

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра електроніки і комп'ютерної техніки

«До захисту допущено»

Завідувач кафедри

_____Анатолій ОПАНАСЮК

(підпис)

_____ 2023р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня бакалавра

зі спеціальності 172 «Телекомунікації та радіотехніка»,
освітньо-професійної програми «Мережеві та інтернет
технології»

На тему: Засоби кодування на основі двійкових біноміальних
чисел для застосування в телекомунікаційних системах

Здобувачки групи ТК-91 Усик Анни Олексіївни

Кваліфікаційна робота містить результати власних
досліджень. Використання ідей, результатів і текстів інших авторів
мають посилання на відповідне джерело.

_____ Анна УСИК
(підпис)

Керівник, старший викладач,
кандидат фізико-математичних наук , доцент

_____Олексій Д'ЯЧЕНКО
(підпис)

Суми – 2023

АНОТАЦІЯ

Дипломна робота присвячена розробці та обґрунтуванню алгоритму та структурної схеми пристрою, що перетворює двійкові числа в біноміальні з використанням методів кодування. Робота складається з трьох основних розділів.

У вступному розділі надається загальне введення в тему роботи та пояснюється актуальність та значущість дослідження. Розглядаються основні мета та завдання телекомунікацій, а також теоретичні основи кодування на основі біноміальних чисел. Також зазначається потенціал використання біноміальних чисел у нових методах кодування та їх практичні застосування в телекомунікаційних системах.

Перший розділ присвячений розробці алгоритму та структурної схеми пристрою, що перетворює двійкові числа в біноміальні. Виконується огляд літератури та постановка задачі проектування. Розглядаються теоретичні аспекти кодування на основі біноміальних чисел і проводиться дослідження можливості їх використання у нових методах кодування. Також розглядаються практичні застосування методів кодування на основі біноміальних чисел у телекомунікаційних системах. Подальше розробляється пристрій з використанням біноміального лічильника, алгоритму роботи та структурної схеми.

Другий розділ присвячений розробці та розрахунку принципової схеми пристрою. Виконується вибір та обґрунтування елементної бази пристрою, а також розробка біноміального лічильника та лічильників з заданим коефіцієнтом перерахунку. Проводиться синтез регістрів зберігання та розглядаються логічні елементи серії K1533, які використовуються у пристрої.

Третій розділ присвячений моделюванню програми телекомунікаційної системи з використанням методів кодування на основі біноміальних чисел. Висновки роботи включають коротке підбиття підсумків та висвітлення результатів дослідження.

Загалом, дипломна робота представляє важливий внесок у галузь телекомунікацій та кодування на основі біноміальних чисел. Результати дослідження можуть бути використані для подальшого розвитку телекомунікаційних систем та оптимізації процесу кодування.

ЗМІСТ

| | |
|---|----|
| Вступ..... | 4 |
| 1 Розробка та обґрунтування алгоритму функціональної та структурної схеми пристрою, що проектується. | 6 |
| 1.1 Огляд літератури та постановка задачі проектування. | 6 |
| 1.1.1 Телекомунікації. Мета та завдання телекомунікацій..... | 6 |
| 1.1.2 Теоретичні основи кодування на основі біноміальних чисел. | 11 |
| 1.1.3 Дослідження можливості використання біноміальних чисел у нових методах кодування. | 14 |
| 1.1.4 Практичні застосування методів кодування на основі біноміальних чисел у телекомунікаційних системах. | 19 |
| 1.2 Розробка пристрою, що перетворює двійкові числа в біноміальні. Алгоритм роботи та структурна схема. | 26 |
| 1.3 Розробка функціональної схеми | |
| 2 Розробка та розрахунок принципової схеми пристрою. | 36 |
| 2.1 Вибір та обґрунтування елементної бази пристрою | 39 |
| 2.2 Розробка біноміального лічильника | 41 |
| 2.3 Розробка лічильників з заданим коефіцієнтом перерахунку..... | 48 |
| 2.4 Синтез регістрів зберігання | 51 |
| 2.5 Логічні елементи серії К1533..... | 52 |
| 3 Моделювання програми телекомунікаційної системи з використанням методів кодування на основі біноміальних чисел | 54 |
| Висновки. | 57 |
| Додатки. | 58 |
| Список використаної літератури. | 61 |

| | | | | | | | |
|-----------|------|---------------|--------|------|---------------------------------|------|---------|
| | | | | | ЕЛІТ 6.172.00.02. 370 ПЗ | | |
| Змн. | Арк. | № докум. | Підпис | Дата | | | |
| Розроб. | | Усик А.О. | | | Літ. | Арк. | Акрушіє |
| Перевір. | | Д'яченко О.В. | | | 3 | | |
| Н. Контр. | | | | | СумДУ, гр. ТК-91 | | |
| Затверд. | | Опанасюк А.С. | | | | | |

ВСТУП

У сучасному світі телекомунікаційні системи відіграють важливу роль в забезпеченні зв'язку між людьми та організаціями на відстані. Проте, під час передачі даних можуть виникати різні помилки, що можуть призвести до некоректної роботи системи. Одним зі способів боротьби з помилками є використання методів кодування на основі двійкових біноміальних чисел.

Мета дослідження:

Метою даної дипломної роботи є дослідження засобів кодування на основі двійкових біноміальних чисел для застосування в телекомунікаційних системах. У результаті дослідження буде проведений аналіз різних методів кодування на основі біноміальних чисел, їх ефективності та застосування в реальних телекомунікаційних системах. Також у ході дипломної роботи буде розроблений перетворювач двійкових чисел в біноміальні. Буде приведений ефективний алгоритм роботи перетворювача чисел і структурна схема його реалізації.

Завдання дослідження:

1. Описати принципи кодування на основі двійкових біноміальних чисел.
2. Провести аналіз різних методів кодування на основі біноміальних чисел та порівняти їх ефективність.
3. Дослідити можливості використання біноміальних чисел у нових методах кодування.
4. Розглянути практичні застосування методів кодування на основі біноміальних чисел у телекомунікаційних системах.
5. Виконати розробку методу синтезу перетворювачу двійкових чисел в біноміальні.

Методи дослідження:

Для досягнення поставленої мети будуть використані наступні методи дослідження:

1. Аналіз літературних джерел, що описують принципи та методи кодування на основі біноміальних чисел.
2. Експериментальне порівняння ефективності різних методів кодування на основі біноміальних чисел з використанням різних метрик, таких як швидкість передачі даних, рівень помилок та інші.

3. Дослідження можливості використання біноміальних чисел у нових методах кодування, що не були досліджені раніше, з використанням математичного аналізу та експериментального порівняння з іншими методами кодування.

4. Аналіз практичних застосувань методів кодування на основі біноміальних чисел у телекомунікаційних системах, зокрема в мережах передачі даних, бездротових мережах та інших.

5. Моделювання пристрою, що базується на методі перетворення двійкових чисел в біноміальні, який використовується для синтезу завадостійкого кодуючого пристрою.

Застосування цих методів дослідження дозволить отримати глибоке розуміння принципів та методів кодування на основі біноміальних чисел, порівняти їх ефективність та визначити можливості їх використання у реальних телекомунікаційних системах.

Також, використання цих методів дослідження дозволить провести комплексний аналіз методів кодування на основі біноміальних чисел, описати їх принципи та переваги, порівняти ефективність різних методів та визначити їх можливості в реальних умовах. Моделювання роботи телекомунікаційних систем з використанням методів кодування на основі біноміальних чисел дозволить провести практичні експерименти та перевірити результати дослідження на практиці. В цілому, застосування різних методів дослідження дозволить детально вивчити проблему та знайти оптимальні рішення для використання методів кодування на основі біноміальних чисел у телекомунікаційних системах.

1 РОЗРОБКА ТА ОБГРУНТУВАННЯ АЛГОРИТМУ ФУНКЦІОНУВАННЯ ТА СТРУКТУРНОЇ СХЕМИ ПРИСТРОЮ, ЩО ПРОЕКТУЄТЬСЯ

1.1 Огляд літератури та постановка задачі проектування

1.1.1 Телекомунікації. Мета та завдання телекомунікацій. Зв'язок та передача інформації є важливими аспектами нашого сучасного світу, і саме цими питаннями займаються телекомунікації. Телекомунікації охоплюють всі технології, системи та послуги, що дозволяють передавати інформацію на відстань, забезпечуючи зв'язок між віддаленими точками.

Основна мета телекомунікацій - забезпечити ефективний обмін даними, голосом та відео між людьми, пристроями та комп'ютерами на великій відстані. Це включає передачу голосу по телефону, обмін повідомленнями, передачу відео, доступ до Інтернету, телевізійне мовлення, радіо та багато іншого.

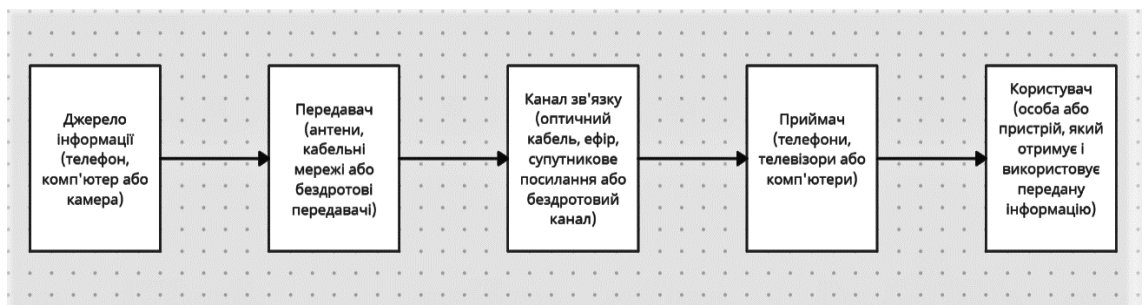


Рисунок 1.1 - Компоненти телекомунікаційної системи передачі інформації

Телекомунікації розв'язують різні задачі:

1. **Забезпечення зв'язку:** Телекомунікаційні мережі дозволяють спілкуватися на великі відстані. Вони забезпечують передачу голосу, повідомлень та даних між людьми через телефони, мобільні пристрої, комп'ютери тощо.

2. **Доступ до інформації:** Телекомунікаційні системи надають можливість швидкого доступу до різноманітної інформації. Інтернет, наприклад, дозволяє отримати доступ до веб-сторінок, електронної пошти, соціальних мереж, онлайн-відео, електронних документів тощо.

3. Передача даних: Телекомунікаційні мережі передають значні обсяги даних між комп'ютерами та іншими пристроями. Це дозволяє підтримувати роботу корпоративних мереж, передавати електронні повідомлення, відправляти файли, використовувати хмарні сервіси та багато іншого.

4. Мультимедіа та розваги: Телекомунікації грають важливу роль у сфері мультимедіа та розваг. Це включає передачу телевізійного сигналу, радіо, відео, онлайн-ігор, музики та іншого контенту.

5. Безпека та надійність: Телекомунікаційні системи також забезпечують засоби для забезпечення безпеки та надійності передачі даних. Це включає шифрування інформації, захист від несанкціонованого доступу, резервне копіювання даних та інші заходи для збереження цілісності та конфіденційності даних.

Загалом, телекомунікації є ключовим елементом сучасного світу, що дозволяє людям і пристроям ефективно спілкуватися та обмінюватися інформацією на великій відстані. Вони вирішують завдання передачі даних, забезпечення зв'язку, доступу до інформації та розваг, а також гарантують безпеку та надійність цих процесів.

Для забезпечення надійності та безпеки комунікаційних процесів використовується таке поняття як достовірність інформації. Воно відноситься до здатності системи передавати та отримувати дані без помилок, зберігати їх цілісність і забезпечувати автентичність інформації. Забезпечення достовірності даних в телекомунікаційних системах вимагає використання різноманітних технологій та протоколів, а також дотримання спеціальних вимог.

Одним з ключових аспектів забезпечення достовірності інформації є захист від несанкціонованого доступу та зміни даних під час передачі. Для досягнення цієї мети використовуються різні методи шифрування та автентифікації. Шифрування даних дозволяє забезпечити конфіденційність інформації шляхом перетворення її в криптографічний код, який може бути розшифрований лише з використанням спеціального ключа. Автентифікація в свою чергу дозволяє перевірити, чи є відправник даних справжнім та підтвердити цілісність інформації.

Для забезпечення достовірності інформації в телекомунікаційних системах важливо враховувати такі вимоги:

1. Конфіденційність: Інформація повинна бути захищена від несанкціонованого доступу або перехоплення. Використання шифрування даних та захищених каналів передачі дозволяє забезпечити конфіденційність.

2. Цілісність: Інформація повинна бути передана без змін або спотворень. Застосування механізмів контролю цілісності, таких як хеш-функції або цифрові підписи, допомагає перевірити, чи були дані змінені під час передачі.

3. Автентичність: Повідомлення або джерело інформації повинні бути перевірені, щоб впевнитися у їхній справжності та надійності. Механізми автентифікації, такі як цифрові сертифікати або системи автентифікації з використанням паролів, забезпечують перевірку автентичності.

4. Надійність: Дані повинні бути захищені від несанкціонованого видалення або знищення. Застосування резервного копіювання, архівації та систем забезпечення надійності допомагають забезпечити надійність даних.

5. Доступність: Інформація повинна бути доступною для користувачів у відповідний час. Забезпечення надійності мережевої інфраструктури, відмовостійкості та налагодження систем моніторингу допомагають забезпечити доступність інформації.

Ці вимоги до достовірності інформації в телекомунікаційних системах є основними для забезпечення надійності, конфіденційності та безпеки комунікаційних процесів. Враховуючи ці вимоги, телекомунікаційні системи можуть забезпечити надійну передачу та зберігання інформації.

Ключовим елементом телекомунікаційної системи є кодування, яке дозволяє ефективно і надійно передавати дані через різні канали зв'язку. Для глибшого розуміння принципів та механізмів передачі інформації необхідно знати про методи кодування, які застосовувались у минулому, які використовуються зараз та в чому їх різниця.

У телекомунікаційних системах, які були використані раніше, було застосовано різні коди з метою ефективної передачі та збереження інформації. Деякі з них, які були важливими в історії телекомунікацій, включають наступні:

1. Азбука Морзе: Розроблена Семюелом Морзе, ця система кодування використовує комбінації коротких і довгих сигналів (крапок і тире) для представлення букв, цифр і пунктуаційних знаків. Азбука Морзе широко використовувалась в телеграфних системах.

2. Код Vaudot: Це п'ятибітний код, розроблений Жаном-Морісом Еміліо Бодо, що використовувався для передачі текстової інформації в телетайпних системах. Код Vaudot дозволяв передавати літери, цифри та символи.

3. Код Грея: Це особливий тип двійкового коду, у якому два сусідні значення відрізняються лише одним бітом. Це забезпечувало зменшення можливих помилок передачі при використанні коду Грея. Він широко використовувався в цифрових системах передачі даних.

4. Код ASCII: Американський стандартний код для обміну інформацією (ASCII) використовує семибітні коди для представлення букв, цифр, спеціальних символів і керуючих символів. Код ASCII став основою для подальших стандартів кодування, таких як UTF-8, які використовуються сьогодні.

5. Код EBCDIC: Розширений двійковий код обміну інформацією з розширеним набором символів (EBCDIC) був використаний в старих системах обробки даних, зокрема комп'ютерах IBM. Цей код використовувався для кодування букв, цифр та спеціальних символів.

Ці коди використовувались в різних телекомунікаційних системах у минулому. Вони були розроблені з метою ефективною передачі, представлення та збереження інформації у відповідних контекстах свого часу.

В телекомунікаційних системах зараз використовуються різноманітні коди залежно від конкретного застосування. Ось декілька прикладів кодів, які широко використовуються:

1. Код Манчестера: Це нелінійний код, в якому кожен біт поділяється на дві періоди рівної тривалості. Логічне значення "1" представляється штрихом, а значення "0" — прогалиною. Цей код використовується, наприклад, в мережах Ethernet для передачі даних.

2. Код Геммінга: Це блочний код для виявлення та виправлення помилок передачі даних. Він використовується для забезпечення надійності передачі даних у різних системах, включаючи мережі зв'язку та сховища даних.

3. Коди корекції помилок Ріда-Соломона: Ці коди використовуються для виявлення та корекції помилок передачі даних. Вони широко застосовуються в оптичних та бездротових комунікаційних системах, де шум та спотворення можуть впливати на якість сигналу.

