МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп’ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

Ващук Віталій Сергійович

Апаратний засіб модулярного експоненціонування криптоалгоритму RSA / The hardware of modular exponentiation of RSA cryptoalgorithm

напрям підготовки: 6.050102 - Комп’ютерна інженерія
фахове спрямування - Комп’ютерні системи та мережі
Бакалаврська робота

Виконав студент групи КСМ-42/1
Ващук Віталій Сергійович

Науковий керівник: к.т.н.,
Маслияк Б.О

Тернопіль - 2018

РЕЗЮМЕ

Дипломний проект містить 57 сторінок пояснюючої записки, 17 рисунки, 7 таблиць, 1 додатки. Обсяг графічного матеріалу 3 аркуші формату А3.

Метою дипломної роботи є апаратна реалізація засобу модулярного експоненціювання в криптосистемі захисту інформації RSA.

Методи досліджень – методи побудови імітаційних моделей цифрових пристроїв.

В дипломній роботі, на основі аналізу навчальної та наукової літератури, стандартів підготовки фахівців з комп’ютерної інженерії, програм курсу захист інформації в комп'ютерних системах та мережах та комп’ютерна схемотехніка, поставлена та реалізована задача розробки апаратного засобу модулярного експоненціонування криптоалгоритму RSA. В роботі послідовно розглянуті питання аналізу методів побудови сучасних систем захисту інформації на основі симетричних та несиметричних алгоритмів. Розглянуто особливості побудови та функціонування програмних та апаратних засобів. Як актуальну виділено задачу модулярного експоненціювання. Проектування структури апаратного засобу та його компонентів створило умови для моделювання електричних функціональних схем, серед яких виділено пристрої множення та ділення двійкових чисел. Практичне значення мають розроблені та налаштовані в середовищі NI Multisim схеми компонентів та засобу в цілому.

Ключові слова: МІКРОСХЕМА, СТРУКТУРНА СХЕМА, ЕЛЕКТРИЧНА СХЕМА, КОМБІНАЦІЙНА ЛОГІКА, МУЛЬТИПЛЕКСОР, ЛІЧИЛЬНИК, РЕГІСТР, СУМАТОР.

SUMMARY

The thesis project contains 57 pages of explanatory note, 17 drawings, 7 tables, 1 annex. The volume of graphic material is 3 sheets of A3 format.

The purpose of the thesis is the hardware implementation of the modular exponential tool in the cryptosystem of information protection RSA.

Methods of research - methods of constructing simulation models of digital devices.

In the thesis, on the basis of the analysis of educational and scientific literature, the standards of training specialists in computer engineering, programs of the course of information protection in computer systems and networks and computer circuit engineering, the task of developing a hardware device for the modular exponentialization of the cryptographic algorithm RSA is set and implemented. The paper examines the problems of analyzing the methods of constructing modern information security systems on the basis of symmetric and asymmetric algorithms. Features of construction and functioning of software and hardware are considered. How relevant is the problem of modular exponentiation. The design of the structure of the hardware and its components has created conditions for the simulation of electrical functional schemes, among which the devices of multiplication and division of binary numbers are allocated. In practice, NI Multisim has developed and configured components and tools in general.

Key words: MICROSHEMA, STRUCTURAL SCHEME, ELECTRICAL SCHEME, COMBINATION LOGIC, MULTIPLEXOR, LICYLIC, REGISTER, SUMATOR.

ЗМІСТ

					ДП.КСМ.07124/14.00.00.000ПЗ	
Зм.	*Арк*	*№ докум.*	*Підпис*	*Дата*		
Розробив		Ващук В.С.			АПАРАТНИЙ ЗАСІБ МОДУЛЯРНОГО ЕКСПОНЕНЦІОНУВАННЯ КРИПТОАЛГОРИТМУ RSA ПОЯСНЮВАЛЬНА ЗАПИСКА	*Літ.* / *Аркуш* / *Аркушів*
Перевірив		Маслияк Б.О.				
Консульт.		Паздрій І.Р.				ТНЕУ, ФКІТ, КСМ-42/1
Н. Контр.		Гураль І.В.				
Затв.		Березький О.М.				

ВСТУП

Комп'ютерні системи та телекомунікації забезпечують надійність функціонування величезної кількості інформаційних систем самого різного призначення. Більшість таких систем несуть у собі інформацію, що має конфіденційний або персональний характер. Рішення задачі автоматизації процесів обробки даних спричинило нову проблему – проблему інформаційної безпеки. Розуміння основних послуг безпеки, грамотне управління механізмами захисту інформації дозволяє забезпечити надійну безпеку персональних та комерційних даних, використовувати процедури криптографічного перетворення даних з метою забезпечення скритності передачі інформації в мережах Internet.

Сучасне вирішення багатьох проблем захисту інформації неможливо уявити без використання криптографічних методів. Серед численних проблем забезпечення інформаційної безпеки, що вирішуються за допомогою криптографічних методів та засобів, завдання забезпечення цілісності та вірогідності передаваної інформації є актуальним. З урахуванням сучасних вимог до інформаційно - телекомунікаційних систем – це завдання все частіше перетворюється на серйозну проблему. Надто актуальною вона є у державній, військовій, бізнесовій та фінансовій сфері, оскільки задля функціонування електронної платіжної системи неодмінною умовою є зберігання цілісності та вірогідності всіх документів.

Асиметричну криптографію винайдено в семидесятих роках минулого сторіччя. Останніми трьома десятиріччями вона набула широкого розвитку і посіла майже таке саме місце, що й блочне симетричне шифрування. Асиметричне шифрування, або шифрування на відкритому ключі, ґрунтується на докорінно відмінній ідеології, і впевнено посіло власну нішу серед систем захисту інформації.

На сьогодні асиметричне шифрування застосовується для ідентифікації та автентифікації користувачів, захисту каналів передавання даних від нав’язування помилкових даних, захисту електронних документів від копіювання та підробки. Процес криптографічного закриття даних може здійснюватися як програмно, так і апаратно. Апаратна реалізація відрізняється істотно більшою вартістю, проте їй притаманні і переваги: висока продуктивність, простота, захищеність.

Програмна реалізація більш практична, допускає відому гнучкість у використанні. Для сучасних криптографічних систем захисту інформації висуваються наступні загальноприйнятні вимоги:

- зашифроване повідомлення піддається читанню тільки при наявності ключа;
- число операцій, необхідних для визначення ключа шифрування за фрагментом шифрованого повідомлення і відповідного йому відкритого тексту, повинно бути не менше загального числа можливих ключів;
- число операцій, необхідних для розшифрування інформації шляхом перебору ключів повинно мати строгу нижню оцінку і виходити за межі можливостей сучасних комп'ютерів (з урахуванням можливості використання мережевих обчислень);
- знання алгоритму шифрування не повинно впливати на надійність захисту;
- незначна зміна ключа повинна приводити до істотної зміни виду зашифрованого повідомлення навіть при використанні одного і того ж ключа;
- структурні елементи алгоритму шифрування повинні бути незмінними;
- додаткові біти, що вводяться в повідомлення в процесі шифрування, повинні бути надійно сховані в зашифрованому тексті;
- довжина шифрованого тексту повинна бути рівною довжині вихідного тексту;

– не повинно бути простих і легко встановлюваних залежностей між ключами, які послідовно використовуються в процесі шифрування;

– будь-який ключ з множини можливих повинен забезпечувати надійний захист інформації.

Алгоритм повинен допускати як програмну, так і апаратну реалізацію, при цьому зміна довжини ключа не повинно вести до якісного погіршення алгоритму шифрування.

1.ЗАХИСТ ІНФОРМАЦІЇ В КОМП'ЮТЕРНИХ СИСТЕМАХ

1.1 Особливості систем захисту інформації

Мета криптографічного системи полягає в тому, щоб зашифрувати осмислений вихідний текст (також званий відкритим текстом), отримавши в результаті абсолютно безглуздий на погляд шифрований текст (шифротекст, криптограма). Одержувач, якій він призначений, повинен бути здатний розшифрувати (кажуть також "дешифрувати") цей шифротекст, відновивши, таким чином, відповідний йому відкритий текст. Криптографія припускає наявність трьох компонентів: даних, ключа і криптографічного перетворення [1].

