**УДК 681.3**

**В.О. Волошок, В.С. Глухов, К.І. Ліпіч, А.В. Юрчук**
Національний університет "Львівська політехніка",
кафедра електронних обчислювальних машин

# ПОРОГОВЕ ДЕКОДУВАННЯ ЗГОРТКОВИХ КОДІВ



**Описано принцип завадостійкого кодування інформації за допомогою згорткових кодів, а також побудову порогового декодера для їх розкодування.**

**In this article principle of antinoise code of information is described by trellis codes, and also construction of threshold decoder for their decoding.**

## Вступ

Процес передавання інформації у будь-якому каналі зв'язку супроводжується дією сторонніх чинників (шумів або завад), які негативно впливають на зміст повідомлення. Існує два основні шляхи боротьби з цими явищами:

використання таких систем сигналів у каналах і таких методів приймання, які б забезпечили найменшу ймовірність помилок;

виправлення помилок шляхом використання для передавання сигналів каналами зв'язку завадостійких кодів.

Існує багато завадостійких кодів, серед них – згорткові. Широковідомий декодер згорткового коду – декодер Вітербі, але у деяких випадках він вимагає великих витрат апаратури або часу на виправлення помилок. Тому актуальним є застосування інших способів декодування (граничного або порогового декодування), які в окремих випадках дають оптимальніше рішення порівняно зі стандартними підходами. Особливості порогового декодування згорткових кодів розглянуто у цій статті.

## Аналіз публікацій та окреслення проблеми

Системою передачі двійкової інформації (СПДІ) називають сукупність джерел і одержувачів повідомлень, передавачів, приймачів і каналів зв'язку, що забезпечують передачу дискретної інформації з визначеними властивостями [1].

СПДІ повинна забезпечувати:

достовірність передавання інформації;

швидкість передавання інформації;

надійність системи передавання інформації;

розкодування вхідної послідовності за однократного її отримання.

Для забезпечення цих вимог потрібне завадостійке кодування, способи використання якого розглянуто в [2–12].

Для декодування завадостійких згорткових кодів широко використовують алгоритм декодування за максимумом правдоподібності, відомий як алгоритм Вітербі [11]. Але декодери, які працюють за алгоритмом Вітербі, порівняно з пороговими декодерами мають велику затримку на декодування і, як наслідок, низьку швидкодію.

Порогові декодери, розглянуті у публікаціях [13, 14], відповідають усім чотирьом вищезгаданим вимогам, забезпечують високу швидкодію і малі апаратні витрати на реалізацію. Але аналіз їхньої роботи порівняно з роботою декодера Вітербі не проводили. Розгляду цього питання присвячено цю статтю.

## Цілі статті

Декодери, які працюють за алгоритмом Вітербі [11], за малих шумів мають велику затримку на декодування і, як наслідок, низьку швидкодію. Тому ціллю статті є синтез та перевірка роботи

декодера, який (так само як і декодер Вітербі) забезпечує виявлення і виправлення до двох помилок, але за деяких умов забезпечує меншу затримку на декодування, більшу швидкодію.

**Синтез порогового декодера згорткових кодів**

На рис. 1 показано каскадну схему двійково-симетричного каналу зв'язку [1]. За каскадними схемами кодування одержано високу достовірність за високого рівня шумів і середнього степеня складності декодера. За каскадного кодування використовують два, а то й більше, ступені кодування. За допомогою першого ступеня (внутрішній код) доводять вірогідність помилки у вихідному каналі до визначеної величини. Потім кодом другого ступеня (зовнішній код) кодують отриманий «рафінований» дискретний канал і знижують імовірність помилки до заданої величини. Виявляється, що каскадний код ефективніший і простіший у реалізації, ніж простий блоковий код такої ж довжини. Під час кодування внутрішній код утворюється кодуванням Ріда–Соломона, а зовнішній – згортковим кодуванням.