4. Коди BCH (Боуза-Чоудхурі-Хоквінгема): Ці коди використовуються для виявлення та корекції помилок передачі даних. Вони широко

використовуються в супутникових комунікаціях, оптичних системах передачі та флеш-пам'яті.

5. Коди згортки: Ці коди використовуються для декодування та виявлення помилок у передачі даних. Вони широко застосовуються в мобільних комунікаціях, супутникових комунікаціях та зв'язку на великі відстані.

Це лише кілька прикладів кодів, які використовуються в сучасних телекомунікаційних системах. Вибір конкретного коду залежить від вимог до надійності передачі даних, швидкості передачі та затримки.

Зробимо порівняльний аналіз чотирьох кодів: кодів БЧХ, коду Грея, коду EBCDIC та коду корекції помилок Ріда-Соломона.

1. Коди БЧХ (Боуза-Чоудхурі-Хоквінгема):

- Використовуються для виявлення та корекції помилок передачі даних.

- Широко застосовуються в супутникових комунікаціях, оптичних системах передачі та флеш-пам'яті.

- Забезпечують високу ефективність у виявленні та корекції помилок.

- Дозволяють виявити та виправити багатобітні помилки.

2. Код Грея:

- Використовується для зменшення можливості виникнення помилок при передачі даних.

- Забезпечує мінімальну кількість змін у значеннях коду при зміні одного біта.

- Використовується в цифрових системах зв'язку, вимірювальних пристроях та інших пристроях з обміном даними.

3. Код EBCDIC (Extended Binary Coded Decimal Interchange Code):

- Використовується для кодування символів, чисел та спеціальних знаків.

- Застосовується в старіших комп'ютерних системах та системах зв'язку.

- Забезпечує широку підтримку різних символів та мов.

4. Коди корекції помилок Ріда-Соломона:

- Використовуються для виявлення та корекції помилок передачі даних.

- Широко використовуються в оптичних та бездротових комунікаційних системах.

- Забезпечують високу надійність у виявленні та корекції помилок, включаючи багатобітні помилки.

Загалом, коди БЧХ забезпечують високу надійність у виявленні та корекції помилок, код Грея зменшує можливість помилок при передачі даних, код EBCDIC використовується для кодування символів, а коди Ріда-Соломона забезпечують високу надійність у виявленні та корекції помилок, включаючи багатобітні помилки. Вибір конкретного коду залежить від вимог конкретної системи та її потреб у надійності, швидкості та ефективності передачі даних. [1,3]

1.1.2 Теоретичні основи кодування на основі біноміальних чисел.

Біноміальні числа - це числа, які виникають в результаті розкладу в степінь бінома $(a + b)^n$, де n - ціле додатне число, а a та b - будь-які числа. Біноміальне число $C(n, k)$ - це коефіцієнт при степені $a^{(n-k)} * b^k$ в розкладі бінома $(a + b)^n$. Вираження $C(n, k)$ називається біноміальним коефіцієнтом.

Основні властивості біноміальних чисел:

- Симетричність: $C(n, k) = C(n, n-k)$
- Сума всіх біноміальних коефіцієнтів в розкладі бінома $(a + b)^n$ дорівнює 2^n
- Трикутник Паскаля: біноміальні коефіцієнти утворюють трикутник, в якому кожен елемент це сума двох елементів з попереднього рядка. Наприклад:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

- Формула Паскаля: $C(n, k) = C(n-1, k-1) + C(n-1, k)$

Біноміальні числа знайшли своє застосування в багатьох галузях математики та прикладних наук. Наприклад, вони використовуються в теорії ймовірності для обчислення ймовірностей випадкових подій та у комбінаториці для розв'язання різних задач про розташування об'єктів.

Також біноміальні числа мають важливе застосування в теорії рядів, де вони використовуються для розкладання функцій у ряди Тейлора. Крім того, біноміальні числа зустрічаються у властивостях бінарних дерев та графів, в

теорії чисел та криптографії, теорії ігор та багатьох інших галузях математики та прикладних наук.

Одним із найвідоміших прикладів застосування біноміальних чисел є теорема Бернуллі, яка відома своїм застосуванням в теорії ймовірностей та статистиці. Ця теорема встановлює, що якщо p - ймовірність того, що випадкова подія відбудеться в одній спробі, а n - кількість спроб, то ймовірність того, що випадкова подія відбудеться k разів у n спробах, обчислюється за допомогою біноміальних коефіцієнтів:

$$P = C_n^k = C_n^{n-k} = \frac{n!}{k!(n-k)!}$$

де $P(k)$ - ймовірність того, що випадкова подія відбудеться k разів у n спробах.

Іншим прикладом застосування біноміальних чисел є формула для розкладання $(a + b)^n$ у біноміальний ряд:

$$(a + b)^n = C_n^0 * a^n * b^0 + C_n^1 * a^{(n-1)} * b^1 + \dots + C_n^k * a^{(n-k)} * b^k + \dots + C_n^n * a^0 * b^n,$$

де $C(n,k)$ - біноміальний коефіцієнт, що розглядається як кількість способів вибрати k елементів з n елементів. Ця формула знайшла застосування в теорії ймовірностей, комбінаториці та теорії чисел.

Біноміальні числа також застосовуються в теорії кодування, де вони використовуються для створення біноміальних кодів. Біноміальний код - це спосіб кодування повідомлень, в якому повідомлення поділяється на частини, які кодуються окремо. Кожна частина кодується за допомогою біноміального числа, і кодові слова вибираються зі списку можливих слів, що відповідає відповідному біноміальному числу.

Методи кодування на основі біноміальних чисел - це клас методів кодування, які використовують комбінаторні властивості біноміальних коефіцієнтів для побудови ефективних кодів. Ці методи використовуються в теорії і практиці передачі даних, криптографії, телекомунікаціях, а також в інших областях інформатики та математики.

Існують різні методи кодування на основі біноміальних чисел, такі як:

1. Кодування за допомогою біноміальних коефіцієнтів - це метод кодування, який використовує біноміальні коефіцієнти для кодування повідомлення. В цьому методі повідомлення кодується у вигляді комбінації біноміальних коефіцієнтів, які визначаються за допомогою формули біноміального коефіцієнта. Цей метод може бути використаний для кодування дискретних повідомлень.

2. Кодування за допомогою біноміального дерева - це метод кодування, який використовує біноміальне дерево для представлення повідомлення. В цьому методі повідомлення розбивається на деяку кількість частин, які кодуються за допомогою біноміального дерева. Кожна частина повідомлення представлена як вершина біноміального дерева, а дочірні вершини визначають кодові символи. Цей метод може бути використаний для кодування повідомлень будь-якого типу.

3. Кодування за допомогою біноміального степеневого розкладу - це метод кодування, який використовує біноміальний степеневий розклад для представлення повідомлення. В цьому методі повідомлення розбивається на підряди фіксованої довжини, які кодуються за допомогою біноміального степеневого розкладу. Кожен підряд представляється як сума біноміальних коефіцієнтів зі змінними степенями, а змінні степені визначають кодові символи. Цей метод може бути використаний для кодування дискретних повідомлень.

Кожен з цих методів має свої переваги та недоліки, і використовується залежно від контексту та вимог застосування.

Кодування за допомогою біноміальних коефіцієнтів є простим та ефективним методом для кодування дискретних повідомлень, але його застосування обмежене наявністю обмеженого числа символів, що можуть бути використані для кодування.

Кодування за допомогою біноміального дерева може бути використано для кодування повідомлень будь-якого типу, але його застосування вимагає більш складних алгоритмів для генерації кодових символів, що може відобразитися на часі кодування та декодування.

Кодування за допомогою біноміального степеневого розкладу також може бути використано для кодування дискретних повідомлень, але воно має обмеження на довжину підрядів, які можуть бути кодовані.

Загалом, методи кодування на основі біноміальних чисел є одним із способів розв'язання проблеми передачі та збереження інформації, і вони використовуються в різних областях, таких як телекомунікації, комп'ютерні мережі, криптографія та інші.[2 , 4]

1.1.3 Дослідження можливості використання біноміальних чисел у нових методах кодування

Метод кодування на основі біноміального індексу:

У методі використовуються біноміальні числа для створення унікального коду для кожного елемента множини.

Для початку, нехай ми маємо множини елементів $\{a_1, a_2, \dots, a_n\}$. За допомогою біноміальних коефіцієнтів ми можемо обчислити кількість підмножин множини, які містять рівно k елементів. Це дає нам можливість присвоїти кожній підмножині унікальний індекс, який буде використовуватися як код.

Щоб обчислити індекс підмножини, необхідно взяти, скільки підмножин множини містять менше елементів, ніж дана підмножина. Це можна обчислити за допомогою наступного рекурентного співвідношення:

$$\text{index} = 0 \text{ for } i \text{ from } 0 \text{ to } k-1: \text{index} += C(n-i-1, k-i-1)$$

де $C(n,k)$ - біноміальний коефіцієнт, який обчислюється за формулою $C(n,k) = n! / (k! * (n-k)!)$.

Цей алгоритм працює наступним чином: на початку ми присвоюємо індекс 0. Потім ми проходимо по всіх можливих значеннях i від 0 до $k-1$ і додаємо до індексу кількість підмножин множини, які містять менше елементів, ніж поточна підмножина. Це обчислюється за допомогою біноміального коефіцієнту $C(n-i-1, k-i-1)$, оскільки ми вибираємо $k-i-1$ елементів з невибраних елементів $n-i-1$.

Наприклад, якщо ми маємо множини $\{a,b,c,d\}$ і хочемо знайти індекс для підмножини $\{a,c,d\}$, то:

- Індекс підмножини, яка містить менше елементів, ніж підмножина $\{a,c,d\}$, є сума біноміальних коефіцієнтів $C(3,0) + C(2,1) = 1 + 2 = 3$.
- Отже, індекс для підмножини $\{a,c,d\}$ - це 3.

Індекси підмножин можна використовувати як унікальні коди для елементів множини. Наприклад, якщо ми маємо множини $\{a,b,c,d\}$ і хочемо закодувати елемент c , то можемо використовувати індекс підмножини, яка містить цей елемент, як унікальний код. У цьому випадку, c належить до трьох підмножин: $\{c\}$, $\{a,c\}$ та $\{a,b,c,d\}$, тому його код буде 2, оскільки це індекс підмножини $\{a,c\}$.

За допомогою методу кодування на основі біноміального індексу можна швидко виконувати різноманітні операції з множинами, такі як перетин, об'єднання, додавання та видалення елементів. Цей метод часто використовується в програмуванні для роботи з бітовими масками та в алгоритмах, що пов'язані з обробкою підмножин.

Приклад кодування на основі біноміального індексу:

Нехай ми маємо наступний алфавіт: A, B, C, D, E, F, G, H, I, J

І нам потрібно закодувати послідовність: J, B, G, I, E

Для цього спочатку потрібно знайти біноміальний індекс для кожного символу послідовності. Для цього ми можемо скористатися формулою: $index = C(n-1, k-1)$

де n - кількість символів у алфавіті, k - позиція символу у послідовності.

Таким чином, ми знаходимо біноміальні індекси нашої послідовності: J - $C(9, 4) = 126$ B - $C(9, 1) = 9$ G - $C(9, 6) = 84$ I - $C(9, 8) = 9$ E - $C(9, 4) = 126$

Тепер нам потрібно перетворити ці числа у двійкову форму. Для цього можна використати алгоритм перетворення десяткового числа у двійкове, або ж використати вбудовану функцію у програмувальних мовах.

Отже, наша закодована послідовність буде мати наступний вигляд: 126 - 1111110 9 - 1001 84 - 1010100 9 - 1001 126 - 1111110

Тепер ми можемо скласти ці бінарні числа у одну бінарну послідовність: 111111010011010101001001111110

Ця послідовність є закодованою формою нашої вихідної послідовності J, B, G, I, E.

Метод кодування на основі біноміальних співвідношень:

Це метод кодування, що використовується для компактного представлення множин об'єктів. Цей метод полягає в тому, що об'єкти кодуються за допомогою їх біноміального співвідношення.

Біноміальне співвідношення - це число, що визначає кількість способів вибрати k об'єктів з n об'єктів без урахування порядку вибірки. Це число позначається як " n по k " і записується як " $C(n,k)$ " або " n choose k ".

Наприклад, для $n = 5$ та $k = 2$, $C(5,2) = 10$, оскільки є 10 способів вибрати 2 об'єкти з 5: $\{1,2\}$, $\{1,3\}$, $\{1,4\}$, $\{1,5\}$, $\{2,3\}$, $\{2,4\}$, $\{2,5\}$, $\{3,4\}$, $\{3,5\}$, $\{4,5\}$.

Загальний алгоритм роботи методу виглядає так:

1. Вхід: Отримати вхідну послідовність символів або блоків символів.

2. Розбиття на блоки: Розбити вхідну послідовність на блоки фіксованої довжини.

3. Обчислення біноміальних співвідношень: Для кожного блоку обчислити біноміальне співвідношення, використовуючи формулу комбінаторики для визначення кількості можливих комбінацій символів у блоку.

4. Кодування: Перетворити числові значення біноміальних співвідношень у відповідні кодові слова. Це може включати переведення числових значень у бітові послідовності або використання спеціальних символів для кодування.

5. Вихід: Отримати вихідні кодові слова, які представляють блоки вхідних символів.

6. Передача або збереження: Передати або зберегти вихідні кодові слова.

7. Розкодування: При отриманні кодових слів розкодувати їх назад у вихідні символи за допомогою зворотного процесу, який включає в себе декодування кодових слів, відновлення числових значень біноміальних співвідношень та відновлення вхідних символів.

Цей алгоритм відображає послідовність операцій, які можуть бути виконані для реалізації методу кодування на основі біноміальних співвідношень. Варто зазначити, що конкретні кроки та реалізація алгоритму можуть варіюватися залежно від використовуваного коду та його застосування.

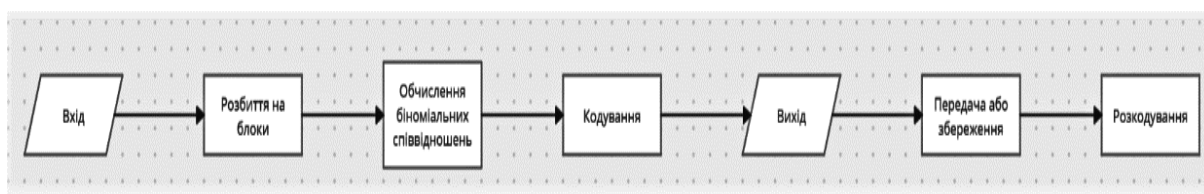


Рисунок 1.2 - Алгоритм роботи методу біноміальних співвідношень

Для застосування методу кодування на основі біноміальних співвідношень об'єкти спочатку нумеруються послідовно від 0 до $N-1$. Потім кожен об'єкт кодується як вектор, де i -й елемент вказує, чи входить i -й об'єкт до множини. Для кодування застосовується формула біноміального співвідношення: k -й елемент вектора містить значення $C(n-1-k, m-1-i)$, де n - кількість об'єктів, m - розмір множини, k - індекс об'єкта, а i - кількість об'єктів, що вибрані до i -го об'єкта.

Наприклад, якщо ми маємо 5 об'єктів (0, 1, 2, 3, 4) і потрібно закодувати множину {1, 3, 4}, то код буде наступним:

- для об'єкта 0: $C(4, 2) = 6$
- для об'єкта 1: $C(3, 1)$

Для об'єкта 1, який є другим елементом у множині $\{1, 3, 4\}$, ми використовуємо формулу біноміального співвідношення $C(3,1)$, що дорівнює 3. Тому другий біт у коді буде встановленим, тобто 1.

Для об'єкта 2: $C(2,1) = 2$

Для об'єкта 3: $C(1,1) = 1$

Для об'єкта 4: $C(0,1) = 0$

Тому код для множини $\{1, 3, 4\}$ буде 10011.

У загальному випадку, для множини розмірності k , яка містить n елементів, кодування на основі біноміальних співвідношень дає послідовність довжини n з бітами, які можуть бути встановленим (1) або скинутим (0). Для кожного i -го елемента у вхідній множині ми використовуємо біноміальне співвідношення $C(n-i, k-i)$, щоб визначити, чи потрібно встановити біт у коді на позиції i .