Шифрування - перетворювальний процес: вихідний текст, що має також назву відкритого тексту, замінюється шифрованим текстом. Дешифрування - зворотний шифруванню процес. На основі ключа шифрований текст перетвориться у вихідний. Ключ - інформація, необхідна для безперешкодного шифрування й дешифрування текстів.

Вважається, що криптографічне перетворення відомо всім, але, не знаючи ключа, за допомогою якого користувач закрив сенс повідомлення від цікавих очей, потрібно витратити неймовірно багато зусиль на відновлення тексту повідомлення. (Слід ще раз повторити, що немає абсолютно сталого від розкриття методу шифрування. Якість шифру визначається лише грошима, які потрібно викласти за його розкриття від $ 10 і до $ 1000000.)

Розкриттям криптосистеми називається результат роботи криптоаналітиків, що приводить до можливості ефективного розкриття будь-якого, зашифрованого за допомогою даної криптосистеми, відкритого тексту. Ступінь нездатності криптосистеми до розкриття називається її стійкістю.

Криптосистеми поділяються на симетричні (з секретним ключем) і з відкритим ключем [2, 3]. У симетричних криптосистемах і для шифрування, і для дешифрування використовується один і той самий ключ.

У системах з відкритим ключем використовуються два ключі - відкритий і закритий, які математично пов'язані один з одним. Інформація шифрується за допомогою відкритого ключа, що доступний усім бажаючим, а розшифровується за допомогою закритого ключа, відомого тільки одержувачу повідомлення.

Симетричні криптосистеми (також симетричне шифрування, симетричні шифри) (англ. Symmetric-key algorithm) - спосіб шифрування, в якому для шифрування і розшифрування застосовується один і той же криптографічний ключ. До винаходу схеми асиметричного шифрування єдиним існуючим способом було симетричне шифрування. Ключ алгоритму повинен зберігатися в секреті обома сторонами. Алгоритм шифрування вибирається сторонами до початку обміну повідомленнями. В даний час симетричні шифри реалізуються як блокові та поточні шифри.

Блокові шифри обробляють інформацію блоками певної довжини (зазвичай 64, 128 біт), застосовуючи до блоку ключ в установленому порядку, як правило, кількома циклами перемішування і підстановки, званими раундами. Результатом повторення раундів є лавинний ефект - наростаюча втрата відповідності бітів між блоками відкритих і зашифрованих даних [4].

В потокових шифрах шифрування проводиться над кожним бітом або байтом вихідного (відкритого) тексту з використанням гамування. Гамування - метод симетричного шифрування, що полягає в «накладенні» послідовності, що складається з випадкових чисел, на відкритий текст. Послідовність випадкових чисел називається гамма-послідовністю і використовується для шифрування і розшифрування даних. Підсумовування, зазвичай, виконується в будь-якому кінцевому полі. Наприклад, в полі Галуа GF (2) підсумовування набирає вигляду операції «виключне АБО (xor)». Поточний шифр може бути легко створений на основі блочного (наприклад, ГОСТ 28147-89 в режимі гамування), запущеного в спеціальному режимі.

Більшість симетричних шифрів використовують складні комбінації великої кількості підстановок і перестановок. Багато таких шифрів виконуються в кілька (іноді до 80) проходів, використовуючи на кожному проході «ключ проходу». Множина «ключів проходу» для всіх проходів називається «розкладом ключів» (key schedule).

Типовим способом побудови алгоритмів симетричного шифрування є мережа Фейстеля. Алгоритм будує схему шифрування на основі функції F (D, K), де D - порція даних розміром вдвічі менше блоку шифрування, а K - «ключ проходу» для даного проходу. Від функції не потрібно оборотність - зворотна їй функція може бути невідома. Переваги мережі Фейстеля - майже повний збіг дешифрування з шифруванням (єдина відмінність - зворотний порядок «ключів проходу» в розкладі), що значно полегшує апаратну реалізацію.

Операція перестановки перемішує біти повідомлення по якомусь закону. В апаратних реалізаціях вона тривіально реалізується як переключення провідників. Саме операції перестановки дають можливість досягнення «ефекту лавини». Операція перестановки лінійна - f (a) xor f (b) == f (a xor b)

Аналіз особливостей використання симетричних криптосистем, виконаний в першому розділі показує, що в основному, симетричні алгоритми шифрування вимагають менше обчислень, ніж асиметричні. На практиці, це означає, що якісні асиметричні алгоритми в сотні або в тисячі разів повільніші за якісні симетричні алгоритми. Недоліком симетричних алгоритмів є необхідність мати секретний ключ з обох боків передачі інформації. Так як ключі є предметом можливого перехоплення, їх необхідно часто змінювати та передавати по безпечних каналах передачі інформації під час розповсюдження. Тобто до переваг симетричних систем слід віднести:

- швидкість (за даними Applied Cryptography - на 3 порядки вище)
- простота реалізації (за рахунок більш простих операцій)
- необхідна менша довжина ключа для порівнянної стійкості

– вивченість (за рахунок більшого віку)

Недоліками симетричних систем є:

– складність управління ключами у великій мережі. Це означає квадратичне зростання числа пар ключів, які треба генерувати, передавати, зберігати і знищувати в мережі. Для мережі в 10 абонентів потрібно 45 ключів, для 100 вже 4950.

– складність обміну ключами. Для застосування необхідно вирішити проблему надійної передачі ключів кожному абоненту, тому що потрібен секретний канал для передачі кожного ключа обом сторонам.

Для компенсації недоліків симетричного шифрування в даний час широко застосовується комбінована (гібридна) криптографічний схема, де за допомогою асиметричного шифрування передається сеансовий ключ, що використовується сторонами для обміну даними за допомогою симетричного шифрування.

При шифруванні асиметричними ключами (іноді званому шифруванням з відкритими ключами або криптографією з відкритими ключами) ми маємо ту ж саму ситуацію, що і при шифруванні симетричними ключами, але з невеликою різницею. По-перше, ми маємо два ключа замість одного: з них один відкритий ключ (public key), інший - індивідуальний або секретний (private key). Для того щоб передати захищене повідомлення приймачу інформації, передавач спочатку шифрує повідомлення, використовуючи відкритий ключ. Щоб розшифровувати повідомлення, приймач використовує свій власний секретний ключ.

Таким чином, аналіз особливостей використання криптографічних систем показав, що асиметричні системи є більш захищені, але разом з тим складнішими та більш трудомісткими.

1.2 Програмно – апаратні засоби реалізації асиметричних криптосистем

Процес криптографічного закриття даних може здійснюватися як програмно, так і апаратно. Апаратна реалізація відрізняється суттєво більшою вартістю, однак їй властиві й переваги: висока продуктивність, простота, захищеність. Програмна реалізація більш практична, допускає відому гнучкість у використанні [3-6].

Огляд існуючих програмних технологій показує ряд програмних засобів, що здійснюють реалізацію процедури захисту інформації в комп'ютерних мережах. Серед них виділяються наступні.

Advanced Encryption Standard (AES), також відомий як Rijndael – симетричний алгоритм блочного шифрування (розмір блока 128 біт, ключ 128/192/256 біт), прийнятий як стандарт шифрування урядом США за результатами конкурсу AES. Цей алгоритм добре проаналізований і зараз широко використовується, як це було з його попередником DES. Національний інститут стандартів і технологій США (англ. National Institute of Standards and Technology, NIST) опублікував специфікацію AES 26 листопада 2001 після п'ятирічного періоду, в ході якого були створені і оцінені 15 кандидатур. 26 травня 2002 AES було оголошено стандартом шифрування. Станом на 2009 рік AES є одним з найпоширеніших алгоритмів симетричного шифрування. Підтримка AES (і тільки його) введена фірмою Intel в сімейство процесорів x86 починаючи з Intel Core i7-980X Extreme Edition, а потім на процесорах Sandy Bridge.

TrueCrypt- комп'ютерна програма для шифрування «на льоту» (On-the-fly encryption) для 32 - і 64-розрядних операційних систем сімейств Microsoft Windows NT 5 і новіше (GUI-інтерфейс), Linux і Mac OS X. Вона дозволяє створювати віртуальний зашифрований логічний диск, що зберігається у вигляді файлу. За допомогою TrueCrypt також можна повністю шифрувати

розділ жорсткого диска або іншого носія інформації, такої як флоппі-диск або USB флеш-пам'ять. Всі збережені дані в томі TrueCrypt повністю шифруються, включаючи імена файлів і каталогів. Змонтований тому TrueCrypt подібний звичайному логічному диску, тому з ним можна працювати за допомогою звичайних утиліт перевірки та дефрагментації файлової системи.