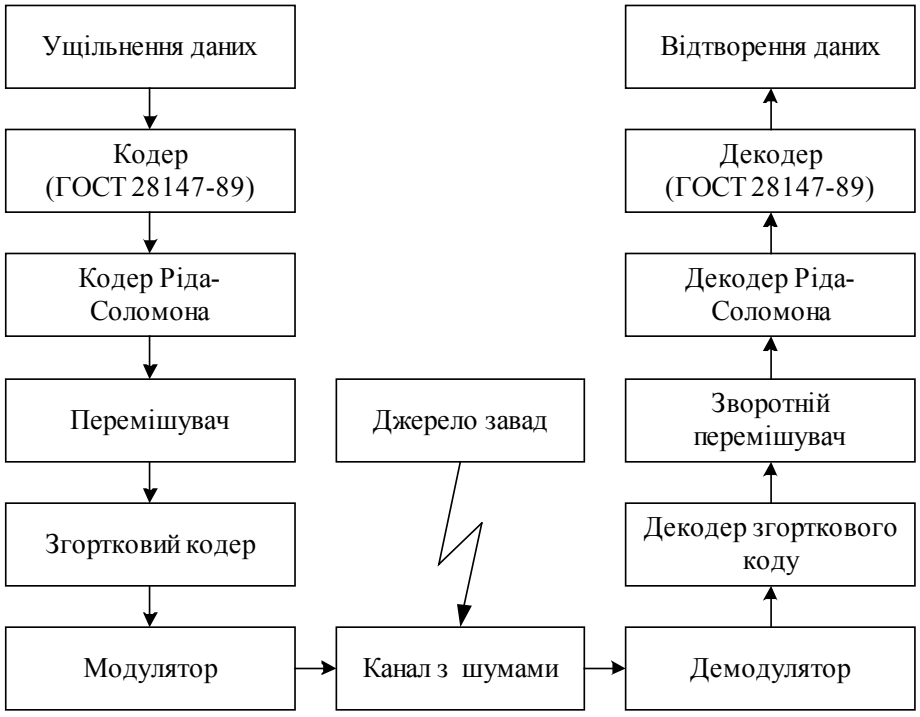


*Рис. 1. Канал зв'язку*

Можна виділити три найважливіші напрямки в теорії і практиці декодування згорткових кодів [9]:

– послідовне декодування являє собою таку форму оброблення закодованої інформації, за якої у буфері декодера послідовно накопичуються прийняті символи і жодна інша послідовність символів не може бути прийнята і декодована, поки не готове рішення про правильність приймання символів, що знаходяться в буфері. Основний недолік цього методу полягає в тому, що часто вірогідність переповнення буфера є більшою за вірогідність помилкового декодування;

– декодування за максимумом правдоподібності, відомий як алгоритм Вітербі [11]. Алгоритм Вітербі для декодування згорткових кодів забезпечує кращу вірогідність і більшу швидкодію виправлення помилок, ніж всі існуючі алгоритми декодування для каналів із середнім і високим рівнем шумів;

– порогове (граничне, мажоритарне) декодування полягає в тому, що під час приймання закодованого повідомлення формується перевірковий вектор – синдром, або послідовність, отримана в результаті його лінійного перетворення; перевірочний вектор подається на вхід граничного елемента під час корекції помилок.

Порогове декодування грунтується на системі перевіркових рівнянь. Кожний символ повідомлення подають *d* різними незалежними способами у вигляді лінійних комбінацій інших символів (*d* – мінімальна кодова відстань). Для виявлення s помилок потрібно скласти $d = s+1$ рівняння, а для виправлення s помилок – $d = 2s+1$ рівнянь.

Для складання потрібної кількості рівнянь кожний символ повідомлення повинен передаватися каналом зв'язку декілька разів. Для цього перед видачею символа в канал зв'язку його підсумовують за модулем 2 з попередніми символами, що обумовлює введення до складу кодера лінії затримки і суматорів за модулем 2. Приклад найпростішого кодера наведено на рис. 2.