Метод кодування на основі біноміальних співвідношень зазвичай застосовується для ефективного кодування послідовностей бінарних символів з високою ймовірністю виникнення одного з символів.

Також метод може бути використаний для створення унікальних ідентифікаторів для елементів множини, якщо порядок елементів у множині не має значення. Наприклад, якщо ми маємо множину $\{a, b, c, d\}$, то за допомогою методу кодування на основі біноміальних співвідношень можна створити унікальні ідентифікатори для кожного елемента: $a - 001$, $b - 010$, $c - 011$, $d - 100$.

Цей метод є досить простим і ефективним, але має деякі обмеження, зокрема він не дозволяє кодувати множини з більш ніж 64 елементами (якщо використовувати 64-бітовий тип даних для зберігання коду). Також варто зазначити, що біноміальні коефіцієнти можуть займати досить багато місця в пам'яті, якщо множина містить багато елементів.

Для прикладу розглянемо кодування множини $\{0, 2, 4\}$ за допомогою методу кодування на основі біноміальних співвідношень.

Кількість елементів в множині: $n = 5$

Для кожного об'єкта i в множині обчислюємо значення біноміального коефіцієнта:

для об'єкта 0: $C(4,0) = 1$ для об'єкта 1: $C(4,1) = 4$ для об'єкта 2: $C(4,2) = 6$
для об'єкта 3: $C(4,3) = 4$ для об'єкта 4: $C(4,4) = 1$

Таким чином, ми отримали наступну послідовність бінарних символів: 100101. Ця послідовність є кодом множини $\{0, 2, 4\}$ за допомогою методу кодування на основі біноміальних співвідношень.

Для декодування множини з таким кодом потрібно знайти всі позиції, на яких стоїть символ "1" і відповідної об'єкти з множини будуть мати відповідні індекси. Таким чином, в даному прикладі ми отримаємо множину $\{0, 2, 4\}$. [8]

Розглянувши ці два методи кодування ми можемо зробити порівняння. Метод кодування на основі біноміальних співвідношень та метод кодування на основі біноміального індексу - це два різних підходи до кодування довгих послідовностей бінарних символів. Обидва методи можуть бути ефективними в певних випадках, але мають свої переваги та недоліки.

Метод кодування на основі біноміальних співвідношень використовує біноміальні коефіцієнти для генерації послідовності бінарних символів. Цей метод може бути ефективним для довгих послідовностей з високою ймовірністю виникнення одного з символів. Однак, його недоліком є те, що кожен символ кодується окремо, що може збільшити загальний обсяг даних.

Метод кодування на основі біноміального індексу, з іншого боку, використовує біноміальні індекси для закодування множин об'єктів. Цей метод може бути ефективним для множин з великою кількістю об'єктів, оскільки він дозволяє ефективно закодувати всю множину з одним символом. Однак, його недоліком є те, що він може бути менш ефективним для множин з невеликою кількістю об'єктів.

Обидва методи кодування на основі біноміальних чисел можуть бути застосовані для ефективного кодування дуже довгих послідовностей бінарних символів з високою ймовірністю виникнення одного з символів. Однак, метод кодування на основі біноміального індексу використовує інший підхід до кодування, який може бути більш ефективним у певних випадках.

Метод кодування на основі біноміального індексу генерує кодові слова, використовуючи біноміальні коефіцієнти, але кодує послідовності не за їхнім порядком, а за їхнім індексом у підмножині. Це означає, що для кодування підмножини з меншою кількістю елементів знадобиться менше кодових слів, що може знизити загальний обсяг даних, що потрібно для кодування.

Метод кодування на основі біноміальних співвідношень також використовує біноміальні коефіцієнти для генерації кодових слів, але кодує послідовності за їхнім порядком. Це може бути менш ефективним, оскільки

кількість кодових слів, які потрібні для кодування, залежить від порядку елементів у послідовності.

Отже, обидва методи мають свої переваги та недоліки, і вибір методу залежить від конкретної задачі та особливостей даних, що потрібно закодувати.

1.1.4 Практичні застосування методів кодування на основі біноміальних чисел у телекомунікаційних системах. Методи кодування на основі біноміальних чисел мають широке застосування в системах передачі даних. Вони можуть бути використані для стиснення даних перед їх передачею, що дозволяє зменшити обсяг передаваних даних і зменшити час передачі.

Також можливе використання для захисту передаваних даних від помилок. Коди, створені за допомогою біноміальних чисел, можуть бути детектовані та відновлені в разі помилок, що зменшує ризик втрати даних під час їх передачі.

Крім того, можна створити кодові таблиці, що дозволяє ефективно зберігати та передавати дані, що мають складну структуру. Наприклад, метод кодування на основі біноміального індексу може бути використаний для створення кодових таблиць, які дозволяють ефективно зберігати інформацію про навчальні курси та предмети, що вивчаються в університеті. Кодові таблиці можуть містити інформацію про кожен курс, його назву, опис, кількість кредитів та інші характеристики.

Також метод кодування на основі біноміальних співвідношень може бути використаний для стиснення текстових даних перед їх передачею. Код, створений за допомогою біноміальних співвідношень, може зменшити обсяг передаваних даних, зберігаючи при цьому всю необхідну інформацію.

Методи кодування на основі біноміальних чисел мають широкі застосування в системах передачі даних, як для стиснення даних, так і для захисту даних від помилок. Вони застосовуються у багатьох системах передачі даних, таких як мережі передачі даних, бездротові комунікаційні системи, супутникові зв'язки тощо. Вони забезпечують ефективне кодування та передачу даних з високою швидкістю та точністю.

Наприклад, у бездротових комунікаційних системах метод кодування на основі біноміального індексу може бути використаний для кодування даних перед відправкою через радіоканал. Завдяки високій ефективності цього

методу, його застосовують у багатьох стандартах бездротових комунікацій, таких як Bluetooth, Wi-Fi, ZigBee, NFC та ін.

У системах передачі даних також використовуються методи кодування на основі біноміальних співвідношень. Наприклад, у сучасних системах супутникового зв'язку вони використовуються для кодування телевізійних сигналів та передачі інших даних з супутників на землю.

Таким чином, методи кодування на основі біноміальних чисел мають велике значення у різноманітних системах передачі даних, де вимагається ефективність та точність передачі.

Ще однією областю використання методів кодування на основі біноміальних чисел є криптографія. Біноміальні числа використовуються у шифруванні повідомлень та ключів, а також для генерації псевдовипадкових послідовностей.

Для прикладу, одним з відомих алгоритмів шифрування, що використовує біноміальні числа, є RSA (Rivest–Shamir–Adleman). RSA є криптографічним алгоритмом, що базується на складності розкладу чисел на прості множники.

RSA використовує два великих простих числа, щоб згенерувати публічний ключ, який може бути використаний для шифрування повідомлень, та приватний ключ, який може бути використаний для розшифрування повідомлень. Для генерації публічного та приватного ключів RSA використовує біноміальні числа.

Отже, методи кодування на основі біноміальних чисел мають великий потенціал у багатьох областях, включаючи системи передачі даних та криптографію.

Методи кодування на основі біноміальних чисел також застосовуються у системах зберігання даних, зокрема, в різних форматах архівування даних.

Один із прикладів використання методу кодування на основі біноміальних чисел у системах зберігання даних - це алгоритм кодування Huffman. Huffman compression використовує методи бінарного кодування, в основі яких лежить використання біноміальних співвідношень. Цей метод базується на статистичному аналізі вхідних даних і використовує ймовірності виникнення символів для створення оптимальних кодів.

Іншим прикладом є алгоритм кодування RLE (Run-Length Encoding), який також використовує методи біноміального кодування. Цей алгоритм

зменшує обсяг даних, кодуючи послідовності з однакових символів у пару - символ та кількість його повторів.

Також методи кодування на основі біноміальних чисел застосовуються в деяких системах зберігання даних для забезпечення інтегритету та відновлення даних при їх пошкодженні або втраті. Наприклад, в таких системах як RAID (Redundant Array of Inexpensive Disks) застосовуються різні методи кодування, зокрема, методи кодування на основі біноміальних чисел, для забезпечення збереження даних при відмові одного або кількох дисків у системі.

Одним з прикладів використання методу кодування на основі біноміальних чисел у системах зберігання даних є технологія RAID (Redundant Array of Independent Disks), яка дозволяє зберігати дані на декількох жорстких дисках, забезпечуючи надійність збереження і доступ до даних навіть у випадку відмови одного з дисків.

У технології RAID використовуються різні рівні (levels) кодування на основі біноміальних чисел. Наприклад, у рівні RAID 5 дані розбиваються на блоки, які записуються на різні диски, при цьому для кожного блоку обчислюється біт контролю парності на основі біноміального числа. Цей біт дозволяє відновити дані у випадку відмови одного з дисків.

Інший приклад використання методів кодування на основі біноміальних чисел у системах зберігання даних - це збереження даних на магнітних стрічках. Для забезпечення більшої надійності збереження і можливості відновлення даних у випадку пошкодження стрічки використовуються різні методи кодування на основі біноміальних чисел, такі як коди Ріда-Соломона (Reed-Solomon codes). Ці коди дозволяють додавати додаткові біти контролю, які дозволяють відновлювати дані навіть у випадку великих пошкоджень стрічки.

Застосування методів кодування на основі біноміальних чисел може бути корисним у системах зберігання даних, таких як жорсткі диски, флеш-накопичувачі, карти пам'яті та інші. Одним з прикладів використання цих методів у системах зберігання даних є кодування Хемінга.

Кодування Хемінга - це метод кодування, який використовує біноміальні співвідношення для додавання додаткових бітів до даних, щоб виявити та виправити помилки в передачі. Наприклад, при записі даних на жорсткий диск можуть виникати помилки при передачі даних, такі як біти, які перетворюються на зернистість або повертаються назад. Кодування Хемінга

дозволяє виявляти та виправляти ці помилки, забезпечуючи більш надійне зберігання даних.

Крім кодування Хемінга, існують інші методи кодування на основі біноміальних чисел, такі як кодування Ріда-Соломона та кодування Боуза-Чоудхурі-Хоквінгема. Вони також широко використовуються у системах зберігання даних для забезпечення більш надійного зберігання та передачі даних.

Кодування Ріда-Соломона (англ. Reed-Solomon coding) - це метод кодування, який використовується для захисту даних від пошкоджень при передачі по ненадійним каналам зв'язку або зберіганні на ненадійних носіях інформації, таких як жорсткі диски або флеш-пам'ять.

Кодування Ріда-Соломона базується на математичних концепціях теорії груп та поліномів. Коди Ріда-Соломона є нерівномірними, що дозволяє їх використовувати для корекції декількох пошкоджень. Кодування Ріда-Соломона є частиною багатьох стандартів зв'язку, таких як стандарти передачі даних на супутникових системах зв'язку, Wi-Fi, Bluetooth і т.д.

Для створення коду Ріда-Соломона спочатку визначається параметр t , який відповідає за кількість пошкоджень, які можуть бути скориговані. Потім вибирається поліном, що має степінь $n-1$, де n - це кількість символів, яку ми хочемо закодувати. Цей поліном є головним компонентом коду Ріда-Соломона. Далі генерується набір символів, які складаються з $n-1$ елементів. Цей набір символів використовується для створення контрольних символів, які додаються до оригінального набору символів для створення закодованого повідомлення.

Після створення закодованого повідомлення його можна передавати по ненадійним каналам зв'язку або зберігати на ненадійних носіях. Якщо повідомлення буде пошкоджено, то за допомогою контрольних символів, які були додані до повідомлення, можна скоригувати пошкодження. Якщо кількість пошкоджень перевищує можливість корекції коду Ріда-Соломона, то повідомлення вважається незрозумілим.

Крім того, код Ріда-Соломона може бути використаний не тільки для забезпечення надійності передачі даних, але й для збереження даних на дисках, відеокасетах, CD та DVD дисках, флеш-картах та інших носіях інформації. Такі коди дозволяють зберігати дані на ненадійних носіях з високою надійністю і скоригувати можливі пошкодження при зчитуванні.

Узагальнюючи, код Ріда-Соломона є ефективним і надійним методом для забезпечення надійності передачі даних по ненадійним каналам зв'язку або зберігання даних на ненадійних носіях з високою надійністю. Цей код широко використовується в різних галузях, де потрібна надійна передача або збереження даних.

Кодування Боуза-Чоудхурі-Хоквінгема (або кодування BCH) є ефективним методом корекції помилок при передачі даних по ненадійним каналам зв'язку. Цей метод використовує алгоритм Ріда-Соломона, який ми розглядали раніше.

Кодування BCH є вдосконаленим варіантом кодування Ріда-Соломона, який дозволяє збільшити швидкість передачі даних і зменшити кількість контрольних символів, не втрачаючи при цьому можливості коригувати помилки.

Основна ідея кодування BCH полягає у використанні спеціальних генераторних многочленів, які створюють контрольні символи для даних. Генераторний многочлен є многочленом, який містить корені деякого поля Галуа і має найменшу можливу степінь, щоб забезпечити потрібний рівень корекції помилок.

Закодоване повідомлення складається з даних та контрольних символів, які створюються за допомогою генераторного многочлена. При отриманні повідомлення, якщо воно було пошкоджено, то можна застосувати алгоритм декодування для відновлення оригінального повідомлення з урахуванням контрольних символів.

Кодування BCH використовується в багатьох системах передачі даних, зокрема у сучасних технологіях зберігання даних, таких як жорсткі диски і флеш-накопичувачі. Воно також використовується у бездротових технологіях, таких як Wi-Fi і Bluetooth.

Методи кодування на основі біноміальних чисел можуть бути корисними в системах керування телекомунікаційними мережами. Зокрема, їх можна використовувати для передачі керуючої інформації, наприклад, для керування потоками даних або для керування ресурсами мережі.

Одним з прикладів використання методів кодування на основі біноміальних чисел у системах керування телекомунікаційними мережами є розробка протоколу управління ресурсами мережі з використанням кодування Боуза-Чоудхурі-Хоквінгема (BCH). У такому протоколі можуть

використовуватися кодові слова, що створюються за допомогою біноміальних співвідношень або біноміальних індексів для передачі керуючої інформації.

Також методи кодування на основі біноміальних чисел можуть бути використані для забезпечення безпеки і захисту від перешкод в телекомунікаційних мережах. Наприклад, для забезпечення безпеки передачі інформації можна використовувати методи кодування з перевіркою парності, які базуються на біноміальних числах.

Методи кодування на основі біноміальних чисел можуть бути корисними для систем керування телекомунікаційними мережами, особливо в контексті передачі великих обсягів даних через мережу. Застосування таких методів дозволяє підвищити ефективність передачі даних та забезпечити більш надійний захист від пошкоджень даних.

Одним з прикладів використання методів кодування на основі біноміальних чисел у системах керування телекомунікаційними мережами є стандарт DOCSIS (Data Over Cable Service Interface Specification), який використовує кодування на основі біноміальних співвідношень для передачі даних через кабельні мережі.

DOCSIS (Data Over Cable Service Interface Specification) є стандартом, який визначає протоколи для передачі даних через кабельні мережі з доступом до Інтернету. Він був розроблений для забезпечення високошвидкісного Інтернет-з'єднання через кабельні мережі, що використовуються для телевізійного трансляційного мовлення.

DOCSIS використовує канали з різною швидкістю передачі даних для різних типів трафіку, таких як інтернет, телефонія та телебачення. Крім того, він забезпечує захист від шуму та перешкод, що можуть виникати в кабельній мережі.

DOCSIS має декілька версій, і кожна з них має свої характеристики та можливості. Наприклад, DOCSIS 3.1 підтримує передачу даних зі швидкістю до 10 Гбіт/с та підвищену ефективність передачі даних, зокрема за допомогою методу кодування на основі біноміальних співвідношень.

Загальною метою DOCSIS є забезпечення швидкого та стабільного Інтернет-з'єднання для користувачів, які підключаються до кабельних мереж. Він є одним з найпоширеніших стандартів для передачі даних через кабельні мережі та використовується багатьма провайдерами Інтернет-послуг по всьому світу.

Крім того, методи кодування на основі біноміальних чисел можуть бути використані для захисту від шуму в радіозв'язку та для забезпечення безпеки даних в безпроводних мережах.