Ліцензія програми вважалася вільною, однак при її перевірці для включення TrueCrypt в Fedora в жовтні 2008 року були виявлені небезпечні і які роблять її невільною неоднозначності. В список підтримуваних TrueCrypt 6.2 алгоритмів шифрування входять AES, Serpent і Twofish. Попередні версії програми також підтримували алгоритми з розміром блоку 64 біта (Потрійний DES, Blowfish, CAST5) (включаючи версії 5.x, яка могла відкривати, але не створювати розділи, захищені цими алгоритмами). До листопада в ліцензію були внесені виправлення.

Програма DigiSecret застосовує стійкі і перевірені часом алгоритми шифрування, здатна архівувати файли, і повністю їх знищувати - файли видаляються, і їх місце на диску багато разів перезаписується по спеціальному алгоритму, щоб виключити можливість відновлення даних. Загалом то спрямована DigiSecret саме на створення закодованих архівів і передача цих архівів між користувачами. Інтерфейс програми, по суті, нагадує різні архіватори: для створення архіву потрібно перетягнути потрібні файли в основне вікно програми і, вибравши відповідний пункт меню, створити закодований архів. Виробник стверджує, що використовується в DigiSecret механізм компресії надзвичайно ефективний, проте на практиці виявляється, що навіть вбудований в Windows XP механізм компресії ZIP справляється з цим завданням краще. Втім, не це головне завдання програми, і її рівень достатній для середнього користувача. При створенні архіву можна вибрати одну з дев'яти ступенів стиснення відповідно до бажаної швидкістю архівації, або відключити компресію зовсім, і тоді програма просто зашифрує вміст файлів. DigiSecret вміє створювати і SFX архіви, що напевно стане в

нагоді в тих випадках, коли виникає необхідність передати інформацію людині, яка не має своєї копії цієї програми. У створеному архіві зберігається структура папок, хоча в основному вікні програми всі файли «звалені в купу». Програма пропонує на вибір декілька алгоритмів кодування – CAST (128-бітний ключ), Blowfish (448-бітний ключ), Twofish (256-бітний ключ) і Rijndael (також відомий як AES, 256-бітний ключ).

Актуальним є і розробка апаратних реалізацій як симетричних, так і асиметричних криптосистем. Прикладом апаратної реалізації симетричних криптосистем є наступні мікросхеми - Clipper chip, Capstone chip.

Стосовно криптосистеми RSA пропонується використання мікросхеми TPM (Trusted Platform Module). TPM є мікроконтролером, який пропонує засоби для безпечного створення ключів шифрування, здатних обмежити використання ключів (як для підпису так і для шифрування / дешифрування), з тим же ступенем повторюваності, як і генератор випадкових чисел. Так само цей модуль включає наступні можливості: віддалену атестацію, прив'язку, і надійне захищене зберігання. Дистанційна атестація створює зв'язок апаратних засобів, завантаження системи, і конфігурації хоста (ОС комп'ютера), дозволяючи третій особі (на кшталт цифрового магазину музики) перевіряти, що в програмне забезпечення, або музику, завантажену з магазину, не внесено жодних змін. Шифрування кодує дані таким способом, що вони можуть бути розшифровані тільки на комп'ютері, де були зашифровані, під керуванням того ж самого програмного забезпечення). Прив'язка шифрує дані, використовуючи ключ схвалення TPM (унікальний ключ RSA, записаний в чіп в процесі його виробництва), або інший ключ, якому довіряють.

Модуль TPM може використовуватися, щоб підтвердити справжність апаратних засобів. Так як кожен чіп TPM унікальний для специфічного пристрою, це робить можливим однозначне встановлення автентичності платформи. Наприклад, щоб перевірити, що система, до якої здійснюється доступ - очікувана система.

Ідентифікація користувача здійснюється просто. При завантаженні комп'ютера людина повинна підтвердити свою особистість: ввести PIN-код або представити відбиток пальця. Потім, якщо користувач працює з веб-сервісами, які підтримують TPM, вони вже не вимагають його додаткової авторизації, перевіряючи лише номер мікросхеми. Крім того, така система надійно захищає користувача від фішингу, оскільки підроблені сайти просто не зможуть отримати номер TPM. Інші застосування TPM включають в себе:

- управління цифровими правами;
- захист ліцензій на програмні продукти;
- захист паролів.

Можливості використання апаратної ідентифікації можна перераховувати дуже довго. Величезна кількість сервісів і додатків в інтернеті засновані на ідентифікації користувача, і мікросхема TPM спростить і убезпечить її. Але є і труднощі при її використанні.

Погоджуючись на апаратну ідентифікацію, ми жертвуємо власною анонімністю, а також отримуємо багато проблем. Наприклад, «захищені» комп'ютери відмовляться працювати з піратським програмним і запускати мультимедійні файли, які не забезпечені міткою DRM. З такою мікросхемою потрібно ліцензувати кожну програму і кожен файл, з яким передбачається працювати - а комп'ютер буде пильно за цим стежити.

Таким чином, з приведеного матеріалу видно, що апаратному забезпеченню приділяється велика увага.

1.3 Постановка задачі по реалізації модулярного експоненціонування

Для сучасних криптографічних систем захисту інформації сформульовані наступні загальноприйняті вимоги [7, 8]:

– зашифроване повідомлення повинне піддаватися читанню тільки при наявності ключа;

– число операції, необхідних для визначення використаного ключа шифрування по фрагменту шифрованого повідомлення й відповідного йому відкритого тексту, повинне бути не менше загального числа можливих ключів;

– число операцій, необхідних для розшифровування інформації шляхом перебору всіляких ключів, повинне мати строгу нижню оцінку й виходити за межі можливостей сучасних комп'ютерів (з урахуванням можливості використання мережних обчислень) або вимагати неприйнятно високих витрат на ці обчислення;

– знання алгоритму шифрування не повинне впливати на надійність захисту;

– незначна зміна ключа повинна приводити до істотної зміни виду зашифрованого повідомлення навіть при шифруванні того самого вихідного тексту;

– незначна зміна вихідного тексту повинне приводити до істотної зміни виду зашифрованого повідомлення навіть при використанні того самого ключа;

– структурні елементи алгоритму шифрування повинні бути незмінними;

– додаткові біти, що уводяться в повідомлення в процесі шифрування, повинні бути повністю й надійно сховані в шифрованому тексті;

– довжина шифрованого тексту не повинна перевершувати довжину вихідного тексту;

– не повинне бути простих і легко встановлюваних залежностей між ключами, послідовно використовуваними в процесі шифрування;

– будь-який ключ із множини можливих повинен забезпечувати надійний захист інформації;

– алгоритм повинен допускати як програмну, так і апаратну реалізацію, при цьому зміна довжини ключа не повинне вести до якісного погіршення алгоритму шифрування.

Аналіз предметної області, виконаний в розділах 1.1, 1.2 показав актуальність апаратної реалізації криптоалгоритмів та їх частин, тому в наступних розділах потрібно вирішити наступні завдання:

– розглянути алгоритм модулярного експоненціонування як механізму визначення таємного ключа;

– здійснити декомпозицію даного алгоритму на елементарні операції;

– передбачити апаратну реалізації елементарних операцій;

– розробити структурну та функційну схеми апаратної реалізації модулярного експоненціонування;

– розробити електричні схеми блоків структурної схеми та здійснити їх налаштування.

Для вирішення даних задач пропонується використати програмне забезпечення NI Multisim.

2. ПРОЕКТУВАННЯ СИСТЕМИ ЗАХИСТУ ІНФОРМАЦІЇ

2.1 Особливості алгоритмів модулярного експоненціонування

Визначальним напрямком розвитку інформаційних технологій є інформаційна інтеграція на основі комп'ютерних мереж. Ефективне використання таких технологій потребує надійного захисту даних та розділення прав доступу в інтегрованому інформаційному середовищі. На сьогоднішній день такий захист та розподілення прав доступу забезпечується застосування ряду спеціальних протоколів мережевого обміну, в основі більшості з яких лежить криптографія відкритих ключів. Базовою обчислювальною операцією цих криптографічних перетворень є модулярне експоненціювання, тобто обчислення $A^E \bmod M$ [8, 9]. Розрядність чисел, що регламентується існуючими протоколами значно більша за розрядність процесора і становить 2048 або 4096 [1]. Фактично швидкість виконання цієї операції визначає час реалізації протоколів криптографічного захисту інформації в мережах.

