністю (рівномірні та нерівномірні). Історично передумови виникнення таких кодів сягають ще XIX століття, коли Семюель Морз створив телеграфний код, що використовував три символи (крапку, тире та паузу), а згодом Еміль Бодо розробив 5-бітний код для телеграфії, який дозволяв передавати 32 різні символи.

Галузь використання недвійкових первинних кодів досить широка. Вони активно застосовуються в сучасних телекомунікаційних системах, зокрема в мобільному зв'язку стандартів 4G/5G, де використовуються багаторівневі схеми квадратурної амплітудної модуляції (QAM) з базами 4, 16, 64 та вище. У комп'ютерній техніці недвійкове кодування реалізоване у флеш-пам'яті типу MLC (Multi-Level Cell), де одна комірка зберігає два або більше бітів інформації. У квантових обчисленнях кубіти можуть перебувати в суперпозиції станів, що також потребує недвійкового представлення даних. Крім того, такі коди знаходять застосування в оптичних лініях зв'язку, системах супутникового зв'язку, а також у задачах штучного інтелекту для оптимізації обробки великих масивів даних. Переваги недвійкового кодування полягають у підвищенні інформаційної ємності, зменшенні апаратних витрат при зберіганні даних та можливості створення більш гнучких алгоритмів корекції помилок. Однак існують і недоліки: складність реалізації, підвищена чутливість до шумів у

каналі зв'язку, потреба в точнішій апаратурі для розрізнення багатьох рівнів сигналу, а також труднощі із синхронізацією на високих швидкостях передачі.

Реалізація алгоритму недвійкових первинних кодів може бути легко автоматизована за допомогою табличних процесорів, таких як Google Sheets або Microsoft Excel, де використовуються вбудовані функції BIN2DEC для перетворення двійкових груп у десяткові числа та DEC2BIN для зворотного перетворення. Для практичної демонстрації алгоритму кодування та декодування недвійкових первинних кодів було обрано Google Sheets (рис. 1) та систему числення з основою $q=16$ (шістнадцяткове кодування), оскільки вона є поширеним компромісом між інформаційною щільністю та простотою технічної реалізації. Як вхідні дані використано довільну 16-бітову двійкову послідовність: 1101011001110101. Процес кодування передбачає розбиття вихідної бітової стрічки на групи по 4 біти, оскільки для бази $q=16$ один символ несе 4 біти інформації ($\log_2 16 = 4$). В результаті отримано чотири групи: 1101, 0110, 0111, 0101. Кожну таку групу перетворено у десяткове число за правилами двійково-десяткового перетворення: $1101_2 = 13_{10}$, $0110_2 = 6_{10}$, $0111_2 = 7_{10}$, $0101_2 = 5_{10}$. Таким чином, вихідне 16-бітне повідомлення було представлено у вигляді кодового слова, що складається з чотирьох символів шістнадцяткової системи: 13, 6, 7, 5. Це кодове слово є більш компактним порівняно з двійковим оригіналом, оскільки замість 16 двійкових символів використовується лише 4 символи з алфавіту потужністю 16. На етапі декодування виконується зворотне перетворення. Для кожного символу кодового слова (13, 6, 7, 5) за допомогою відповідної таблиці відповідності або обчислювальної формули відновлюється його 4-бітне двійкове представлення: $13 \rightarrow 1101$, $6 \rightarrow 0110$, $7 \rightarrow 0111$, $5 \rightarrow 0101$. Отримані бітові групи конкатенуються у вихідну послідовність 1101011001110101, що повністю збігається з початковими даними, підтверджуючи коректність роботи алгоритму.

Вхідні дані	Групи бітів	Символ у базі $q=16$
1101011001110101	1101	13
	0110	6
	0111	7
	0101	5

Рис. 1. Приклад алгоритму недвійкових первинних кодів завдяки Google Sheets

Такий підхід забезпечує наочність і дозволяє швидко обробляти великі обсяги даних без потреби у спеціалізованому програмному забезпеченні. Узагальнюючи, недвійкові первинні коди, зокрема на базі системи числення з основою 16, є потужним інструментом для підвищення ефективності передавання та зберігання інформації, а їх практична реалізація не потребує надмірних обчислювальних ресурсів.

Подальші дослідження можуть бути спрямовані на порівняння завадостійкості недвійкових кодів з двійковими в каналах з різним рівнем шуму, а також на розробку адаптивних алгоритмів кодування, які динамічно змінюють основу системи числення залежно від умов передачі.