Практичне застосування методів кодування на основі біноміальних чисел у телекомунікаційних системах є дуже широким і має багато переваг порівняно з іншими методами кодування. Застосування таких методів дозволяє забезпечити надійність передачі даних в умовах шуму та спотворень, що є особливо важливим в системах зв'язку.

Застосування методів кодування на основі біноміальних чисел у стандарті DOCSIS для передачі даних по кабельних мережах є добре відомим і широко використовується в інтернет-провайдерах. Застосування цих методів кодування дозволяє забезпечити надійну передачу даних в умовах шуму та інших перешкод, що забезпечує високу якість послуг.

Крім того, методи кодування на основі біноміальних чисел також знайшли своє застосування у системах зберігання даних, де вони дозволяють забезпечити надійність та цілісність збережених даних. Використання таких методів кодування дозволяє зменшити вартість зберігання даних за рахунок зменшення кількості потрібних дисків та збільшити швидкість доступу до даних.

Таким чином, можна зробити висновок, що методи кодування на основі біноміальних чисел є дуже важливими та ефективними в телекомунікаційних системах та системах зберігання даних. Застосування цих методів дозволяє забезпечити високу надійність та швидкість передачі даних в умовах різних видів шуму та інших перешкод. [5 , 7]

1.2 Розробка пристрою, що перетворює двійкові числа в біноміальні. Алгоритм роботи та структурна схема

Перетворювач двійкових чисел в біноміальні - це програмний або апаратний засіб, який перетворює двійкові числа в біноміальні коди. Цей процес називається кодуванням.

Кодування двійкових чисел у біноміальний код полягає у виборі певного числа бітів для представлення десяткового числа, яке відповідає символу. Це число бітів залежить від кількості символів, які необхідно закодувати та від довжини коду.

Один з варіантів перетворювача двійкових чисел в біноміальні коди - це таблиця кодів, де кожен символ має свій унікальний код. Наприклад, для використання 8-бітового біноміального коду ASCII, кожному символу відповідає послідовність з 8 бітів.

Існують також алгоритмічні методи перетворення двійкових чисел у біноміальні, такі як алгоритм Хаффмана, який використовується для зменшення довжини кодування символів за рахунок виділення меншої кількості бітів для найбільш часто використовуваних символів.

Перетворювач двійкових чисел в біноміальні коди є важливим засобом для ефективного зберігання та передачі даних в електронних системах.

У зв'язку із постійним збільшенням обсягів інформації, що передається, виникає питання підвищення надійності та завадостійкості цифрової апаратури, в тому числі, що використовується для передачі даних. Отже, пошук нових методів та алгоритмів побудови таких пристроїв є на сьогодні важливим завданням. Серед цих пристроїв особливе значення мають кодуючі та декодуючі пристрої, що використовуються для завадостійкого кодування даних, стиснення інформації, захисту її від несанкціонованого доступу. При цьому в ряді випадків для реалізації таких пристроїв використовуються нетрадиційні системи числення, такі як, наприклад, фібоначчівська або факторіальна система.

На сьогодні існують прості та надійні розробки цифрових пристроїв на основі біноміальних систем числення, які використовуються для стиснення та завадостійкого кодування інформації. У цих пристроях необхідно перетворювати біномні числа на їх номери і назад номери перетворювати на біномні числа. Вирішення цих завдань на сьогодні є актуальним питанням у разі ефективного застосування біноміальних кодів у мережах передачі.

Далі буде розглянуто метод перетворення біномних чисел у двійкові числа, що використовується для синтезу завадостійкого кодуючого пристрою, однак, до теперішнього часу відсутній метод зворотного перетворення двійкових чисел у біномні числа, що необхідно для синтезу пристрою, що декодує. Тому метою даного проекту є розробка методу синтезу перетворювача двійкових чисел у біномні.

Біноміальними системами числення називаються позиційні системи числення, в якості основи яких використовуються біномні коефіцієнти.

Кожна система числення характеризується числовою (кодоутворюючою) функцією та обмеженнями на значення цифр її чисел.

Будь-яка кінцева послідовність цифр $a_{j-1} \dots a_1 a_0$, що задовольняє обмеження на числову функцію біноміальної системи числення, називається біноміальним числом.

Числова (кодоутворююча) функція біноміальної системи числення:

$$A_i = a_{j-1} C_{n-1}^{k-q_j} + a_{j-2} C_{n-2}^{k-q_{j-1}} + \dots + a_l C_{n-j+l}^{k-q_{l+1}} + \dots + a_0 C_{n-j}^{k-q_1}$$

Двійкова біноміальна формула та всі біноміальні числа контролюються двома системами обмежень:

1. Контролює кількість $1(k)$. У кодовій комбінації може бути до k одиниць. Нулів може не бути, або бути менше ніж $n-k$.

2. Має наступні системи обмежень для утворення двійкових біноміальних чисел:

$$\begin{cases} k \leq j \leq n - 1 \\ q = k \\ a_0 = 1 \end{cases} \quad \begin{cases} n - k = j - q \\ 0 \leq q \leq k - 1 \\ a_0 = 0 \end{cases}$$

де n і k – цілочисленні параметри біноміальної системи числення;

j – кількість розрядів (довжина) біноміального числа, $j < n$;

a_j – біноміальна двійкова цифра – 0 або 1;

q – число одиниць в біноміальному числі;

q_l – сума одиничних значень a_j від $(j-1)$ -го розряду до l -го включно:

$$q_l = \sum_{\gamma=l}^j a_\gamma$$

де $a_j = q_j = 0$.

Як ваговий коефіцієнт l -го розряду в числовій функції (1) виступає біноміальний коефіцієнт $C_{n-j+l}^{k-q_{l+1}}$. Він залежить як від позиції розряду, так і від суми q_j попередніх цьому розряду двійкових значень цифр a_j . Остання

залежність характерна лише для структурних систем числення і надає їм стійких до перешкод і структуроутворюючих властивостей.

Кожна біномна кодова комбінація повинна задовольняти одній із систем обмежень.

Перебір двійкових біноміальних чисел здійснюється на основі наступного алгоритму:

1. Формується початкове двійкове слово, що складається з $n-k$ нулів.
2. В молодший розряд, що містить нуль, записується одиниця, і до неї записується нуль.
3. Пункт 2 повторюється до тих пір, поки число одиниць у кодовому слові не стане рівним k , а довжина – $(n-1)$ розрядів.
4. В молодший розряд, що містить нуль, записується одиниця, а в розряди справа від нього (якщо число одиниць не рівне k) послідовно записуються нулі до тих пір, поки їх загальне число не стане рівним $(n-k)$.
5. Повернення до пункту 2. Цикл повторюється до тих пір, поки старші розряди кодового слова не заповняться одиницями. [16]

Біноміальні числа – це двійкові числа довжиною n , кількість одиниць в яких менша або дорівнює деякій величині k , а число нулів до першої 1 ліворуч має бути не більше $n - k - 1$. Наприклад, при $n = 6$, $k = 4$ комбінації 01000, 10000, 01110 є біноміальними числами, а комбінації 00001, 11111, 00100 – не біноміальними, які сприймаються як помилкові. [12]

У табл. 1 представлені всі двійкові біномні числа із зазначеними в прикладі параметрами k і n . Для вирішення поставленої в цій роботі завдання необхідно розробити алгоритм, що однозначно встановлює відповідність між номером, що представляє двійкове число, та біноміальним числом.

Таблиця 1 – Двійкові та біноміальні числа з параметрами $n = 6$ та $k = 4$

| № | Двійкове число | Біноміальне число |
|---|----------------|-------------------|
| 0 | 0000 | 00000 |
| 1 | 0001 | 01000 |
| 2 | 0010 | 01100 |
| 3 | 0011 | 01110 |

| | | |
|----|------|-------|
| 4 | 0100 | 01111 |
| 5 | 0101 | 10000 |
| 6 | 0110 | 10100 |
| 7 | 0111 | 10110 |
| 8 | 1000 | 10111 |
| 9 | 1001 | 11000 |
| 10 | 1010 | 11010 |
| 11 | 1011 | 11011 |
| 12 | 1100 | 11100 |
| 13 | 1101 | 11101 |
| 14 | 1110 | 11110 |

Маючи дані n та k , можна розрахувати діапазон біноміальної системи числення з двійковим алфавітом:

$$P = C_n^k = C_n^{n-k} = \frac{n!}{k!(n-k)!}, \text{ де}$$

P -діапазон біноміальних чисел, тобто кількість дозволених кодових комбінацій.

На Рисунок 1.3 та Рисунок 1.4 наведена залежність параметрів від діапазону біноміальних чисел з двійковим алфавітом.

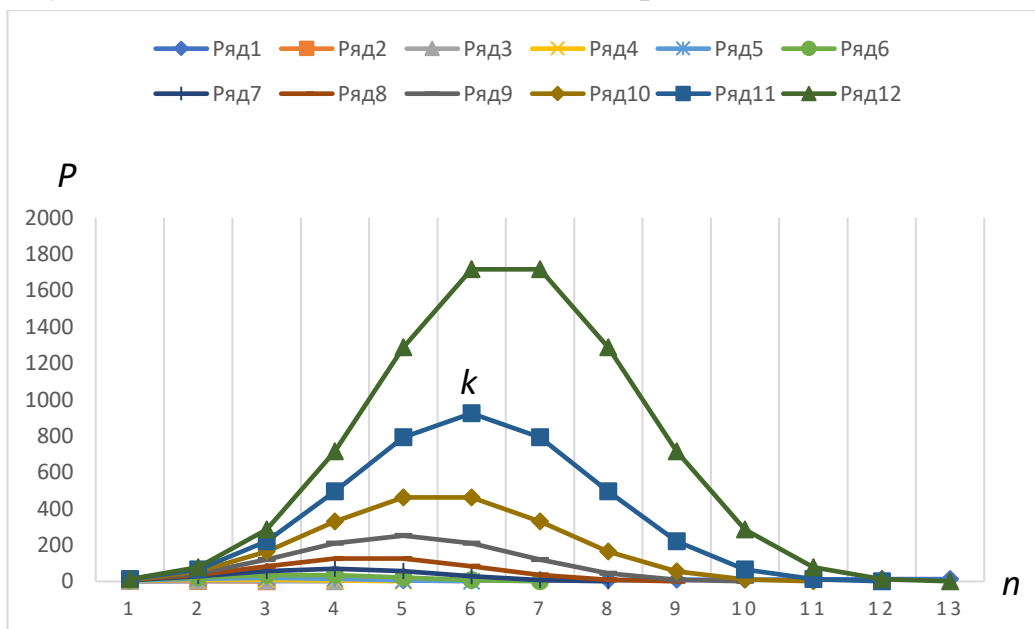


Рисунок 1.3 - Графік залежності діапазону біноміальних чисел з двійковим алфавітом від параметрів n та k

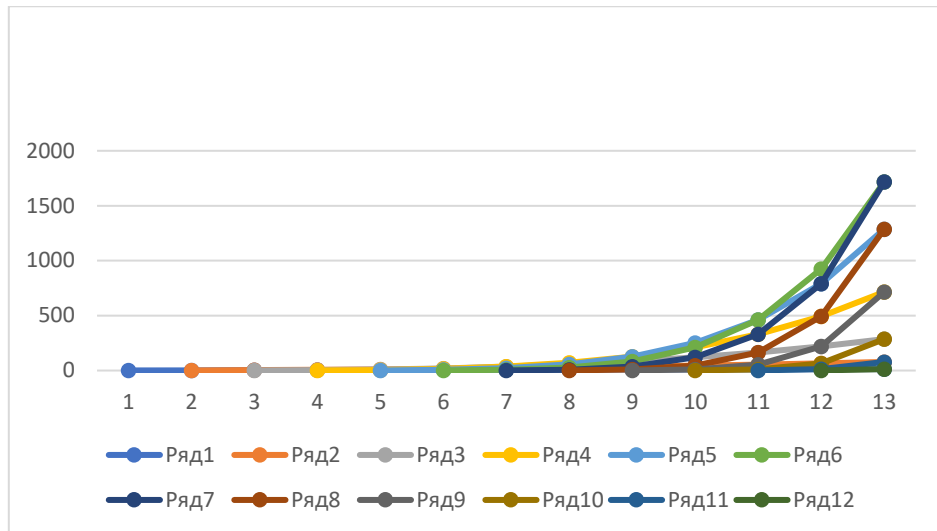


Рисунок 1.4 - Графік залежності діапазону біноміальних чисел з двійковим алфавітом від контрольного числа

Також за допомогою відомих параметрів n та k можна розрахувати частку виявлених помилок D .

$$D = 1 - \frac{N_p}{N_b} = 1 - \frac{C_n^k}{2^n}$$

Із формули видно, що чим більша різниця між дозволеними кодовими комбінаціями та можливими кодовими комбінаціями, тим вище шанс виявлення помилки в коді.

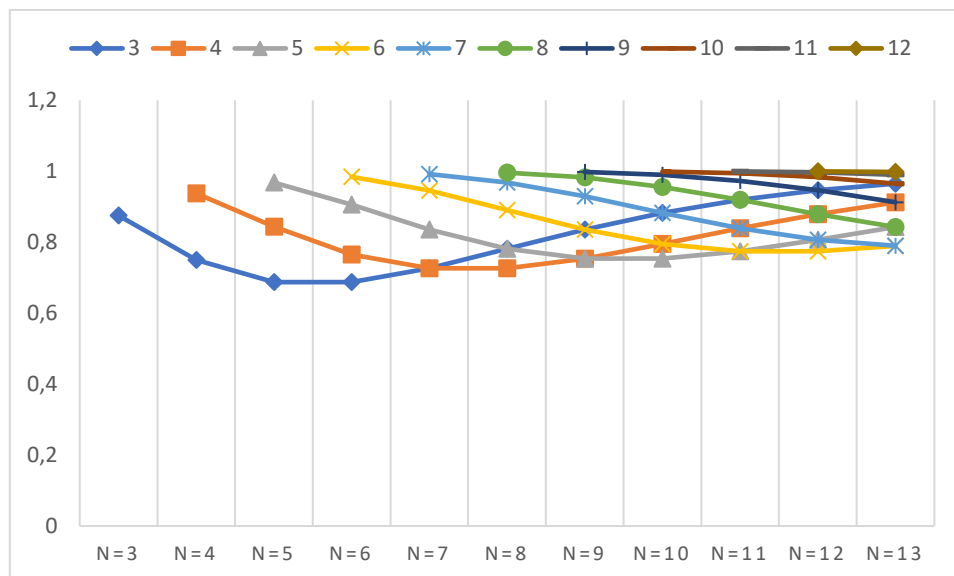


Рисунок 1.5. Графіки контрольних значень

Ідея перетворювача полягає в наступному: двійкове число яке необхідно перетворити, записують в регістр. Застосовують 2 лічильники додавання. Один працює в біноміальних кодах, другий звичайний двійковий лічильник. На обидва одночасно подають імпульси перетворення. На кожному кроці відбувається порівняння вмісту лічильника додавання і змістом регістру двійкового числа. У разі співпадіння вмісту лічильника додавання і регістру процес перетворення закінчується, на виході біноміального лічильника маємо біноміальний код, що є еквівалентом поданої двійкової комбінації. Схематично це зображено на Рисунку 1.6.