Обчислювальну складність алгоритмів модулярного експоненціювання при їх реалізації на різних обчислювальних платформах зазвичай оцінють кількістю використаних в них опреацій процесорного множення. Процесорними називають цілочисельні операції, які виконуються над k-розрядними числами, довжина яких відповідає розрядності процесора. Процес модулярного експоненціювання зводиться до послідовного виконання $\log 2^E = n$ циклів, у кожному із яких виконується операція піднесення до квадрату результату операції попереднього циклу й залежно від поточного біта степені Е, виконується операція множення. Залежно від порядку, в якому аналізуються розряди степені Е можна розглянути 2 типи алгоритмів експоненціювання [3]:

1) алгоритми, які передбачають аналіз розрядів степені E, починаючи зі старших розрядів (зліва-направо). У нотаціях мови C++ алгоритм цього типу може бути представлений у вигляді:

1. R = 1.
2. for (j=n-1; j >=0; j -) { 2.1.
R = R·R mod M 2.2.
if (ej == 1) R = R·A mod M }
3. Результат: R.

При цьому, під час кожної ітерації циклу виконується модулярне піднесення числа в квадрат і множення на постійне число, рівне A, що створює потенційні передумови для підвищення швидкості множення. Недоліком є те, що оперції виконуються строго послідовно й належать критичному шляху. Це не дозволяє реалізувати паралельне обчислення операцій.

2) Алгоритми, які передбачають аналіз розрядів степені E починаючи із молодших (справа наліво). З використанням мови C++, алгоритм модулярного експоненціювання цього типу може бути представлений у вигляді:

1. $R = 1, Q = 1.$
2. for (j=0; j <n; j ++)
{
2.1. $R = R \cdot R \bmod M$
2.2. if ($e_j == 1$)
$Q = Q \cdot R$
}
3. Результат: Q.

При такій реалізації алгоритму є потенціальна можливість розпаралелити модулярне експоненціювання. Аналіз вказаного алгоритму показує, що

базовими операціями виконання модулярного експоненціювання є модулярне піднесення до квадрату і модулярне множення на фіксоване число, час виконання яких фактично визначається продуктивністю обчислення $A^E \text{mod} M$.

Більшість алгоритмів модулярного експоненціювання для реалізації згаданих двох операцій використовують єдину операцію модулярного множення. У свою чергу, час виконання модулярного множення визначається двома складовими: часом, що необхідний для реалізації власне множення і часом, який витрачається на модулярну редукцію. У класичному множенні модулярна редукція реалізується з використанням операції ділення і, відповідно, друга складова відіграє значну роль. Значна ефективність обчислювальної реалізації модулярного множення досягається при використанні алгоритму Монтгомері, в якому модулярна редукція зводиться до здвигу на k розрядів. Тому, на практиці, при виконанні модулярного експоненціювання у більшості використовується алгоритм Монтгомері. Позначимо як Mont(A,B) множення Монтгомері, яке формує результат

$$R = A \cdot B \cdot U \bmod M,$$

де U модулярна інверсія числа 2^n по модулю M, тобто $U = (2^n)^{-1} \text{mod } M$. Алгоритм модулярного експоненціювання Монтгомері можна представити з використання введених нотацій наступним чином:

1. $\tilde{x} = Mont(X, U^2 \bmod M) =$
   $X \cdot U \bmod M, A = U \bmod M$
2. for ($j = n; j >= 0; j$ --)
   {
   2.1. A = Mont(A,A);
   2.2. if ($e_j = 1$) $A \leftarrow Mont(A, \tilde{x})$
   }
3. $A \leftarrow$Mont(A,1).

Аналіз розглянутих алгоритмів модулярного експоненціонування показав, що найбільш простим для апаратної реалізації є алгоритм пониження степеня числа $K=A^E \bmod M$. Блок-схема даного алгоритму приведена на рисунку 2.1.

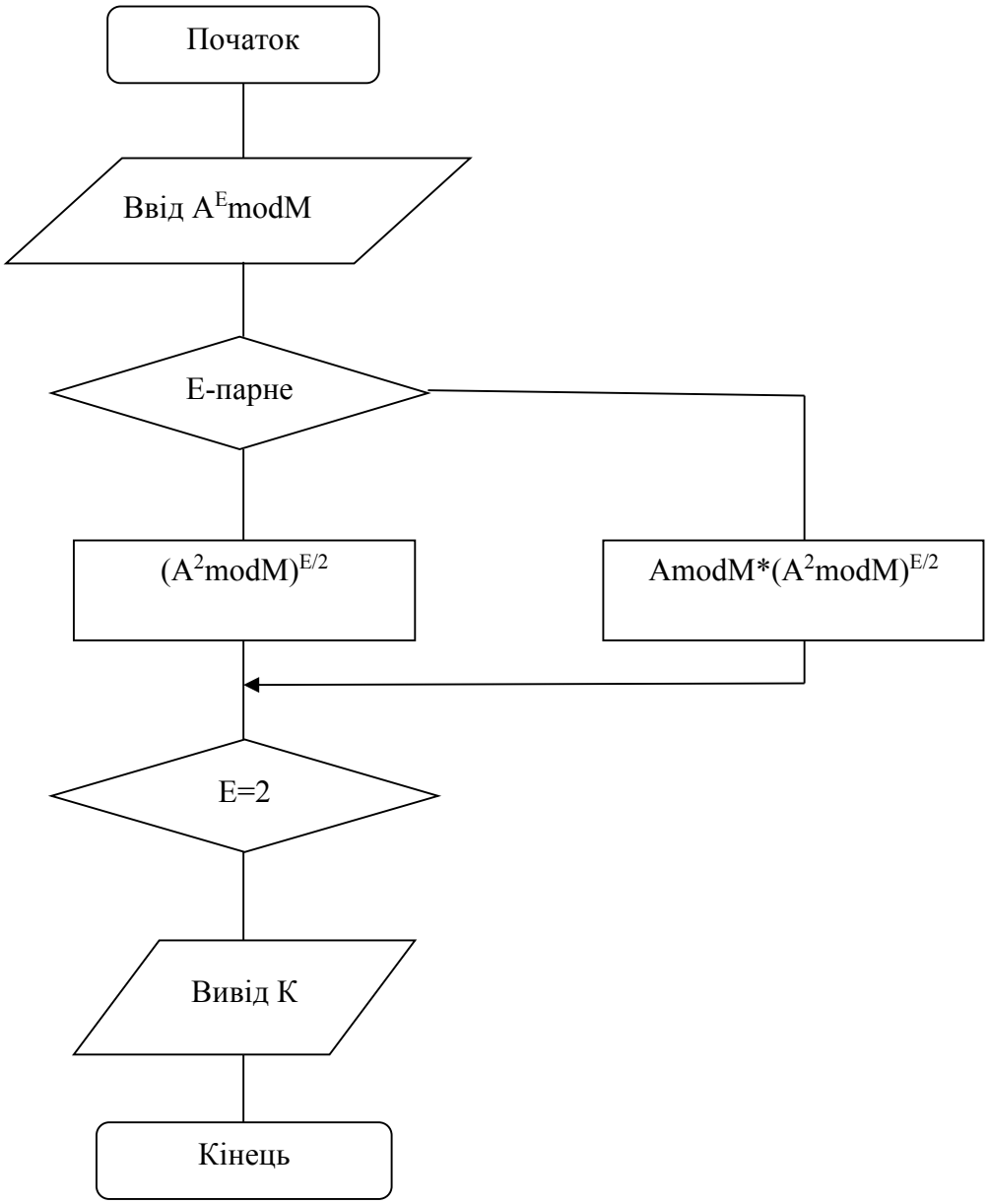

Рисунок 2.1 – Блок-схема алгоритму визначення K шляхом пониження степеня числа $A^E \bmod M$

Таким чином, в даному розділі здійснено аналіз методів модулярного експоненціонування та запропоновано використання методу пониження степеня.