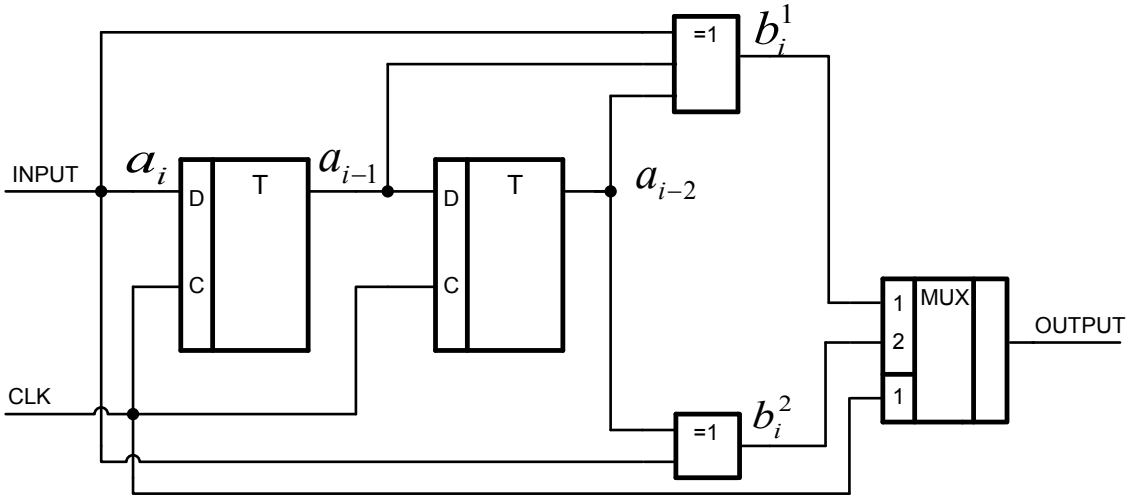


*Рис. 2. Схема кодера для перетворення послідовності бітів у згортковий код*

Нехай на вхід кодера (рис. 2) надходить деяка послідовність 0101110010 зі швидкістю *k* біт/с. Селектор-мультиплексор, встановлений на виході, працює з удвічі більшою частотою, ніж швидкість надходження біт на вхід кодера, причому за першу половину такту роботи кодера селектор зчитує дані з елемента XOR2, а за другу половину такту – з логічного елемента XOR3. У результаті, кожному вхідному біту ставиться у відповідність два вихідних біта, тобто дибіт, перший біт якого формується елементом XOR2, а другий – елементом XOR3.

На рис. 3 наведено часову діаграму роботи кодера.

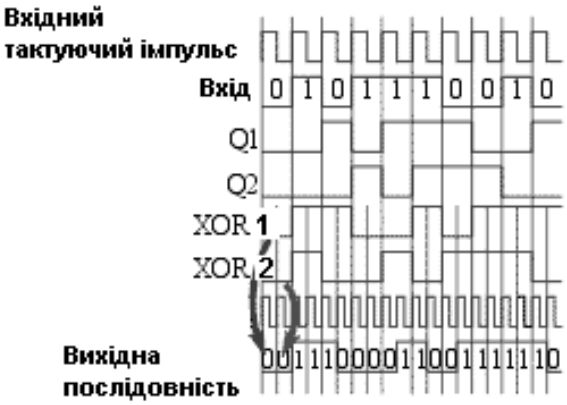


*Рис. 3. Часова діаграма роботи кодера*

За часовою діаграмою (рис. 3) нескладно визначити, що за вхідної послідовності 0101110010 вихідна послідовність матиме вигляд: 00 11 10 00 01 10 01 11 11 10.

Роботу вищерозглянутого кодера описують за допомогою графа (рис. 4).