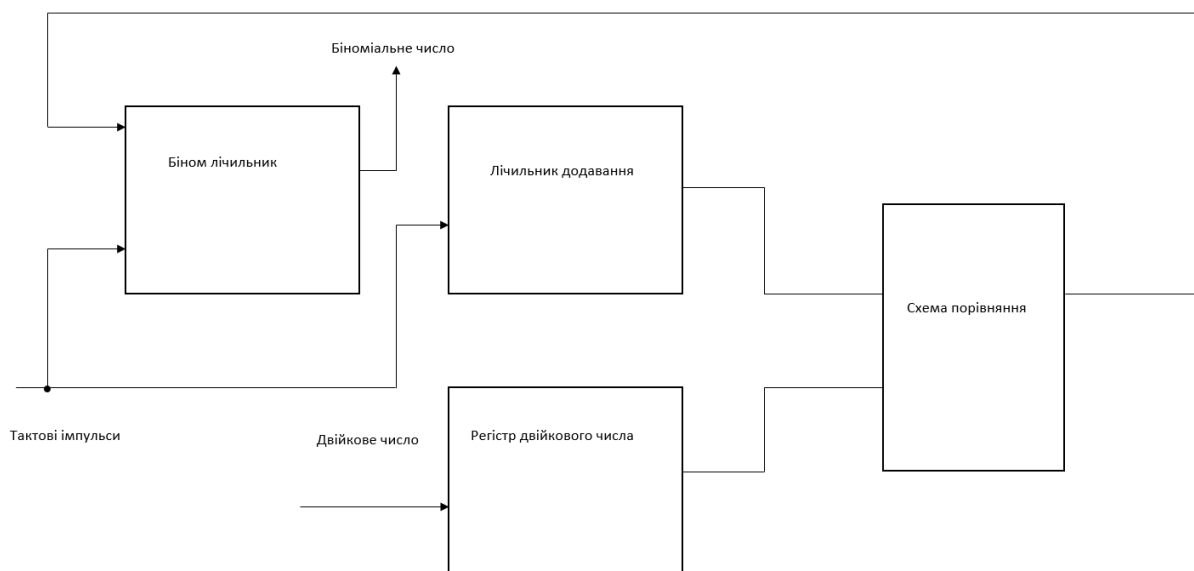


Рисунок 1.6 – Ідея перетворювача

На Рисунку 1.7 представлена структурна схема перетворювача двійкового коду біноміальний. Схема містить регістр зберігання, блок порівняння, схему контролю, схему заборони, пристрій управління і два підсумовуючі лічильника – двійковий та біноміальний. При вступі на вхід перетворювача двійкової кодової комбінації, вона за сигналом «Запис», що надходить з виходу блоку управління, зберігається в регістрі зберігання. З виходів регістра зберігання двійковий код надходить на входи блоку порівняння. У той самий час на інші входи блоку порівняння надходить двійковий код з виходів двійкового підсумовуючого лічильника. Підсумовуючий двійковий лічильник і підсумовуючий біноміальний

лічильник працюють синхронно. Перемикання лічильників відбувається за тактуючими імпульсами, що надходять з виходу блоку керування. Коли двійкові коди в блоці порівняння будуть рівні, то в підсумовуючому біноміальному лічильнику, в цей час, буде перебувати біноміальне число, яке відповідає двійковому числу. Тому, за одиничним сигналом з виходу блоку порівняння, пристрій керування припиняє тактування лічильників, а з виходу надходить сигнал на схему заборони, що дозволяє передачу біномного числа на виходи перетворювач кодів. У цьому цикл перетворення закінчено. З виходу пристрою управління надходить сигнал встановлює лічильники у вихідне положення, а з виходу надходить сигнал «Запис» яким у реєстр зберігання занесеться наступний двійковий код.

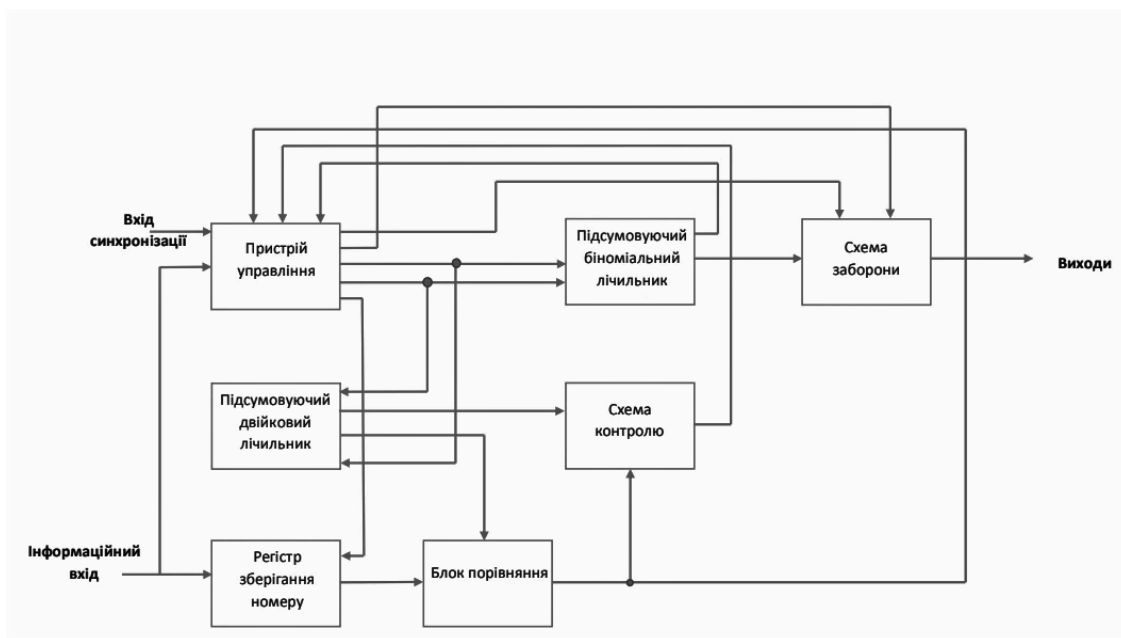


Рисунок. 1.7 – Структурна схема перетворювача коду

У випадку, коли двійковий підсумовуючий лічильник перебере всі кодові комбінації, а сигналу з виходу блоку порівняння не надійде, з виходу схеми контролю на вхід пристрою управління надходить сигнал, що повідомляє про помилку. При надходженні сигналу помилки зі схеми контролю або сигналу про збій у роботі підсумовуючого біноміального лічильника, лічильники скидаються в вихідний стан, а з виходу пристрою керування на виходи перетворювача надходить сигнал про помилку.

Алгоритм роботи пристрою ґрунтується на ідеї одночасного використання операцій двійкового та біноміального рахунки у відповідних лічильниках. У обнуленому на початку роботи алгоритму біноміальному лічильнику рахунок відбувається до того часу, поки вміст двійкового лічильника не зрівняється з перекладеним двійковим числом. Після цього рахунок припиняється, та інформація знімається з біномного лічильника. Безпосередньо алгоритм перетворення полягає в наступному:

1. Здійснюється запис двійкового числа, що надходить на вхід перетворювача, у реєстр зберігання.

2. Підсумовуючі двійковий і біноміальний лічильники обнуляються.

3. На входи лічильників подаються імпульси перетворення від схеми керування.

4. На кожному кроці проводиться порівняння двійкового числа, що записане в реєстрі двійкового числа з числом, що надходить з виходів двійкового сумарного лічильника.

5. У разі рівності двійкового числа в лічильнику та реєстрі підсумовування чисел у лічильниках припиняється.

6. Таким чином на виході біноміального лічильника маємо біноміальне число, що є еквівалентом заданого двійкового числа.

Графічна схема цього алгоритму наведена на Рисунку 1.8.

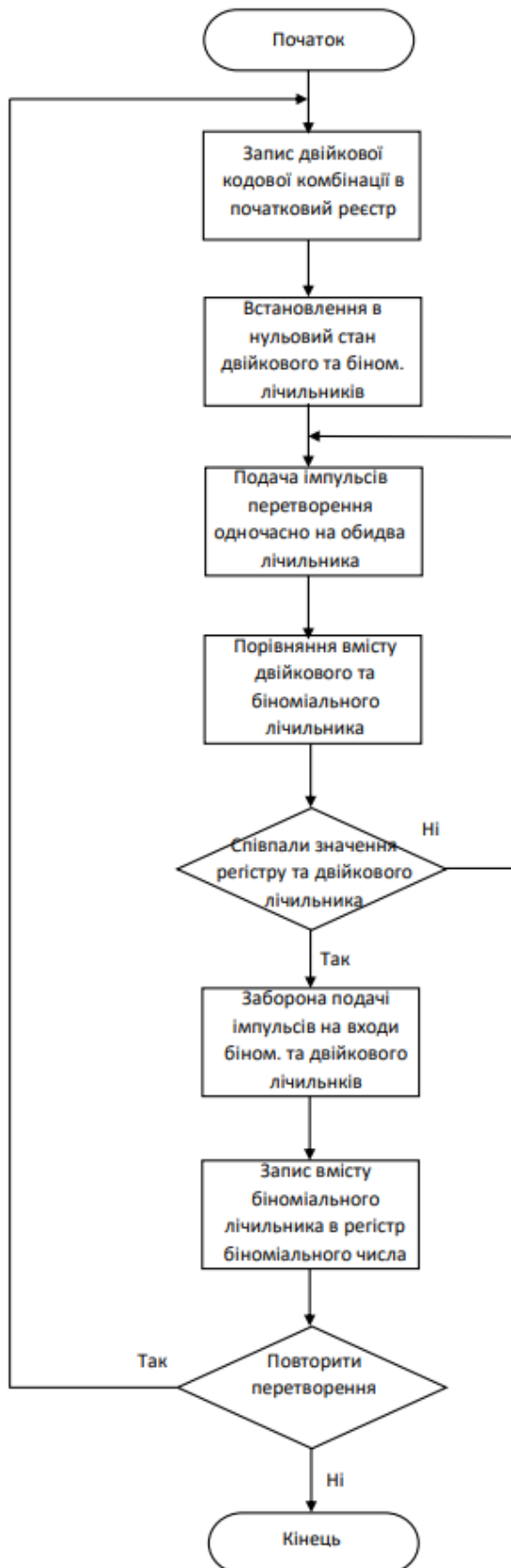


Рисунок 1.8 – Графічна схема алгоритму функціонування перетворювача коду

Докладніше роботу пристрою керування перетворювача кодів поданого на Рисунок 1.9. За сигналом "Пуск" 36 на виході тригера 27 з'явиться одиничний сигнал «Запис» за яким до регістру зберігання заноситься двійкова кодова комбінація. У разі надходження кодової комбінації на елемент АБО23, тригер 24 переходить в одиничний стан і через елемент І25 тактуючі імпульси надходять на лічильники, в той же час тригер 27 скидається. При надходженні сигналу 17 рівності з блоку порівняння, видається сигнал про видачу інформації на схему заборони 7, якщо відсутні помилки в роботі пристрою. Якщо ж на виході 18 схеми контролю є сигнал помилки або відбудеться збій біномного лічильника, видається сигнал про помилка перетворення 43.

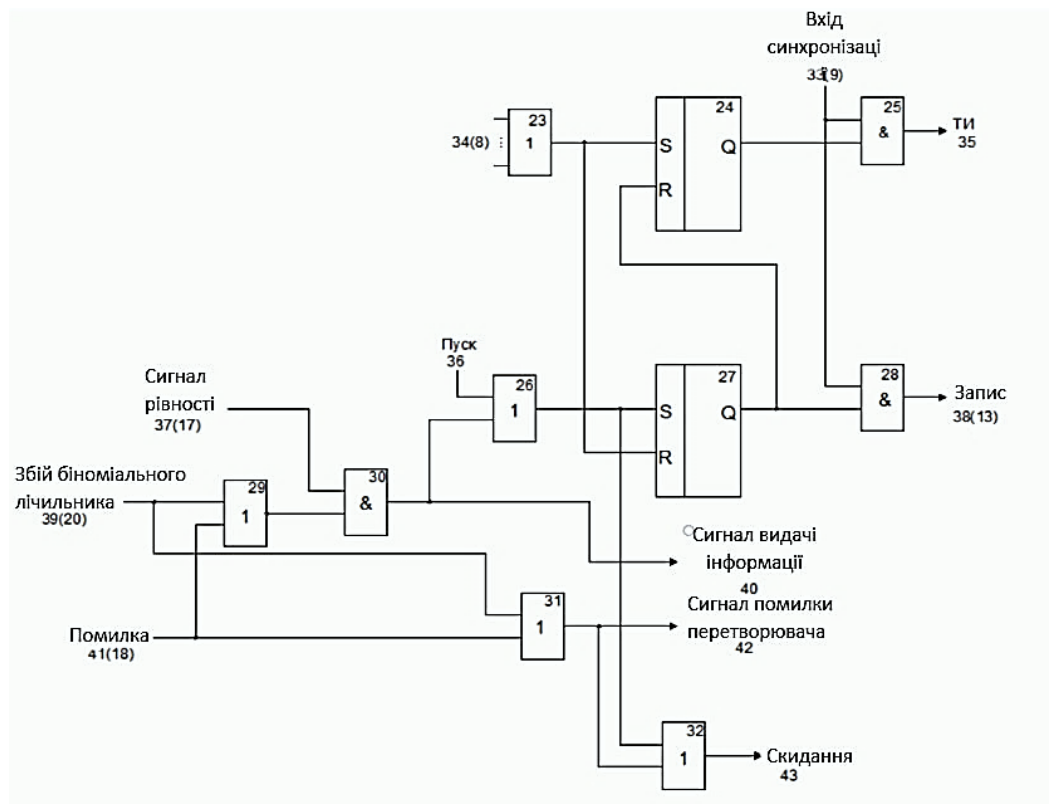


Рисунок. 1.8 - Схема пристрою управління

1.3 Розробка функціональної схеми

DI – джерело інформації, звідки надходить двійкові кодові комбінації (виходи з датчиків, виходи з аналого-цифрового перетворювача (при реалізації даного пристрою, це чотирьохрозрядні кодові комбінації))

Далі передається на регістр зберігання. СТ 1 – двійковий лічильник, СТ 2- біноміальний лічильник. Перед перетворенням вони встановлюються в нульовий стан. (СТQ) – тригер керування.

За сигналом ПУСК тригер керування встановлюється в одиничний стан та дозволяє через елемент І надходження імпульсів від генератора одночасно на обидва лічильники.

За тим же імпульсом ПУСК в регістр записується двійкове число, яке необхідно перетворити. За кожним кроком перетворення виконується порівняння вмісту регістру двійкового числа та двійкового лічильника. Цю операцію виконує компаратор. При співпадінні значень процес перетворення зупиняється. З виходу компаратора (=) сигнал подається на вхід скидання тригера керування та на вхід запису регістру біноміального числа і для подальшого зберігання в нього переписується вміст біноміального лічильника.

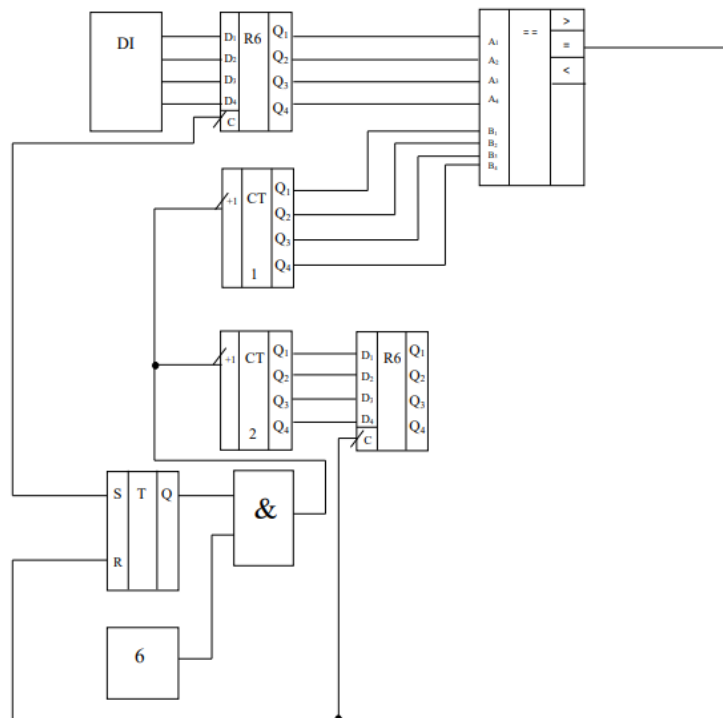


Рисунок. 1.9 - Функціональна схема

За допомогою перетворювача двійкових кодових комбінацій в біноміальні числа підвищується достовірність перетвореної інформації. Алгоритм являється настільки простим, що дає можливість реалізації пристрою в апаратному вигляді, чим підвищує надійність його роботи. [12, 14]

2 РОЗРОБКА ТА РОЗРАХУНОК ПРИНЦИПОВОЇ СХЕМИ ПРИСТРОЮ

2.1 Вибір та обґрунтування елементної бази пристрою

Метою вибору елементної бази є обґрунтування серії (або серій) інтегральних мікросхем, а також інших електрорадіоелементів, необхідних для раціональної реалізації пристрою, що проектується.