2.2 Проектування структури апаратного модуля модулярного експоненціонування

Розробка структурної схеми апаратного модуля реалізації операції модулярного експоненціонування згідно постановки задачі, здійсненої в розділі 1.3 передбачає реалізацію алгоритму пониження степеня, виділення елементарних операцій, пошук схем апаратної реалізації та синтез на цій основі структурної схеми. Структурна схема апаратної реалізації модулярного експоненціонування приведена на рисунку 2.2 [10].

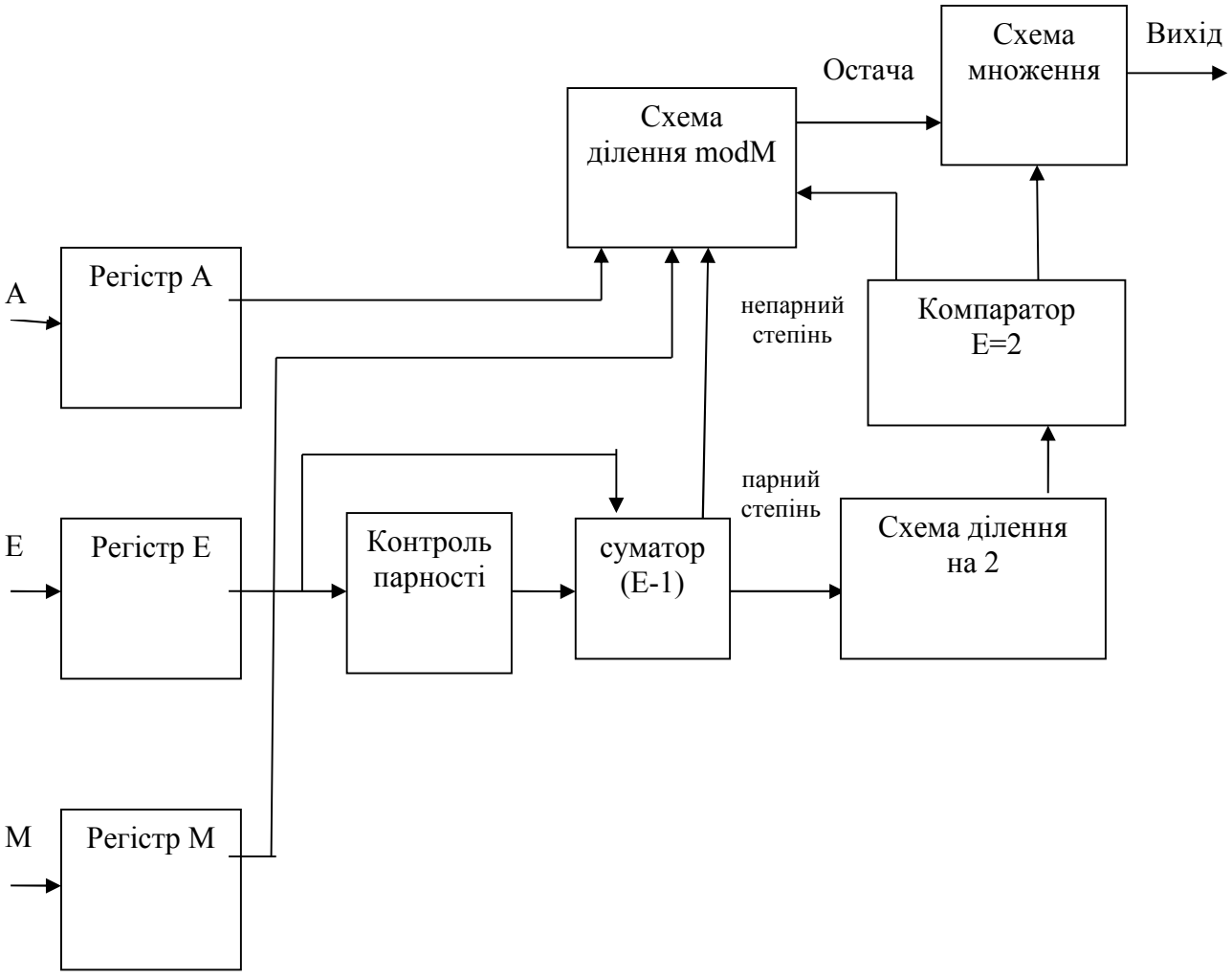

Рисунок 2.2 – Структурна схема засобу модулярного експоненціювання

Аналіз блок-схеми алгоритму модулярного експоненціонування приведений в розділі 2.1 та структурної схеми на рисунку 2.2 дозволив виділити наступні типові операції обробки:

- множення;
- ділення;
- порівняння двійкових кодів;
- визначення парності чисел.

Наступним кроком буде розробка структурних схем компонентів, які реалізують операцію модулярного експоненціювання.

Множення в двійковому вигляді проводиться подібно множенню в десятковій системі числення. При цьому потрібно перемножити кожен розряд множеного на відповідний розряд множника, а потім підсумувати отримані часткові добутки.

Розглянемо як приклад множення двох четирирозрядних двійкових чисел. Нехай потрібно помножити число 1011_2 (11_{10}) на число 1101_2 (13_{10}). В результаті множення очікуємо отримати число 10001111_2 (143_{10}). Контрольний приклад виконання операції множення в стовпчик приведено нижче по тексту.

$$\begin{array}{r} \times\ 1101 \\ 1011 \\ \hline 1101 \\ +\ 1101 \\ +\ 0000 \\ +\ 1101 \\ \hline 10001111 \end{array} \qquad (2.1)$$

Для формування добутку потрібно обчислити чотири часткових добутки. Зверніть увагу, що в двійковій арифметиці потрібно виконувати множення тільки на числа 0 і 1. Це означає, що потрібно або підсумовувати множене до суми інших часткових добутків, або ні. Таким чином, для

формування часткового добутку можна скористатися логічними елементами "2И". Структурна схема пристрою множення двійкових кодів приведена на рисунку 2.3.

Для формування часткового добутку, крім операції множення на один розряд, потрібно здійснювати його зсув вліво на число розрядів, відповідне вазі розряду множника. Зсув можна здійснити простим з'єднанням відповідних розрядів часткових добутків до необхідних розрядів двійкового суматора [12 - 15].