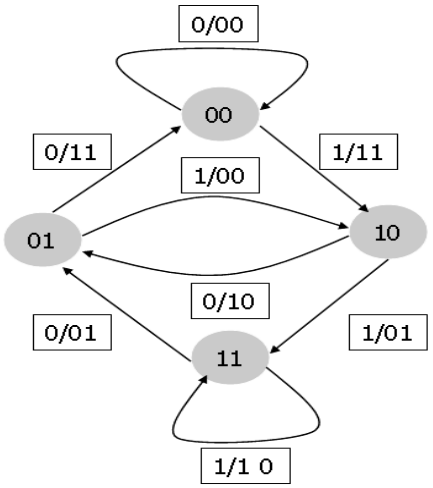


*Рис. 4. Граф, який описує роботу згорткового кодера*

Із схеми кодера (рис. 2) видно, що вхідний інформаційний біт може потрапити на вихід п'ятьома різними шляхами.

Ці шляхи описуються робочими формулами (1) для сигналів на виході кодера (*b*) і декодера (*a*) [13].

$$
\begin{aligned}
b_i^1 &= a_i \oplus a_{i-1} \oplus a_{i-2} & a_i^{(5)} &= b_i^1 \oplus a_{i-1} \oplus a_{i-2} \\
b_i^2 &= a_i \oplus a_{i-2} & a_i^{(4)} &= b_i^2 \oplus a_{i-2} \\
b_{i+1}^1 &= a_{i+1} \oplus a_i \oplus a_{i-1} & a_i^{(3)} &= b_{i+1}^1 \oplus b_{i+1}^2 \\
b_{i+1}^2 &= a_{i+1} \oplus a_{i-1} & a_{i+1} &= b_{i+1}^2 \oplus a_{i-1} \\
b_{i+2}^1 &= a_{i+2} \oplus a_{i+1} \oplus a_i & a_i^{(1)} &= b_{i+2}^1 \oplus b_{i+3}^1 \oplus b_{i+3}^2 \oplus b_{i+1}^2 \oplus a_{i-1} \\
b_{i+2}^2 &= a_{i+2} \oplus a_i & a_i^{(2)} &= b_{i+2}^2 \oplus b_{i+3}^1 \oplus b_{i+3}^2 \\
b_{i+3}^1 &= a_{i+3} \oplus a_{i+2} \oplus a_{i+1} & a_{i+2} &= b_{i+3}^1 \oplus b_{i+3}^2 \\
b_{i+3}^2 &= a_{i+3} \oplus a_{i+1}
\end{aligned}
\qquad (1)
$$

З формул видно, що значення інформаційного біта *a* у декодері можна отримати п'ятьма різними способами, причому беруть до уваги:

1) значення двох вже визначених попередньо бітів *a* (що вносить затримку на 2*2=4 такти);

2) значення двох наступних вхідних дибітів *b* (що також вносить затримку на 4 такти).

Загалом затримка на прийняття рішення дорівнює 8-ми тактам, не залежить від довжини повідомлення, а тільки від кількості тригерів у схемі кодера (рис. 2).

На рис. 5 показано функціональну схему порогового декодера, який працює згідно з наведеними формулами.

Декодер складається з таких основних частин:

регістра зсуву RGb, який слугує для отримання із закодованої послідовності сигналів $b_i^1$, $b_i^2$;

регістра зсуву RGa, який слугує для отримання із закодованої послідовності сигналів $a_{i-1}$, $a_{i-2}$;

матриці логічних елементів XOR, яка слугує для схемної реалізації наведених формул;

порогового елемента M, який пропускає на вихід той сигнал, який частіше з'являється на його входах (сигнал, імовірність якого найвища). Логіку роботи порогового елемента описує формула (2)

$$F = abc \vee abd \vee abe \vee acd \vee ade \vee ace \vee bcd \vee bce \vee bde \vee cde, \qquad (2)$$

де *a, b, c, d, e* – позначення входів порогового елемента.