Критеріями вибору серії (серій) ІМС є:

- наявність необхідних функціональних вузлів у складі серії ІМС;
- мала споживана потужність;
- виконання вимог по швидкодії (граничній робочій частоті) і умовам експлуатації;
- низька вартість;
- можливість керувати необхідними елементами, наприклад, індикаторами без додаткових підсилень і перетворень вихідних сигналів і т.п.

Вибір елементної бази необхідно проводити в наступній послідовності:

- за функціональною схемою пристрою визначаються необхідні функціональні вузли (лічильники, реєстри, шифратори, перетворювачі коду тощо) та їх параметри;
- по довідниках визначаються серії ІМС, що містять всі або частину відповідних функціональних вузлів. При відсутності функціональних вузлів визначається можливість їх побудови за допомогою вхідних до складу серії елементів;
- на основі проведеного аналізу визначається одна або декілька серій, що застосовуються для побудови пристрою.

При виборі дискретних елементів (індикаторів, електромагнітних реле і т.д.), які входять до складу пристрою, що проектується, доцільно використовувати ті, які керуються сигналами з мікросхем або спеціальними мікросхемами сполучення, що входять до складу серій. Інакше проводиться розрахунок схем сполучення на дискретних елементах.

Вибір елементної бази доцільно ілюструвати таблицями, наприклад:

- таблиця відповідності складу серій потрібним функціональним вузлам і можливість реалізації функціональних елементів на дискретних логічних елементах серії;
- таблиця характеристик обраних серій ІМС;

- таблиця характеристик необхідних дискретних елементів.

На підставі аналізу даних таблиць проводиться вибір елементної бази.

[17]

Мікросхеми TTL серії К 1533 відносяться до логічних інтегральних схем (ЛІС) типу TTL (Transistor-Transistor Logic) і мають важливе застосування в електроніці.

Мікросхеми серії К 1533 є 4-бітними багатозначними мультиплексорами з відкритим колектором. Вони мають 16 входів даних (D0-D15), 4 входи управління (A, B, C, D) і 4 вихідні лінії (Y0-Y3). Кожен вхід даних може бути підключений до відповідного вихідного лінії за допомогою відповідної комбінації сигналів на входах управління.

Мікросхеми серії К 1533 використовуються для реалізації функцій мультиплексування, деяких комбінаційних логічних функцій та управління даними. Вони можуть бути використані в цифрових системах збору даних, керування, комутації та інших додатках, де потрібна обробка даних з використанням мультиплексорів.

Переваги мікросхем TTL серії К 1533:

1. Швидкість: Мікросхеми TTL взагалі відомі своєю високою швидкістю роботи. Вони можуть оперувати на високих частотах, що робить їх популярними у застосуваннях, де важлива швидкість обробки сигналів.

2. Надійність: TTL має високу надійність і добре переносить зовнішні впливи, такі як шум або електромагнітні перешкоди. Вони можуть працювати в широкому діапазоні температур і умов середовища.

3. Сумісність: Мікросхеми серії К 1533 використовують стандартні TTL-логічні рівні, що робить їх сумісними з іншими TTL-схемами. Це дозволяє їх легко інтегрувати в існуючі цифрові системи.

4. Простота використання: Мікросхеми TTL є досить простими у використанні. Вони мають прямий доступ до входів та вихідних ліній, що спрощує їх підключення та налаштування.

Недоліки мікросхеми TTL серії К 1533:

1. Енергоспоживання: TTL має відносно високе споживання енергії порівняно з іншими логічними сімействами, такими як CMOS (Complementary Metal-Oxide-Semiconductor). Це може бути проблемою в мобільних або батарейно-живлених пристроях.

2. Тепловиділення: TTL може виділяти багато тепла під час роботи, особливо при високих швидкостях і великих масштабах інтеграції. Це може вимагати додаткового охолодження або розсіювачів тепла.

3. Обмеження напруги: Мікросхеми TTL потребують стабільного живлення в межах певного діапазону напруги. Неправильна напруга живлення може призвести до неправильної роботи або пошкодження мікросхеми.

4. Обмежена кількість входів/виходів: Мікросхеми серії K 1533 мають обмежену кількість входів і виходів, що може бути недостатньо для деяких складних додатків. Для більш складних функцій можуть знадобитися додаткові мікросхеми або використання інших сімейств логічних схем. [18].

2.2 Розробка біноміального лічильника

В пристрої, що синтезується, застосовується паралельний лічильник, який працює в біноміальних кодах.

Переваги паралельних лічильників (також відомих як синхронні лічильники) включають:

1. Швидкість операцій: Паралельні лічильники можуть працювати швидше, оскільки кожен біт лічильника обробляється одночасно. Усі біти лічильника змінюють свої стани одночасно, що дозволяє отримати високу швидкодію роботи.

2. Простота проектування: Паралельні лічильники мають просту структуру, оскільки кожен біт лічильника має свій власний вхід і вихід. Це спрощує їх проектування та розуміння принципу роботи.

3. Простота зчитування стану: У паралельних лічильниках стан кожного біту може бути прочитаний окремо, без необхідності послідовного зчитування. Це дозволяє швидше і простіше отримувати значення лічильника для подальшої обробки.

4. Гнучкість: Паралельні лічильники можуть бути використані для реалізації різних послідовностей лічильників, включаючи двійкові, десяткові та інші системи лічення. Вони можуть бути конфігуровані для відповідності конкретним вимогам проекту.

5. Легкість синхронізації: У паралельних лічильниках всі біти синхронізуються за допомогою спільного тактового сигналу. Це спрощує процес синхронізації та забезпечує правильну роботу лічильника.

Виконаємо синтез паралельного біноміального лічильника.

Таблиця 2.1. - Таблиця функціонування біноміального лічильника

| № | Q_1 | Q_2 | Q_3 | Q_4 | Q_1 | Q_2 | Q_3 | Q_4 | D_1 | D_2 | D_3 | D_4 |
|----|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | * | * | * | * | * | * | * | * |
| 2 | 0 | 0 | 1 | 0 | * | * | * | * | * | * | * | * |
| 3 | 0 | 0 | 1 | 1 | * | * | * | * | * | * | * | * |
| 4 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 5 | 0 | 1 | 0 | 1 | * | * | * | * | * | * | * | * |
| 6 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 9 | 1 | 0 | 0 | 1 | * | * | * | * | * | * | * | * |
| 10 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| 11 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 12 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |
| 13 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| 14 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 15 | 1 | 1 | 1 | 1 | * | * | * | * | * | * | * | * |

В таблиці застосовуємо рівномірні дозволені біноміальні кодові комбінації для параметрів $n = 5$, $k = 3$. В якості тригерів виберемо D-тригери.

D-тригери (D-flip-flops) є одним з найпоширеніших типів тригерів у логічних інтегральних схемах. Вони використовуються для збереження і синхронізації даних у цифрових системах. Кожен D-тригер має один вхід даних (D), один вхід такту (clock) і два виходи: вихід даних (Q) і зворотний вихід (\bar{Q}).

Робота D-тригера базується на принципі збереження значення вхідного сигналу D на фронті (спаді) тактового сигналу. Коли тактовий сигнал знаходиться в стані спаду (або фронту), значення на вході D копіюється і зберігається на виході Q. Зворотний вихід \bar{Q} має протилежне значення від Q.

Основні переваги використання D-тригерів:

1. Збереження даних: D-тригери дозволяють зберегти та синхронізувати дані у цифрових системах. Це дозволяє контролювати часові послідовності та правильність передачі даних.

2. Синхронізація: D-тригери можуть використовуватися для синхронізації операцій та управління в цифрових системах, де важливо забезпечити правильну послідовність виконання команд і дій.

3. Реалізація регістрів і лічильників: D-тригери є основними будівельними блоками для створення регістрів і лічильників. Вони можуть використовуватися для зберігання та переміщення даних у великих масивах, що дозволяє реалізувати складні функціональні блоки.

4. Простота використання: D-тригери є простими у використанні та розумінні. Вони мають стандартний інтерфейс з одним вхідним сигналом даних і тактовим сигналом, що дозволяє їх легко використовувати в різних логічних схемах.

5. Гнучкість: D-тригери можуть бути поєднані та конфігуровані для реалізації різних функцій та логічних операцій. Вони можуть використовуватися для створення різних типів лічильників, регістрів та послідовних схем.

Отримані функції (в таблиці 2.1) є неповністю визначеними логічними функціями. Виконаємо мінімізацію кожної з чотирьох неповністю визначених функцій.

Так як аналізовані функції D_1 - D_4 є неповністю визначеними логічними функціями, то мінімізувати їх найзручніше допомогою методу діаграм Вейча. Нанесемо функції і зробимо мінімізацію. Отримаємо мінімальні ДНФ (диз'юнктивні нормальні форми).

Складемо для функції, що мінімізуються діаграми Вейча. Так як мінімізації підлягають чотири функції, то і діаграм Вейча також буде чотири.

| | | | | | |
|------------------|--|------------------|------------------|------------------|----|
| | | Q_2 | $\overline{Q_2}$ | | |
| | | 1 | * | 1 | |
| | | 12 | 13 | 9 | 8 |
| Q_1 | | 0 | * | 1 | 1 |
| | | 14 | 15 | 11 | 10 |
| | | 0 | 1 | * | * |
| | | 6 | 7 | 3 | 2 |
| $\overline{Q_1}$ | | 0 | * | * | 0 |
| | | 4 | 5 | 1 | 0 |
| | | $\overline{Q_4}$ | Q_4 | $\overline{Q_4}$ | |

| | | | | | |
|------------------|--|------------------|------------------|------------------|----|
| | | Q_2 | $\overline{Q_2}$ | | |
| | | 1 | * | 1 | |
| | | 12 | 13 | 9 | 8 |
| Q_1 | | 0 | * | 1 | 1 |
| | | 14 | 15 | 11 | 10 |
| | | 0 | 1 | * | * |
| | | 6 | 7 | 3 | 2 |
| $\overline{Q_1}$ | | 0 | * | * | 0 |
| | | 4 | 5 | 1 | 0 |
| | | $\overline{Q_4}$ | Q_4 | $\overline{Q_4}$ | |

Запишемо мінімальну ДНФ:

$$D_1 = Q_1 \overline{Q_3} \vee Q_3 Q_4 \vee \overline{Q_2} Q_3$$

Перетворимо вираз для реалізації в базисі Шеффера. Застосуємо закон подвійного заперечення, а потім закон Де Моргана:

$$D_1 = \overline{\overline{Q_1 \overline{Q_3} \vee Q_3 Q_4 \vee \overline{Q_2} Q_3}} = \overline{\overline{Q_1 \overline{Q_3}} \times \overline{Q_3 Q_4} \times \overline{\overline{Q_2} Q_3}}$$

Побудуємо схему для функції збудження D_1 першого тригера:

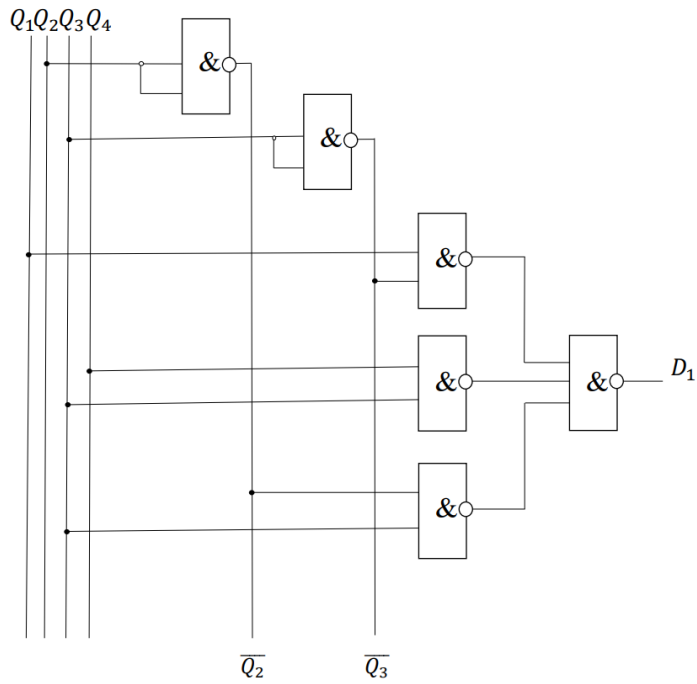


Рисунок 2.1 - Реалізація функції D1 в базисі Шеффера

Далі повторюємо ту ж саму процедуру ще 3 рази, для кожної D:

| | | | | | |
|-------|------------------|------------------|------------------|------------------|------------------|
| | | Q_2 | $\overline{Q_2}$ | | |
| | | 1 | 1 | * | 1 |
| | | 12 | 13 | 9 | 8 |
| Q_1 | | 0 | * | 1 | 1 |
| | | 14 | 15 | 11 | 10 |
| | | 0 | 1 | * | * |
| | $\overline{Q_1}$ | 6 | 7 | 3 | 2 |
| | | 0 | * | * | 0 |
| | | 4 | 5 | 1 | 0 |
| | | $\overline{Q_4}$ | Q_4 | $\overline{Q_4}$ | |
| | | | | | $\overline{Q_3}$ |
| | | | | | Q_3 |
| | | | | | $\overline{Q_3}$ |

Мінімальна ДНФ:

$$D_2 = Q_1 Q_4 \vee Q_2 \overline{Q_3} \vee \overline{Q_1} \overline{Q_4} = \overline{\overline{Q_1 Q_4} \vee \overline{Q_2 \overline{Q_3}} \vee \overline{\overline{Q_1} \overline{Q_4}}} = \overline{\overline{Q_1 Q_4} \times \overline{Q_2 \overline{Q_3}} \times \overline{\overline{Q_1} \overline{Q_4}}}$$

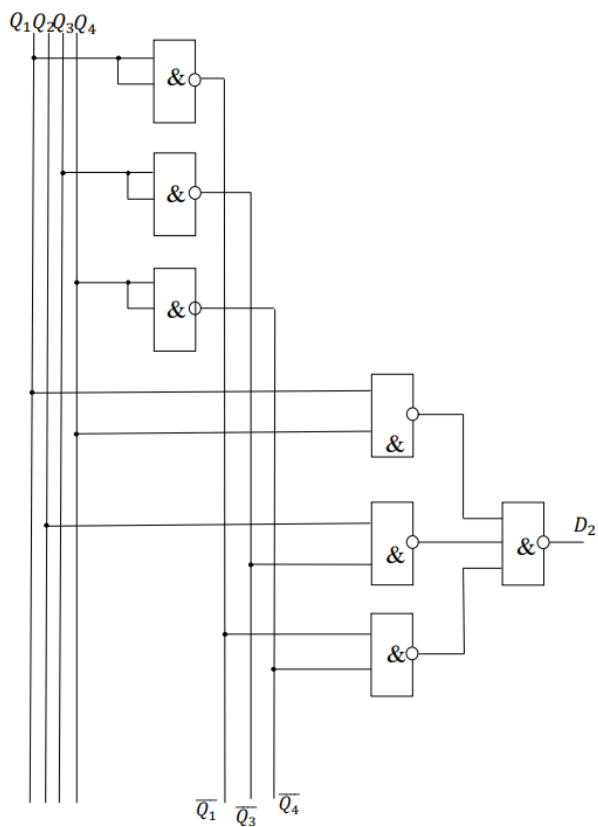


Рисунок 2.2 - Реалізація функції D2 в базисі Шеффера

| | | | | |
|------------------|----|------------------|------------------|------------------|
| | | Q_2 | $\overline{Q_2}$ | |
| | | 0 | 1 | * |
| Q_1 | 12 | 13 | 9 | 8 |
| | | * | 0 | 1 |
| | 14 | 15 | 11 | 10 |
| | | 1 | 0 | * |
| | | * | * | * |
| $\overline{Q_1}$ | 6 | 7 | 3 | 2 |
| | | 1 | * | 0 |
| | 4 | 5 | 1 | 0 |
| | | $\overline{Q_4}$ | Q_4 | $\overline{Q_4}$ |
| | | $\overline{Q_3}$ | Q_3 | $\overline{Q_3}$ |

Мінімальна ДНФ:

$$\begin{aligned}
 D_3 &= Q_1 \overline{Q_3} Q_4 \vee Q_1 \overline{Q_2} \overline{Q_4} \vee \overline{Q_1} Q_2 \overline{Q_4} = \overline{Q_1 \overline{Q_3} Q_4 \vee Q_1 \overline{Q_2} \overline{Q_4} \vee \overline{Q_1} Q_2 \overline{Q_4}} \\
 &= \overline{Q_1 \overline{Q_3} Q_4} \times \overline{Q_1 \overline{Q_2} \overline{Q_4}} \times \overline{\overline{Q_1} Q_2 \overline{Q_4}};
 \end{aligned}$$