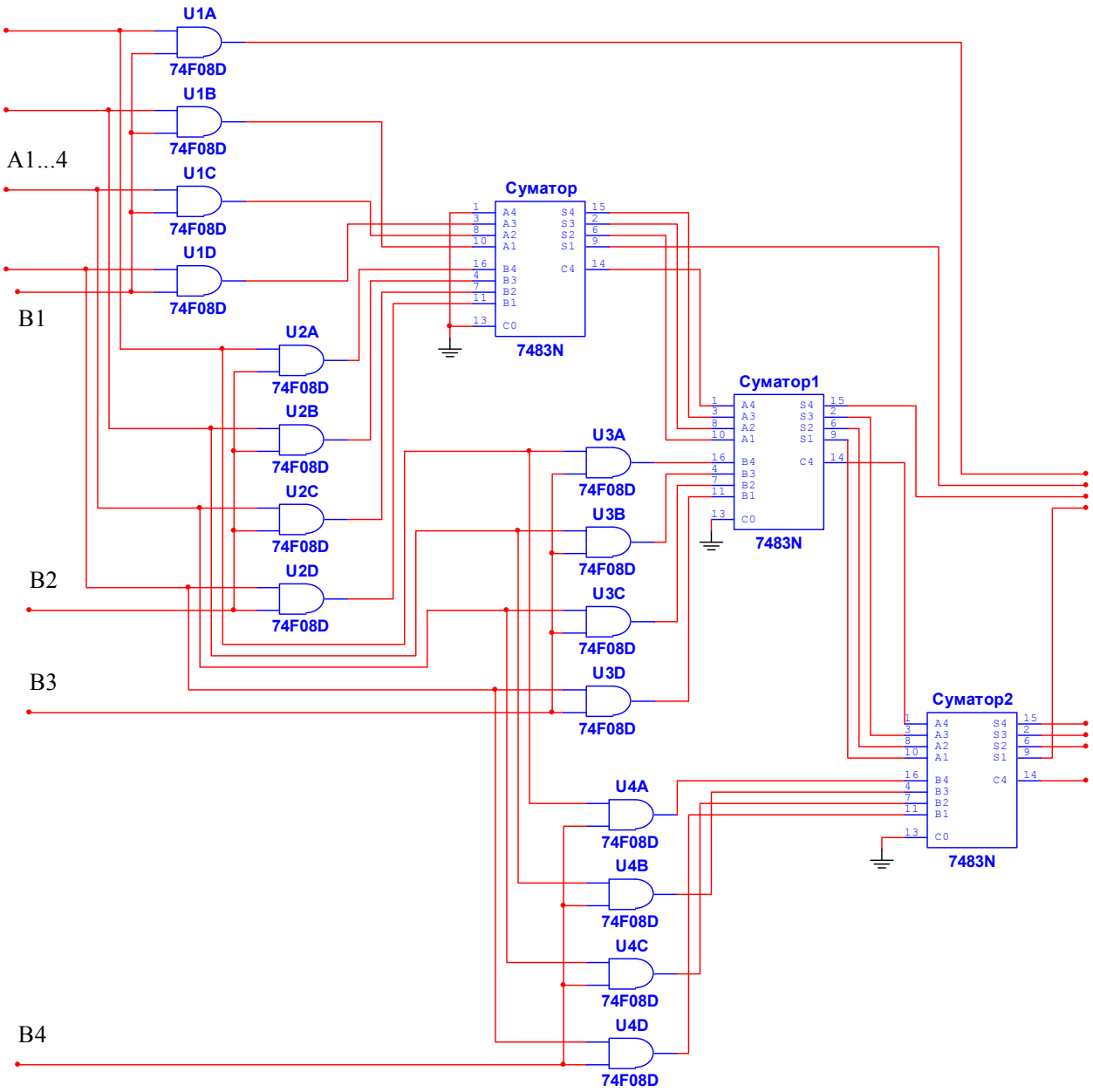

Рисунок 2.3 – Структурна схема чотирирозрядного пристрою множення

У відповідності з алгоритмом множення в стовпчик нам будуть потрібні три чотирирозрядних суматора. Структурна схема помножувача, що реалізує алгоритм двійкового множення в стовпчик передбачає, що формування часткових добутків в цій схемі здійснюють мікросхеми логіки. У цих мікросхемах міститься відразу по чотири логічних елемента "2И".

Суматор, виконаний на мікросхемі, підсумовує перший і другий частковий добутки. При цьому молодший розряд першого часткового добутку не потребує підсумовуванні. Тому він подається на вихід помножувача безпосередньо (молодший розряд). Другий частковий добуток має бути зсунутий на один розряд. Це здійснюється тим, що молодший розряд вихідного числа першого суматора з'єднується з другим розрядом добутку (наступний розряд). Але тоді перший частковий добуток необхідно зсунути на один розряд по відношенню до другого часткового добутку. Це виконується тим, що молодший розряд групи входів А з'єднується з першим розрядом часткового добутку, перший розряд групи входів А з'єднується з другим розрядом часткового добутку, і так дальше по розрядам. Точно таким же чином здійснюється підсумовування третього і четвертого часткового добутку. Це підсумовування виконують мікросхеми другого та третього суматорів відповідно. Відмінність полягає лише в тому, що тут не потрібно замислюватися про старший розряд попередньої суми, адже попередня мікросхема суматора формує сигнал перенесення.

Таким чином схема помножувача утворює матрицю, сформовану провідниками, по яких передаються розряди числа А і числа В. У точках перетину цих провідників знаходяться логічні елементи "2И". Саме з цієї причини помножувачі, реалізовані за даною схемою, отримали назву матричних помножувачів [16 - 18].

Швидкість роботи схеми, наведеної на рисунку 2.3, визначається максимальним часом поширення сигналу. Це ланцюг мікросхем. Час роботи схеми можна скоротити, якщо суматори розташовувати не послідовно один за одним, як це передбачається алгоритмом, а підсумовувати часткові

добутки попарно, потім підсумовувати пари часткових добутків. В цьому випадку час виконання операції множення значно скоротиться.

Наступним кроком є розробка структурної схеми для операції ділення двійкових кодів. Структурна схема пристрою ділення приведена на рисунку 2.4. Основним елементом його є віднімаючий пристрій побудований на паралельних суматорах. Мультиплексор є перемикачем з двох на один вихід. При подачі імпульсу «Завантаження» мультиплексор MS включається в положення, коли на його вихід надходить ділене A1. З виходу мультиплексор ділене поступає на вхід A схеми віднімання. На другий вхід мультиплексора надходить різниця з регістра зберігання 2. Одночасно в регістр 1 по старшим розрядам записується дільник A2, який надходить на входи B схеми віднімання [19, 20].

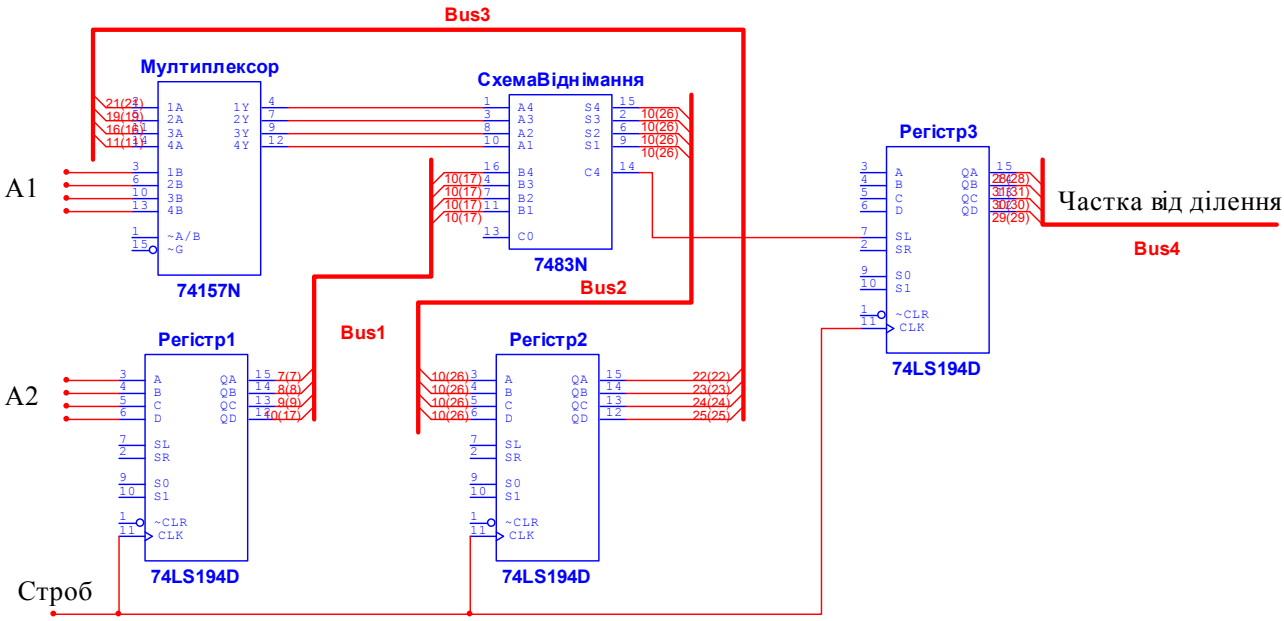

Рисунок 2.4 – Структурна схема пристрою ділення

На виході схеми віднімання утворюється різниця. Якщо ця різниця більше нуля, то на виході C4 знаходиться "1", яка по тактовому імпульсу заноситься в регістр RG3. Якщо є переповнення (різниця менше нуля), то на виході P знаходиться "0", який і заноситься в регістр RG3. З кожним тактом

роботи послідовно зсуваються розряди дільника А2 на один розряд вниз в регістрі RG1, а результат на один розряд вгору в регістрі RG3. Число тактів роботи визначається розрядністю результату, але при цьому потрібно пам'ятати, що розрядність регістрів RG1, RG2 і схеми віднімання. визначається як сума числа розрядів діленого і числа розрядів результату.

Наступною операцією, що використовується при реалізації модулярного експоненціювання є пристрій контролю парності.