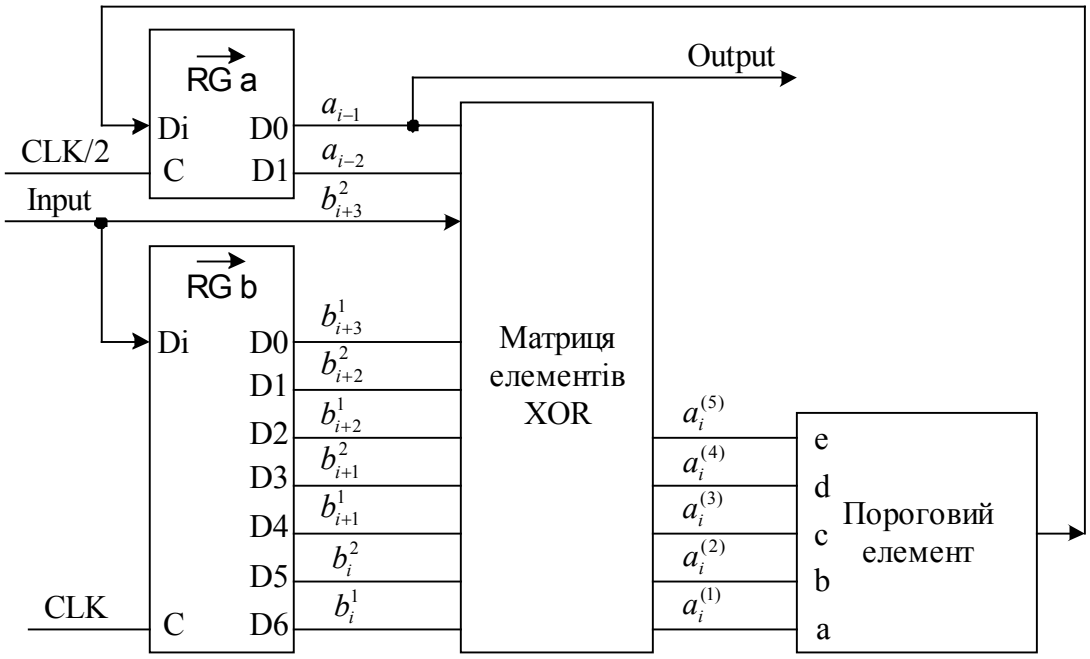


*Рис. 5. Схема порогового декодера*

У цьому випадку пороговий елемент є мажоритарним елементом.

На рис. 6 наведено часову діаграму роботи порогового декодера у двійково-симетричному каналі зв'язку для наведеного прикладу.

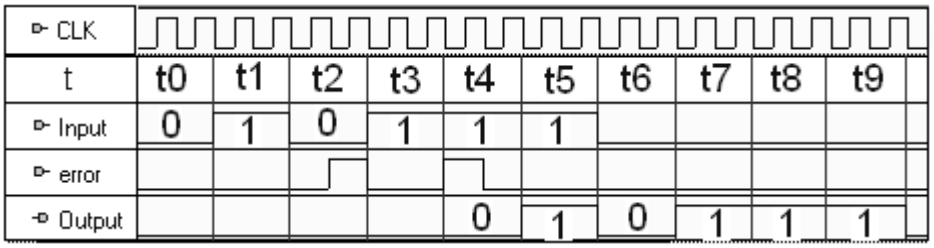


*Рис. 6. Часова діаграма роботи порогового декодера*

На рис. 6 позначено:

CLK – послідовність синхроімпульсів;

Input – не закодований вхідний сигнал:

Output – сигнал, знятий з виходу декодера;

Error – помилки, внесені у вхідний сигнал.

З часової діаграми видно, що декодований сигнал на виході декодера з'являється із затримкою на 8 тактів від початку отримання закодованої послідовності (після надходження 4-х дибітів). Цей декодер працює адекватно, якщо серед 8-ми послідовних вхідних бітів є не більше двох помилкових.

Декодер Вітербі, з яким порівнюється робота порогового декодера, має схему (рис. 7).

Декодер Вітербі містить такі блоки:

вузол керування (Control);

вузол генерування метричних переходів (BMG). Вузол призначений для обрахунку відстані Хеммінга між отриманим дибітом і дибітами, які можливо отримати;

вузол додавання – порівняння – вибірки (ACS). Вузол призначений для вибору найменшої сумарної помилкової метрики;

вузол зворотного відстеження (Traceback). Вузол відтворює правильну послідовність бітів;

вузол керування пам’яттю остаточного залишку (MMU);
пам’ять остаточного залишку (RAM Survivor);
пам’ять метрик (RAM Metric).