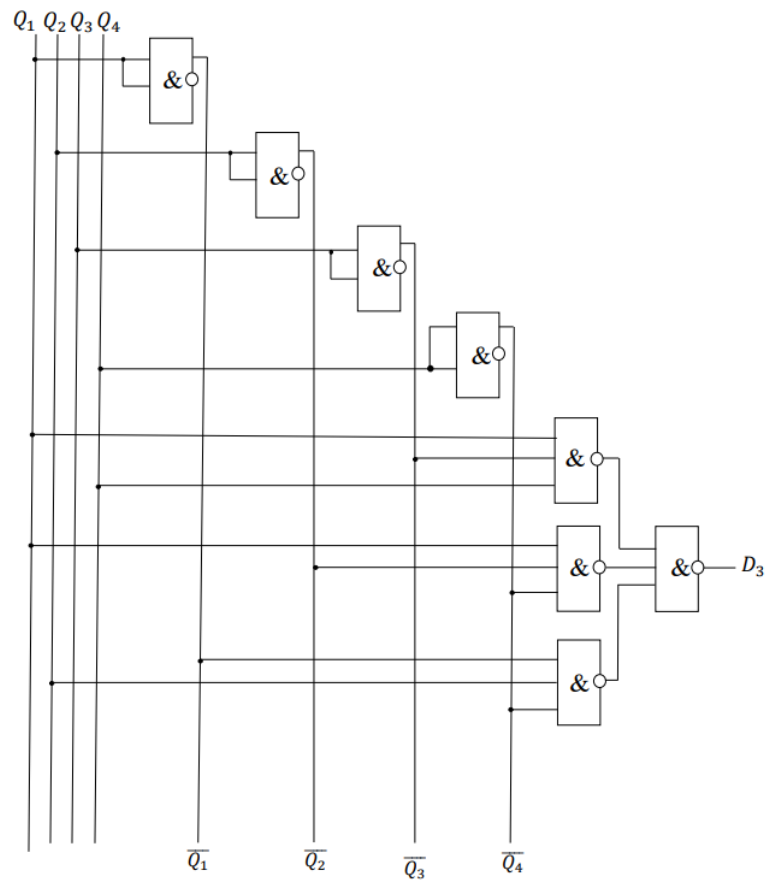


Рисунок 2.3 - Реалізація функції D3 в базисі Шеффера

| | | | | | | |
|------------------|----|------------------|------------------|------------------|------------------|------------------|
| | | Q_2 | $\overline{Q_2}$ | | | |
| Q_1 | 12 | 1 | 0 | * | 0 | $\overline{Q_3}$ |
| | 13 | 0 | * | 0 | 1 | |
| $\overline{Q_1}$ | 14 | 1 | 0 | * | * | Q_3 |
| | 15 | 0 | * | * | * | |
| Q_1 | 6 | 0 | * | * | 0 | $\overline{Q_3}$ |
| | 7 | 0 | * | * | 0 | |
| | 4 | $\overline{Q_4}$ | Q_4 | $\overline{Q_4}$ | Q_4 | |
| | 5 | Q_4 | $\overline{Q_4}$ | $\overline{Q_4}$ | Q_4 | |
| | 1 | $\overline{Q_4}$ | Q_4 | $\overline{Q_4}$ | Q_4 | |
| | 0 | Q_4 | $\overline{Q_4}$ | Q_4 | $\overline{Q_4}$ | |

Мінімальна ДНФ:

$$\begin{aligned}
 D_4 &= Q_1 Q_2 \overline{Q_3} \overline{Q_4} \vee \overline{Q_2} Q_3 \overline{Q_4} \vee \overline{Q_1} Q_3 \overline{Q_4} = \overline{\overline{Q_1 Q_2 Q_3 Q_4}} \vee \overline{\overline{Q_2 Q_3 Q_4}} \vee \overline{\overline{Q_1 Q_3 Q_4}} = \\
 &= \overline{Q_1 Q_2 Q_3 Q_4} \times \overline{Q_2 Q_3 Q_4} \times \overline{Q_1 Q_3 Q_4}
 \end{aligned}$$

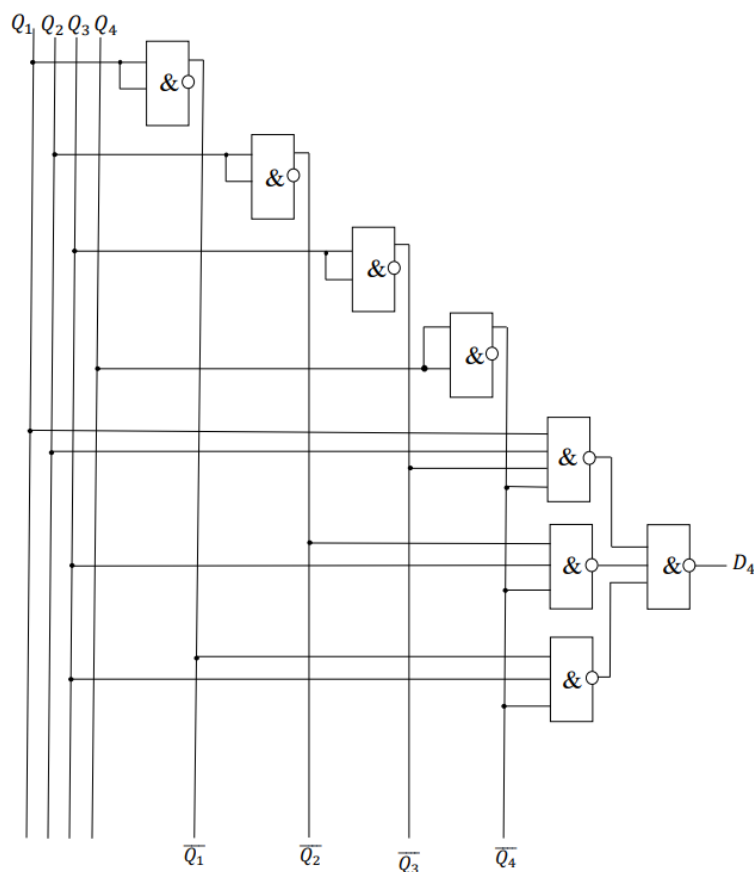


Рисунок 2.4 - Реалізація функції D4 в базисі Шеффера

2.3 Розробка лічильників з заданим коефіцієнтом перерахунку

В процесі функціонування довільних пристроїв, двійкові лічильники, що застосовуються, частіше працюють з числами, що мають відмінність від цілого ступеня 2. Пристрій, який підраховує кількість вхідних імпульсів і записує це число в код, називається лічильником.

Лічильники використовуються в різноманітному цифровому обладнанні, особливо в обладнанні з комп'ютерним керуванням. Вони складаються з взаємопов'язаних бітових схем і містять тригери та логічні елементи.

Унікальним для лічильників є мікрооперація збільшення (або зменшення) або зміни їхнього вмісту на одну одиницю. Крім того, в них виконуються такі мікрооперації, як ініціалізація, збереження та виведення слів.

Лічильник спочатку характеризується модулем (ємністю) рахунку. У міру надходження вхідних сигналів він змінює стан у стан, повертаючись до

початку циклу кожні K сигналів. Лічильний модуль визначає максимальну кількість імпульсів, які може підрахувати лічильник. Наприклад, якщо $K_{сч}=8$, лічильник матиме 8 стійких станів і кожні 8 імпульсів, що надходять на вхід, повертатимуть лічильник у вихідний стан.

Основним режимом роботи лічильника є реєстрація та розподіл кількості сигналів, що надійшли на лічильник.

У першому режимі лічильник записує свій вміст (цифровий код), а в другому режимі вихідним сигналом є імпульс переповнення лічильника.

В двійкових лічильниках коефіцієнт перерахунку пов'язаний певною залежністю з числом розрядів (тригерів) лічильника $K_{сч}=2^N$ і може дорівнювати 2, 4, 8, 16, 32, 64 і т.д. Однак на практиці нерідко виникає необхідність у лічильниках, коефіцієнт перерахунку яких відмінний від 2^N . Зокрема потрібні лічильники з коефіцієнтом рахунку $K_{сч}=3, 10$ і т.д., тобто лічильники, що мають відповідно 3, 10 і т.д. стійких станів.

Принцип побудови таких лічильників полягає у виключенні зайвих стійких станів у лічильника з $K_{сч}=2N$, тобто в організації схем, що забороняють деякі стани.

Залежно від того, які стани лічильника вибираються в якості основних, всі лічильники з довільним коефіцієнтом можна розділити на лічильники з природним і довільним порядком рахунку.

Особливістю лічильників з природним порядком рахунку є те, що порядок їх рахунку відповідає порядку рахунку звичайних лічильників, що додають або віднімають. Відмінність полягає в тому, що шляхом введення додаткових зв'язків рахунок закінчується раніше значення 2^N . При побудові лічильника даним способом вибирається двійковий лічильник розрядності N , такий, щоб 2^N було більше K , доповнюється елементом I , який по станам виходів Q_i виявляє код кінця рахунку, після чого по ланцюгу R скидає лічильник в нуль. Структурна схема, що реалізує цей спосіб, показана на рисунку 2.5.

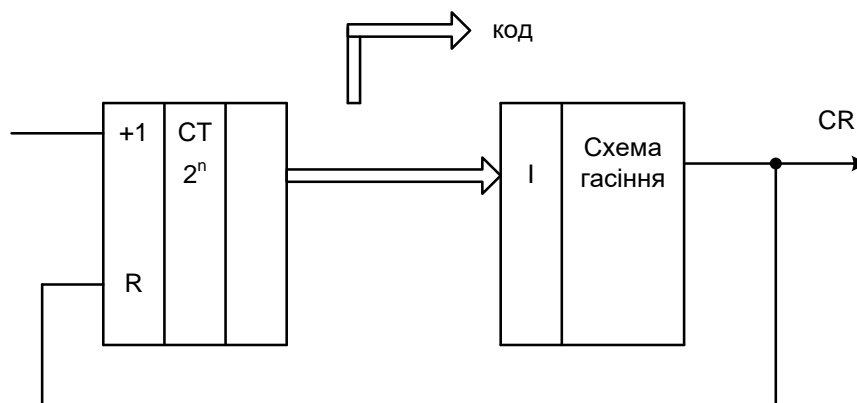


Рисунок 2.5 – Структурна схема лічильника з заданим коефіцієнтом та природним порядком рахунку

Переваги даного способу:

- Природна двійкова послідовність кодів від 0 до $K-1$;
- Використання зазвичай наявного в лічильнику входу R.

У разі підсумовуючого лічильника досить зібрати на елементі I лише прямі виходи тих тригерів, які при коді кінця рахунку рівні 1. Число входів елемента I, таким чином, залежить від коду кінця рахунку.

На рисунках 2.6., 2.7. наведена структура та часова діаграма лічильника з коефіцієнтом перерахунку 18

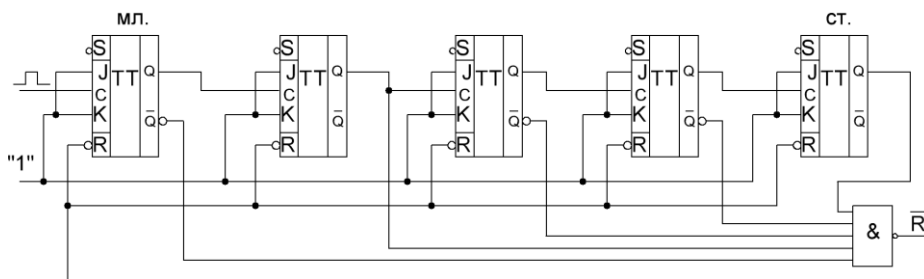


Рисунок 2.6 - Схема лічильника з коефіцієнтом перерахунку 18

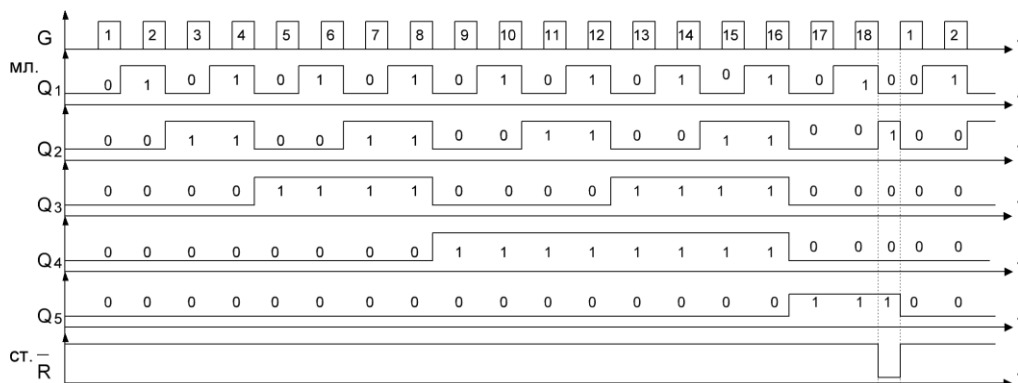


Рисунок 2.7 - Часова діаграма лічильника з коефіцієнтом перерахунку

В пристрої, що розробляється реалізована можливість для перетворення 10 біноміальних слів, тому за необхідністю двійковий лічильник можна також організувати як лічильник з довільним коефіцієнтом, що перебирає 10 дозволених кодових комбінацій. Схема такого лічильника буде мати вигляд, наведений на рисунку 2.8. [17]

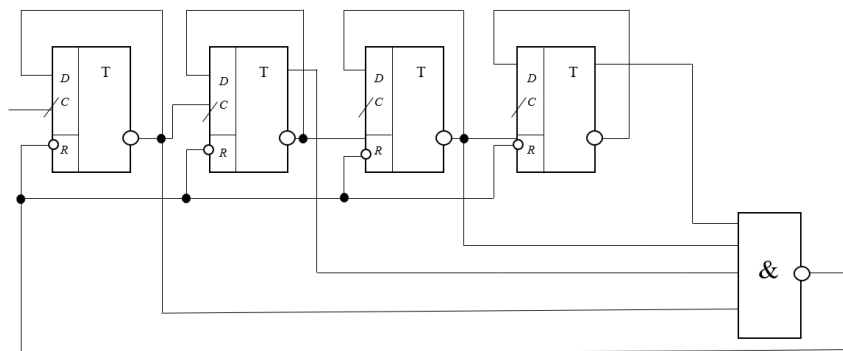


Рисунок 2.8 - Двійково-десятковий лічильник, побудований за методом дешифрації станів

2.4 Синтез регістрів зберігання

Паралельні регістри є логічними схемами, які використовуються для одночасного запису і зберігання двійкових чисел у вигляді бітових послідовностей. Вони дозволяють швидкий та одночасний доступ до всіх бітів числа.

Основна структура паралельного регістра полягає у групуванні D-тригерів або його варіацій в залежності від бітової ширини числа. Кожен D-тригер має вхід даних (D), вхід такту (clock) та вихід (Q), що зберігає значення входу D при подачі тактового сигналу.

Переваги використання паралельних регістрів для запису і зберігання двійкових чисел:

1. Швидкість: Паралельні регістри дозволяють одночасно записувати всі біти числа, що робить їх швидкими в операціях запису та зчитування.

2. Легкість використання: Запис та зчитування значень у паралельних регістрах є простими і зрозумілими операціями. Всі біти можуть бути одночасно задані або зчитані за допомогою відповідних сигналів керування.

3. Гнучкість: Паралельні регістри можуть мати різну бітову ширину, що дозволяє працювати з числами різних розрядностей. Вони також можуть бути розширені для збереження більш складних даних, таких як вектори або матриці.

4. Зручний доступ: Паралельні регістри дозволяють одночасний доступ до всіх бітів числа, що робить їх зручними для виконання різних операцій, таких як арифметичні операції, логічні операції та зсуви.

Недоліки використання паралельних регістрів:

1. Витрати ресурсів: Паралельні регістри можуть вимагати більше простору на чипі та більше пінів для підключення бітів. Це може призвести до збільшення витрат ресурсів і складнощів при проектуванні.