Промисловістю випускаються схеми контролю парності в інтегральному виконанні – наприклад мікросхема К155ИП2. Вона обробляє восьмибітові слова і має входи розширення, що дозволяють нарощувати розрядність. Аналогічні функції виконує мікросхема К1533ИП5, що працює з дев'ятирозрядними числами. Принцип побудови такої схеми приведено на рисунку 2.5 [19].

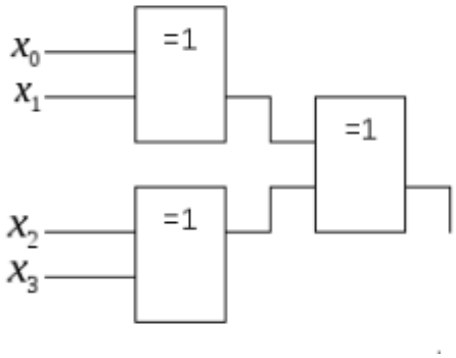

Рисунок 2.5 – Схема пристрою контролю парності

В цілому, мікросхеми, що виконують роль цифрового компаратора мають входи для двійкових чисел А0 ... А3 і В0 ... В3. Входи A>B, A=B і A<B використовуються для нарощування розрядності схеми порівняння кодів. Мікросхема цифрового компаратора має три виходи результатів аналізу: A>B, A=B і A<B. Принцип дії заснований на наступному: в момент рівності вхідних сигналів вихідний сигнал компаратора змінюється стрибком. При цьому формується перепад від одного рівня вихідної напруги до іншого.

Критерієм рівності чисел є збіг їх в усіх розрядах. На виході компаратора встановлюється одиниця, якщо обидва числа рівні, і 0 - в іншому випадку.

Таким чином, в даному розділі розроблено структурну схему модулярного екпоненціонування та приведені структури пристроїв за допомогою яких можна здійснити його апаратну реалізацію.

2.3 Обґрунтування вибору засобу реалізації та компонентної бази

Апаратну реалізацію модуля модулярного експоненціонування пропонується здійснити в імітаційному пакеті NI Multisim.

NI Multisim - це емулятор електронних схем, який дозволяє розробку цифрових пристроїв за мінімальний час. Він включає в себе ПП Multicap, що дозволяє здійснювати програмний опис та тестування розроблених схем. Цей процес реалізується на основі технології віртуальних електронних приладів. NI Multisim використовується для інтерактивного SPICE-моделювання та аналізу електричних схем, проектуванні друкованих плат і комплексному тестуванні. Ця платформа пов'язує процеси проектування та тестування, надаючи розробнику гнучкі можливості віртуальних приладів. Це дозволяє порівнювати теоретичні дані з реальними безпосередньо в процесі створення схем друкованих плат, що знижує кількість ітерацій, число помилок в макетах і прискорює вихід розробки на ринок.

Інтерфейс користувача, характерний для NI Multisim, представлений на рисунку 3.1.

Типова структура містить головне меню, піктограми швидкого доступу, панелі компонентів та службові вікна, що надають розширені можливості для моделювання. Панель компонентів містить SPICE-моделі електронних пристроїв, що використовуються для побудови принципових електронних схем.

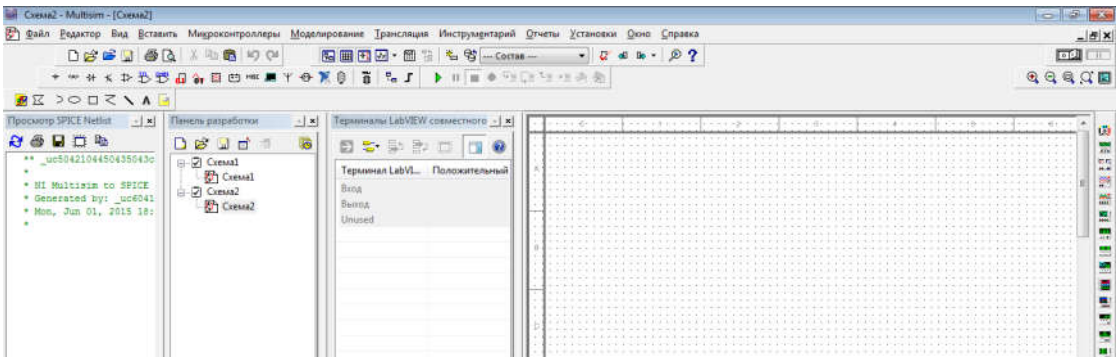

Рисунок 2.7 – Середовище NI Multisim

Особливістю, що сприяє широкому використанню NI Multisim є велика база даних SPICE-моделі електронних пристроїв, що можна використовувати для розробки електричних схем. Тому завданням даного розділу буде вибір компонентів.

Аналіз матеріалів попередніх розділів показав, що основними елементами структурних схем є мікросхеми:

– комп'ютерної логіки;
– суматори;
– мультиплексори;
– регістри;
– лічильники.

Пошук в базі даних NI Multisim показав, що комп'ютерна логіка та інші види мікросхем представлені широкою номенклатурою мікросхем, що дає можливість практичної реалізації електричних схем для створення засобу модулярного експоненціювання.

3 РОЗРОБКА ТА ВЕРИФІКАЦІЯ АПАРАТНОГО ЗАСОБУ МОДУЛЯРНОГО ЕКСПОНЕНЦІЮВАННЯ

3.1 Розробка та налаштування схеми множення

Основою для розробки одного з основних компонентів апаратного засобу модулярного експоненціювання є структурна схема пристрою множення двійкових кодів на основі матричної структури. Згідно структурної схеми пристрою множення побудуємо та налаштуємо функційну схему отримання першого часткового добутку. Для цього використаємо контрольний приклад за формулою 2.1, приведений в другому розділі. Згідно даного прикладу код 1101 множиться на одиницю молодшого розряду 1. Звідси перший частковий добуток буде рівний 1101. Це значення і буде використовуватися при налаштуванні функційної електричної схеми пристрою множення. В середовищі NI Multisim змоделюємо електричну схему формування першого часткового добутку. Основними елементами даної схеми будуть мікросхеми логічних елементів, що реалізують операцію кон'юнкції та двійкового суматора. В якості двійкового суматора з бази даних SPICE-моделей NI Multisim вибрано схему 7483N, умовно-графічне позначення якої приведено на рисунку 3.1 [18].

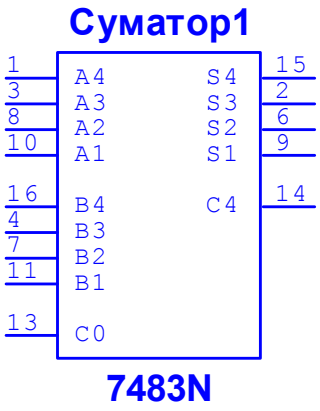

Рисунок 3.1 – Умовно-графічне позначення схеми чотири розрядного суматора 7483N

В даній схемі на входи A1-A4 та B1-B4 подаються двійкові коди чисел, що додаються. Вхід C0 є входом переносу з молодших розрядів, а C4 виходом переносу в старші розряди. Дані входи/виходи використовуються при масштабуванні схем на основі суматорів. Для формування вхідних сигналів та візуалізації результатів використаємо константу цифрових сигналів та LED-індикатор.

Фрагмент схеми, що реалізує операцію додавання кодів 1101 та 1001 приведено на рисунку 3.2. Результат додавання, тобто значення суми рівне 10110 ілюструється LED – індикатором.