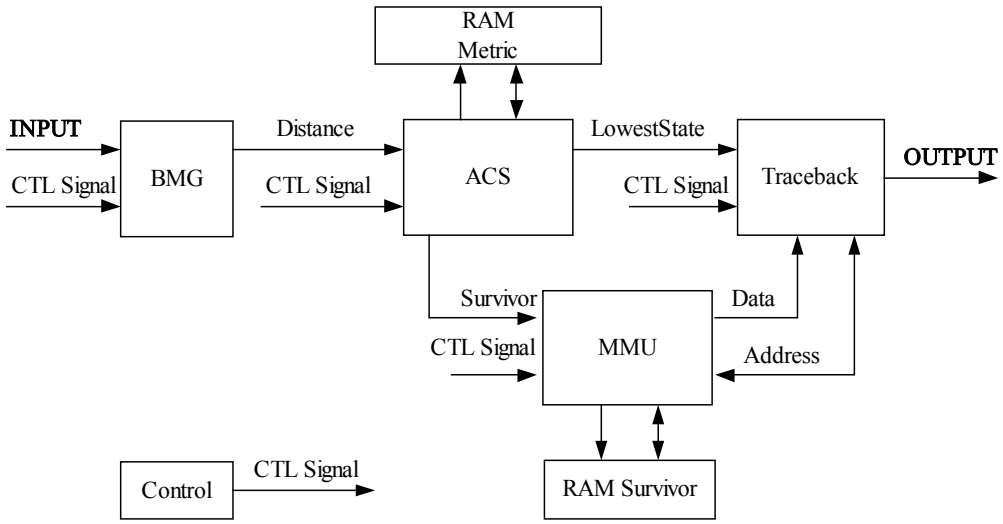


*Рис. 7. Структурна схема декодера Вітербі*

Пороговий декодер порівнювали з декодером Вітербі для випадку передавання двійково-симетричним каналом зв’язку послідовності 010111 – перших 6 біт (рис. 3), код на виході кодера матиме вигляд: 00 11 10 00 01 10.

Принцип і результати декодування прийнятої послідовності за допомогою декодера Вітербі зручно для наочності зображати за допомогою часової діаграми станів, де за допомогою стрілок зображено усі можливі переходи у заданий момент часу, а над кожною стрілкою надписано значення метрики помилок (величини, яка відповідає кількості інформаційних бітів, які змінили своє значення внаслідок дії завад) і через риску – значення вхідного дибіта, що відповідає заданому переходу [12]. Результуючу метрику помилок утворюють шляхом підсумовування метрик від попередніх переходів. Після отримання найімовірнішого шляху усі зайві шляхи декодер відкидає. Отже, у момент часу t=6 (після 12 тактів роботи і прийняття всіх дибітів повідомлення) декодер може прийняти рішення і визначити шлях, за яким у зворотному напрямку можна визначити вхідну послідовність, яка є найбільш правильною.

Результат розкодування того ж самого повідомлення за допомогою порогового декодера показано на рис. 6. З наведеної часової діаграми видно, що затримка між початком надходження закодованої послідовності на вхід порогового декодера і виходом першого декодованого біта (затримка на декодування) у цьому випадку менша ніж у декодера Вітербі і дорівнює 8 тактам (менша у 1,5 рази). Це і є основною перевагою порогового декодера над декодером Вітербі. Пороговий декодер приймає рішення про значення вихідного розряду після аналізу обмеженої кількості дибітів повідомлення (у наведеному прикладі – 4), а декодер Вітербі – після аналізу усіх дибітів повідомлення. Це одночасно є і недоліком порогового декодера, оскільки при прийнятті рішення не враховують спотворення попередніх дибітів повідомлення.

## Висновки

У статті описано пороговий декодер згорткового коду, який для повідомлень малого розміру і для каналу зв’язку з невеликими завадами має кращі часові параметри, ніж декодер Вітербі. Для розглянутого у статті прикладу час декодування може бути меншим у 1,5 раза.