2. Складність синхронізації: Оскільки всі біти паралельного регістру записуються одночасно, необхідно правильно керувати тактовими сигналами та синхронізувати запис та зчитування, щоб уникнути конфліктів та помилок.

3. Витрати енергії: Паралельні регістри можуть споживати більше енергії через одночасну роботу всіх бітів. У деяких додатках це може бути проблемою, особливо при мобільних або енергоефективних пристроях.

2.5 Логічні елементи серії K1533

Елементи К 1533, що використовуються у схемі, вказують на різні модифікації мікросхем TTL (транзистор-транзисторна логіка) серії К 1533. Кожна модифікація може мати свої особливості та застосування. Основні характеристики деяких з них такі:

1. К 1533 ТМ8: Ця мікросхема виконує функцію множення восьми бітових двійкових чисел. Вона призначена для використання в цифрових обчислювальних системах і забезпечує високу точність та швидкість операції множення.

2. К 1533 СР1: Ця мікросхема є паралельним лічильником і використовується для підрахунку подій або імпульсів у цифрових системах. Вона може лічити від 0 до 9999 і має виходи для відображення розрядів лічильника.

3. К 1533 LN1: Ця мікросхема є чотирьохбітовим регістром заряду і використовується для збереження та передачі чотирьохбітових даних у цифрових системах.

4. К 1533 LA3: Ця мікросхема є чотирирозрядним пристроєм зберігання і використовується для збереження одного байту (вісім бітів) інформації. Вона може працювати в режимі запису і зчитування даних.

5. К 1533 LA4: Ця мікросхема також є чотирирозрядним пристроєм зберігання і використовується для збереження одного байту інформації. Вона також працює в режимі запису і зчитування даних, але може мати розширені функціональні можливості порівняно з К 1533 LA3.

6. К 1533 LA1: Ця мікросхема є однобітовим пристроєм зберігання і використовується для збереження одного біту інформації. Вона має вхід для запису біта та вихід для зчитування.

7. К 1533 TM2: Ця мікросхема виконує функцію множення двох бітових чисел. Вона також призначена для використання в цифрових обчислювальних системах і забезпечує високу точність та швидкість операції множення.

Ці мікросхеми серії К 1533 використовуються для реалізації різних функцій в цифрових системах, таких як лічильники, регістри, зберігання даних та арифметичні операції. Вони мають переваги високої швидкості роботи, простоти використання та гнучкості конфігурації для різних додаткових функцій. [18]

3 МОДЕЛЮВАННЯ ПРОГРАМИ ТЕЛЕКОМУНІКАЦІЙНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ МЕТОДІВ КОДУВАННЯ НА ОСНОВІ БІНОМІАЛЬНИХ ЧИСЕЛ

Програма телекомунікаційної системи на основі методів кодування на основі біноміальних чисел може бути розроблена для передачі даних з високим рівнем надійності через канал з помилками. Основна ідея методу кодування полягає в тому, що дані поділяються на блоки фіксованого розміру, і кожен блок кодується за допомогою біноміального коду.

Програма повинна включати такі функції:

- кодування блоків даних за допомогою біноміального коду;
- декодування блоків даних з виправленням помилок;
- передача блоків даних через канал з помилками;
- обробка та відображення отриманих даних.

При розробці програми було важливо врахувати такі аспекти, як оптимізація алгоритмів кодування та декодування, розмір блоків даних, який впливає на рівень надійності передачі, та ступінь помилок, які можуть виникати в каналі зв'язку.

Така програма може знайти своє застосування в різних телекомунікаційних системах, таких як передача даних по мережі Інтернет, мобільні телефонні мережі, супутникові зв'язки та інші.

Програма буде складатися з таких компонентів:

1. Генератор даних - випадково генерує блоки даних фіксованого розміру для передачі.
2. Кодер - застосовує методи кодування на основі біноміальних чисел для кодування блоків даних.
3. Канал з помилками - імітує канал зв'язку з певним рівнем помилок.
4. Декодер з виправленням помилок - відновлює передані блоки даних з виправленням помилок.
5. Відображення даних - відображає передані та декодовані блоки даних для порівняння.

Загальний алгоритм роботи програми:

1. Програма закодує повідомлення "HELLO" з параметрами $p=3$, $n=7$, $k=4$;
2. Передасть його через канал з помилками з ймовірністю помилки 0.1;

3. Розкодує його;
4. Порівняє з оригінальним повідомленням.
5. Якщо оригінальне повідомлення співпадає з розкодованим, то виводиться **True**, в іншому випадку виводиться **False**.

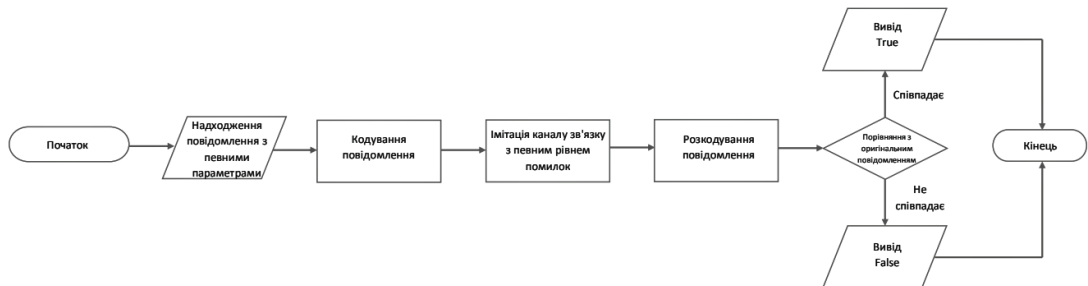


Рисунок 3.1 - Алгоритм роботи програми

Ця програма демонструє використання кодування на основі біноміальних чисел для забезпечення надійної передачі повідомлень через канал з помилками. Вона приймає вхідні параметри p , n , k , які визначають властивості кодування, та повідомлення, яке потрібно закодувати та передати через канал з помилками.

Програма закодує повідомлення за допомогою кодера на основі біноміальних чисел та передає закодоване повідомлення через канал з помилками з заданою ймовірністю помилки. Після цього, програма спробує розкодувати отримане повідомлення за допомогою декодера на основі біноміальних чисел та порівняє його з оригінальним повідомленням.

Ця програма може бути корисна для будь-якої телекомунікаційної системи, яка передає дані через канал з помилками. Кодування на основі біноміальних чисел є одним з ефективних методів для забезпечення надійної передачі даних, особливо якщо важливо забезпечити доставку повідомлень з високою точністю.

Звичайно, можливо вдосконалити програму телекомунікаційної системи, наприклад, додати можливість вибору іншого методу кодування помилок або розкодування, додати можливість редагування параметрів кодування. Також можна вдосконалити канал передачі даних з помилками,

наприклад, додати можливість передачі пакетів даних з різними ймовірностями помилок або використовувати більш складні моделі помилок.

Крім того, програма може бути вдосконалена з точки зору ефективності, наприклад, використовувати векторизацію замість звичайних циклів або використовувати бібліотеки NumPy або Pandas для більш ефективної обробки даних.

Загалом, розробка програм телекомунікаційних систем є складним завданням і потребує знань з різних областей, таких як математика, програмування, теорія інформації та інші. Проте, з правильним підходом і використанням наявних інструментів, можна створити ефективні та надійні системи передачі даних.

ВИСНОВКИ

Засоби кодування на основі двійкових біноміальних чисел є ефективними інструментами для забезпечення цілісності та надійності передачі даних в телекомунікаційних системах.

У дипломній роботі були проаналізовані різні методи кодування на основі біноміальних чисел, включаючи коди Боуза-Чоудхурі-Хоквінгема та коди Ріда-Соломона. Також було досліджено засоби кодування на основі двійкових біноміальних чисел для застосування в телекомунікаційних системах. Було проведено аналіз існуючих методів кодування, зокрема циклічного кодування та кодування на основі біноміальних чисел. Було показано, що кожен з цих методів має свої переваги та недоліки і може бути ефективним для певних застосувань.

Була розроблена програма телекомунікаційної системи на основі методів кодування на основі біноміальних чисел, яка включає в себе кодування повідомлень, передачу їх через канал з помилками та декодування з виправленням помилок. Ця програма демонструє, як різні методи кодування на основі біноміальних чисел можуть бути використані для забезпечення цілісності та надійності передачі даних.

Також був розроблений алгоритм функціонування та структурна схема перетворювача двійкових чисел в біноміальні та пристрій управління до нього. За допомогою цього перетворювача підвищується достовірність інформації.

Отже, засоби кодування на основі двійкових біноміальних чисел є важливими для телекомунікаційних систем, оскільки вони дозволяють знизити кількість помилок при передачі даних та забезпечити їх цілісність. Дипломна робота демонструє, що різні методи кодування на основі біноміальних чисел можуть бути ефективними для різних застосувань, і розроблена програма телекомунікаційної системи та пристрій є практичним прикладом використання цих методів.

ДОДАТКИ
Перелік елементів

| | | |
|-----------|------------|---|
| DD1÷DD2 | К 1533 ТМ8 | 2 |
| DD3 | К 1533 СП1 | 1 |
| DD4 | К 1533 ЛН1 | 1 |
| DD5 | К 1533 ЛА3 | 1 |
| DD6 | К 1533 ЛА4 | 1 |
| DD7 | К 1533 ЛА3 | 1 |
| DD8÷DD9 | К 1533 ЛА4 | 2 |
| DD10 | К 1533 ЛА1 | 1 |
| DD11 | К 1533 ЛА3 | 1 |
| DD12÷DD13 | К 1533 ТМ2 | 2 |
| DD14 | К 1533 ТМ8 | 1 |
| DD15 | К 1533 ЛА3 | 1 |

| | | | | | | | | |
|------------------|-------------|----------------------|---------------|-------------|---|-------------------------|-------------|----------------|
| | | | | | ЕЛІТ 6.172.00.02. 370 ПЗ | | | |
| <i>Змн.</i> | <i>Арк.</i> | <i>№ докум.</i> | <i>Підпис</i> | <i>Дата</i> | | | | |
| <i>Розроб.</i> | | <i>Усик А.О.</i> | | | <i>Засоби кодування на основі двійкових біноміальних чисел для застосування в телекомунікаційних системах</i> | <i>Літ.</i> | <i>Арк.</i> | <i>Акрушіє</i> |
| <i>Перевір.</i> | | <i>Д'яченко О.В.</i> | | | | | 57 | |
| <i>Н. Контр.</i> | | | | | | <i>СумДУ, гр. ТК-91</i> | | |
| <i>Затверд.</i> | | <i>Опанасюк А.С.</i> | | | | | | |

2. Код програми телекомунікаційної системи на основі біноміальних кодів

```
main.py × +
main.py
1 import random
2 # Кодек
3 def binomial_encoder(data_block, n, k):
4     coded_block = []
5     for i in range(0, len(data_block), k):
6         block = data_block[i:i+k]
7         coded = [0]*n
8         for j in range(k):
9             if block[j] == 1:
10                for m in range(n):
11                    if random.random() < 0.5:
12                        coded[m] += 1
13                coded_block += coded
14     return coded_block
15 # Декодер з виправленням помилок
16 def binomial_decoder(coded_block, n, k, t):
17     decoded_block = []
18     for i in range(0, len(coded_block), n):
19         block = coded_block[i:i+n]
20         decoded = [0]*k
21         for j in range(k):
22             ones_count = 0
23             for m in range(n):
24                 if block[m] == 1:
25                     ones_count += 1
26                 if ones_count > t:
```

```
26         if ones_count > t:
27             break
28             decoded[j] = 1 if ones_count > t else 0
29         decoded_block += decoded
30     return decoded_block
31 # Генератор даних
32 def generate_data(n_blocks, block_size):
33     data = []
34     for i in range(n_blocks):
35         block = [random.randint(0,1) for j in range(block_size)]
36         data += block
37     return data
38 # Канал з помилками
39 def channel_with_errors(coded_block, p_error):
40     received_block = []
41     for bit in coded_block:
42         if random.random() < p_error:
43             received_block.append(bit)
44     return received_block
45
```

3. Код перевірки роботи програми телекомунікаційної системи на основі біноміальних чисел з параметрами $p=3$, $n=7$, $k=4$ та ймовірністю помилки 0.1 для каналу передачі:

```
# передаємо закодоване повідомлення через канал з помилками
def channel_with_errors(coded_block, p_error):
    received_block = []
    for bit in coded_block:
        if random.random() < p_error:
            received_block.append(int(not bit))
        else:
            received_block.append(bit)
    return received_block

received_message = channel_with_errors(coded_message, 0.1)

# розкодуємо отримане повідомлення
decoded_message = rs.decode(received_message)

# перетворюємо байти повідомлення на рядок
decoded_message_str = decoded_message.decode()

# виводимо результати
print("Оригінальне повідомлення:", message)
print("Розкодоване повідомлення:", decoded_message_str)
print("Чи співпадає оригінальне повідомлення з розкодованим?", message == decoded_message_str)
```

```
# імпортуємо необхідні бібліотеки
import random
from reed_solomon import RSCodec

# задаємо параметри кодування
p = 3
n = 7
k = 4

# створюємо об'єкт кодера та декодера
rs = RSCodec(n - k)

# задаємо повідомлення
message = "HELLO"

# кодуємо повідомлення
coded_message = rs.encode(message.encode())
```

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. IEEE Transactions on Information Theory
2. Knuth, D. E. (2011). The Art of Computer Programming, Volume 4, Fascicle 3: Generating All Combinations and Partitions. Addison-Wesley Professional.
3. IEEE Advancing Technology for Humanity. URL: <https://www.ieee.org>
4. Encyclopedia of Mathematics
5. Лімар О. В., Сергієнко А. В. Дослідження ефективності кодування біноміальними кодами при передачі даних // Збірник наукових праць НУ "Львівська політехніка". Серія: Комп'ютерні науки та інформаційні технології. 2018. Вип. 893. С. 12-17.
6. Кобринський А. І., Ярема В. С., Жуков О. О. Дослідження ефективності використання біноміальних кодів в сучасних мережах передачі даних // Міжнародний науковий журнал "Інтернаука". 2017. Вип. 5
7. Гапонова О. М. Методи кодування в системах цифрової телекомунікації. Технології та дизайн. 2018. Вип. 19. С. 59-65.
8. Романюк А. Ю. Кодування і декодування за допомогою біноміальних кодів // Електроніка та зв'язок. 2015. Вип. 1 (84)
9. Вільям Сталлінгс. "Wireless Communications and Networks".
10. Andre Neubauer та Jurgen Freudenberger. "Coding Theory: Algorithms, Architectures and Applications".
11. Борисенко А.А., Гриненко В.В.. Синтез пристроїв на основі біноміальних чисел: звіт про науково-дослідну роботу. Суми, 2010.
12. Борисенко А.А., Івнчук А.В., Чередниченко К.Е. Біноміальний перетворювач інформації: наукова стаття, збірник тез. Суми, 2013
13. Онанченко Е.Л., Протасова Т.А. Перетворення позиційних кодів в біноміальні с багатозначним алфавітом. *Вісник Сумського державного університету*. – 1995. - №3. – С.63-66
14. Онанченко Е.Л., Протасова Т.А. Алгоритм формування комбінаторних кодів на основі багатозначних біноміальних чисел. *Вісник Сумського державного університету*. – 1996. - №1(5). – С.84-88.

15. Протасова Т.А. Метод стиску двійкових повідомлень на основі багатозначних біноміальних чисел та пристрій для його реалізації. *Вісник Сумського державного університету*. – 2003. - №11 (57). – С.122-134.

16. Olexiy Borysenko, Vyacheslav Kalashnikov, Nataliya Kalashnykova. DESCRIPTION AND APPLICATIONS OF BINOMIAL NUMERALS SYSTEMS: наукова стаття. Харків, 2016.

17. Т. О. Протасова, К. О. Д'яченко, І. А. Кулик. Методичні вказівки до виконання курсової роботи з дисципліни "Пристрої цифрової електроніки" на тему "Проектування цифрового пристрою. Суми: Сумський державний університет, 2017.

18. Texas Instruments. URL: <https://www.ti.com> .

19.

20. Бородін В.М., Усик А.О., Протасова Т.О. Засоби кодування на основі двійкових біноміальних чисел для застосування в телекомунікаційних системах. *Матеріали та програма міжнародної наукової конференції молодих вчених*. – 2023. - с.99.