*1. Глухов В.С., Мельник А.О., Пуйда В.Я. Дослідження шляхів створення кодера та декодера відеосигналу // Вісн. Держ. ун-ту “Львівська політехніка”. – 2003. – Вип. 492. 2. Теоретические*

*основы информационной техники: Учеб. пособие для вузов / Ф.Е. Темников, В.А. Афонин, В.И. Дмитриев. – 2-е изд., перераб. и доп. – М.: Энергия, 1979. – 512 с., Wr W.W., Haccoun D., Peile R., Hirata Y. Coding for Satellite Communication // IEEE Journal on Selected Areas in Communications. – May, 1987. – VoI. SAC-5, № 4. 4. Берлекэмп Э.Р. Техника кодирования с исправлением ошибок // ТИИЭР. – 1980. – № 5. – С. 24–58. 5. Бояринов И.М. Помехоустойчивое кодирование числовой информации. – М.: Наука, 1983. – 189 с. 6. Питерсон У., Уэлдон Э. Коды, исправляющие ошибки / Пер. с англ.; Под ред. Добрушина. – М.: Радио и связь, 1985. – 248 с. 7. Кларк Дж., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи. – М.: Радио и связь, 1987. – 392 с. 8. Дмитриев В.И. Прикладная теория информации: Учеб. для студ. вузов по спец. "Автоматизированные системы обработки информации и управления". – М.: Высш. шк., 1989. – 320 с. 9. Цымбал В.П. Теория информации и кодирование: Учебник. – 4-е изд., перераб. и доп. – К.: Вища. шк., 1992. – 263 с. 10. Муттер В.М. Основы помехоустойчивой телепередачи информации. – М.: Радио и связь, 1994. – 293 с. 11. Витерби А.Г. Границы ошибок для сверточных кодов и ассимптотический оптимальный алгоритм декодирования // Некоторые вопросы теории кодирования. – М.: Мир, 1976. – 360 с. 12. The Art of Error Correcting Coding. Robert H. Morelos-Zaragoz. Copyright © 2002 John Wiley & Sons Ltd. 13. А. с. СССР 492872. Устройство для декодирования линейных сверточных кодов / В.В. Золотарев. 14. Волошок В.О., Ліпіч К.Л., Юрчук А.В. Порогове декодування згорткового коду // Матеріали наук.-техн. конф. ІППТ при Нац. ун-ті "Львівська політехніка". – Львів, 2006.*

**УДК 681.3, 004.3**

**В.С. Глухов**
Національний університет "Львівська політехніка",
кафедра електронних обчислювальних машин

# ОБЧИСЛЮВАЛЬНИЙ ПРИСТРІЙ ДЛЯ ОПЕРАЦІЙ НАД ЕЛІПТИЧНИМИ КРИВИМИ



**Описано особливості побудови обчислювальних пристроїв для операцій над еліптичними кривими.**

**The aspects of dedicated processors arcitecture for operations over elliptic curves are described.**

## Вступ

З 1 січня 2004 року в Україні офіційно дозволено користуватися електронним цифровим підписом замість звичайного. Тому актуальними є розроблення й порівняння різних архітектур спеціалізованих процесорів, які можуть апаратними способами реалізовувати потрібні для отримання і перевірки цифрового підпису операції. Цьому питанню, недостатньо розкритому у вітчизняній літературі, присвячена ця стаття.

## Аналіз публікацій та окреслення проблеми

В Україні діють два стандарти на цифровий підпис: міждержавний стандарт ГОСТ 34.310-95 та національний стандарт України ДСТУ 4145-2002 [1]. Існують національні стандарти в інших країнах, а також міжнародний стандарт IEEE1363-2000 [2]. Математичним основам застосування еліптичних кривих присв'ячені роботи [3–8].

Стандарт [1] визначає використання для отримання цифрового підпису еліптичних кривих і полей Галуа $GF(2^p)$, елементи яких можна подати як у поліноміальному, так і у нормальному