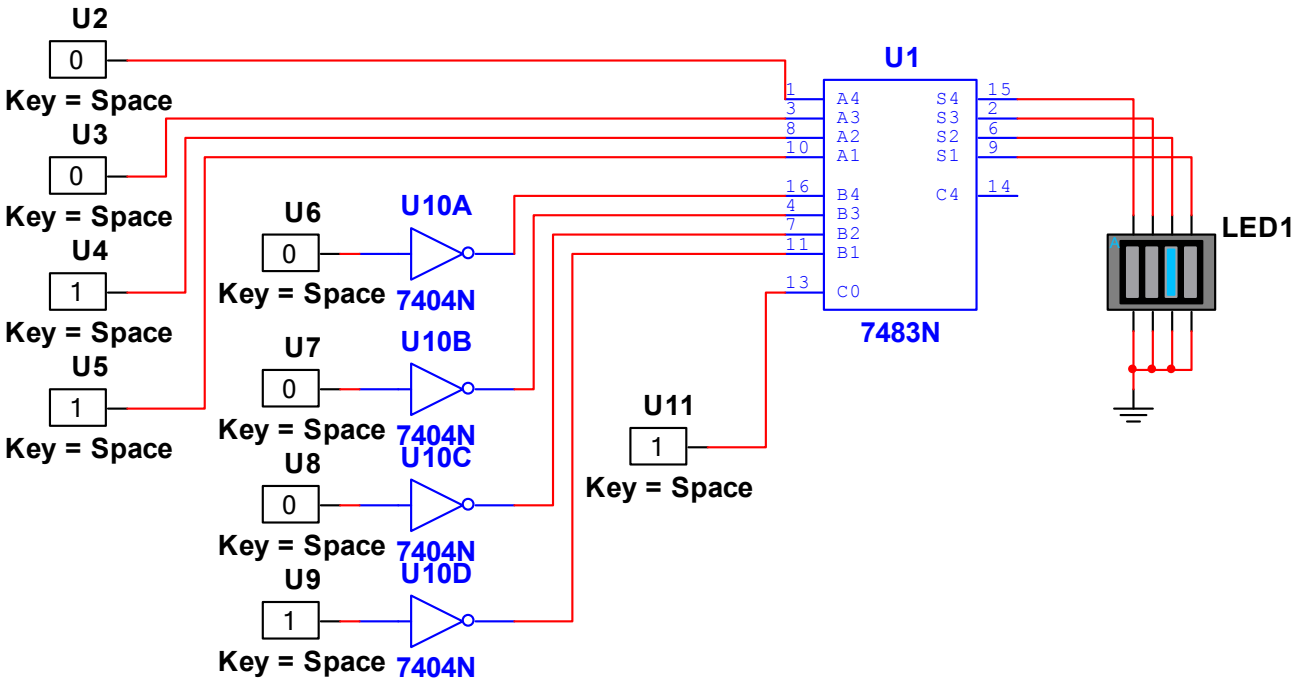

Рисунок 3.2 – Схема віднімання двійкових чисел

Аналогічним чином здійснюється визначення другого множника (q-1). При цьому в якості вимірювальних приладів використовувалися LED-індикатори.

Реалізацію функції Ейлера пропонується реалізувати із застосуванням схеми матричного помножувача [20].

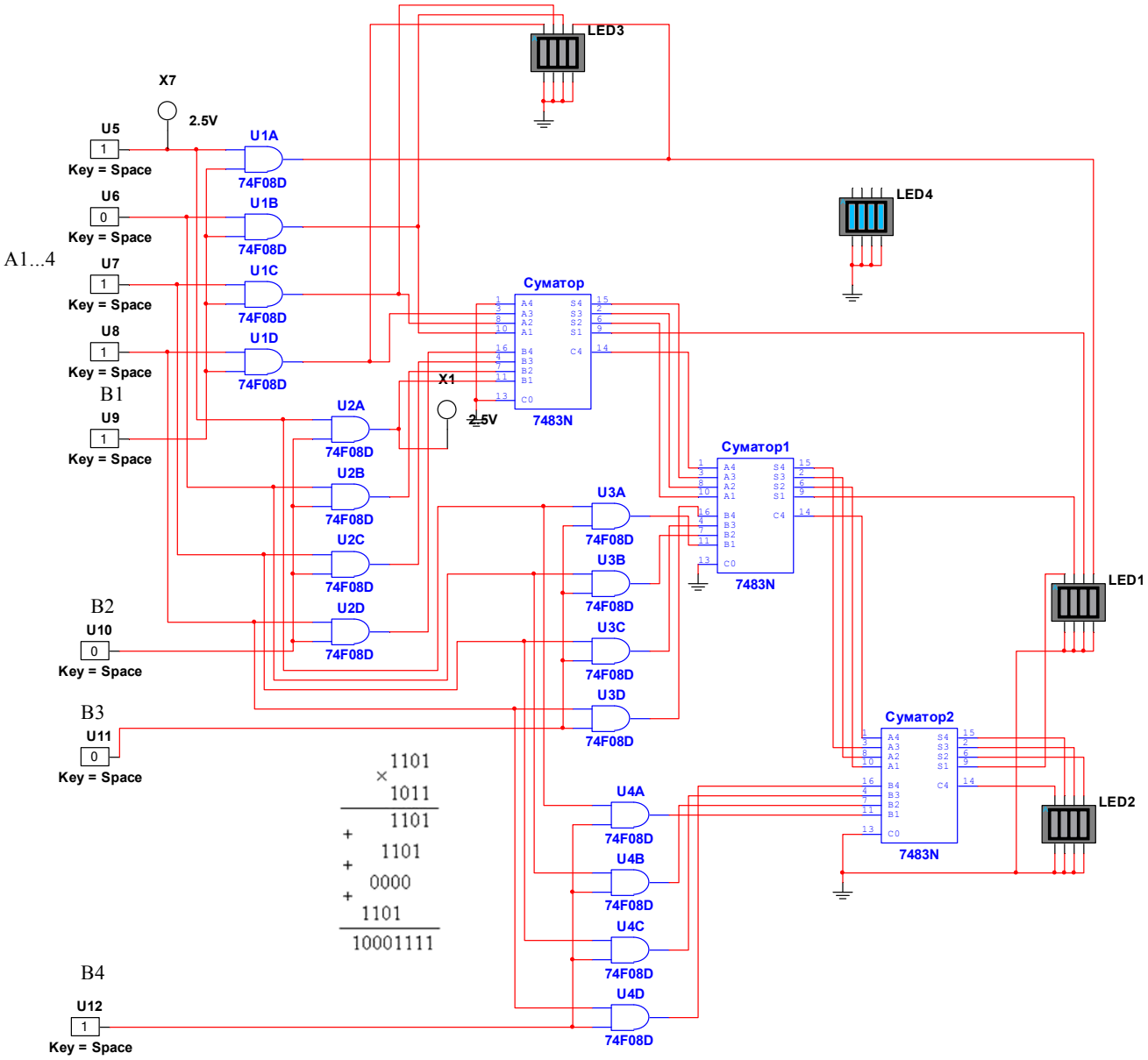

Рисунок 3.3 – Схема налаштування пристрою множення двійкових чисел

Таким чином в даному розділі здійснено розробку на налаштування пристрою множення двійкових чисел.

3.2 Розробка та налаштування схеми ділення

Розробку та налаштування пристрою, що апаратно реалізує операцію ділення двійкових чисел здійснюємо у відповідності з розробленою на

рисунку 2.4 структурною схемою. Основними елементами даної структури виступають схеми мультиплексора, регістрів, суматора. В якості мультиплексор з бази даних NI Multisim вибрано мікросхему 74LS194D – рисунок 3.4 [22-24].

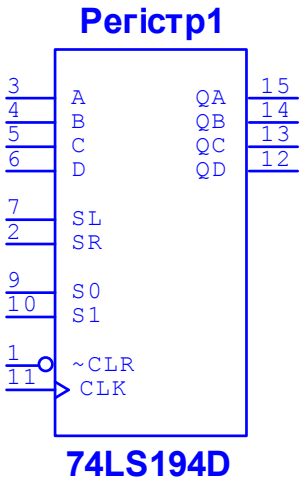

Рисунок 3.4 – Умовно-графічне позначення регістра 74LS194D

Мікросхема 74194 містить реверсивний 4-розрядний регістр зсуву даних з паралельним і послідовним введенням-виведенням інформації, а також з входом скидання. Робота схеми відбувається наступним чином. Якщо на вхід скидання Clear мікросхеми 74194 надходить напруга низького рівня, то на всіх виходах Q0 - Q3 встановлюється напруга низького рівня незалежно від логічного стану всіх інших входів.

Якщо на вхід скидання Clear мікросхеми 74194 подається напруга високого рівня, то режим роботи визначається станами входів S0 і S1. Зсув даних вліво відбувається, коли низька напруга подається на входи S0 і S1. При цьому дані послідовно надходять на вхід DA.

Зсув даних вправо відбувається в мікросхемі 74194, коли на вхід SR подається напруга високого, а на вхід SL - низького рівня, при цьому дані послідовно надходять на вхід DA. Дослідження проводилося згідно рисунку 3.5.

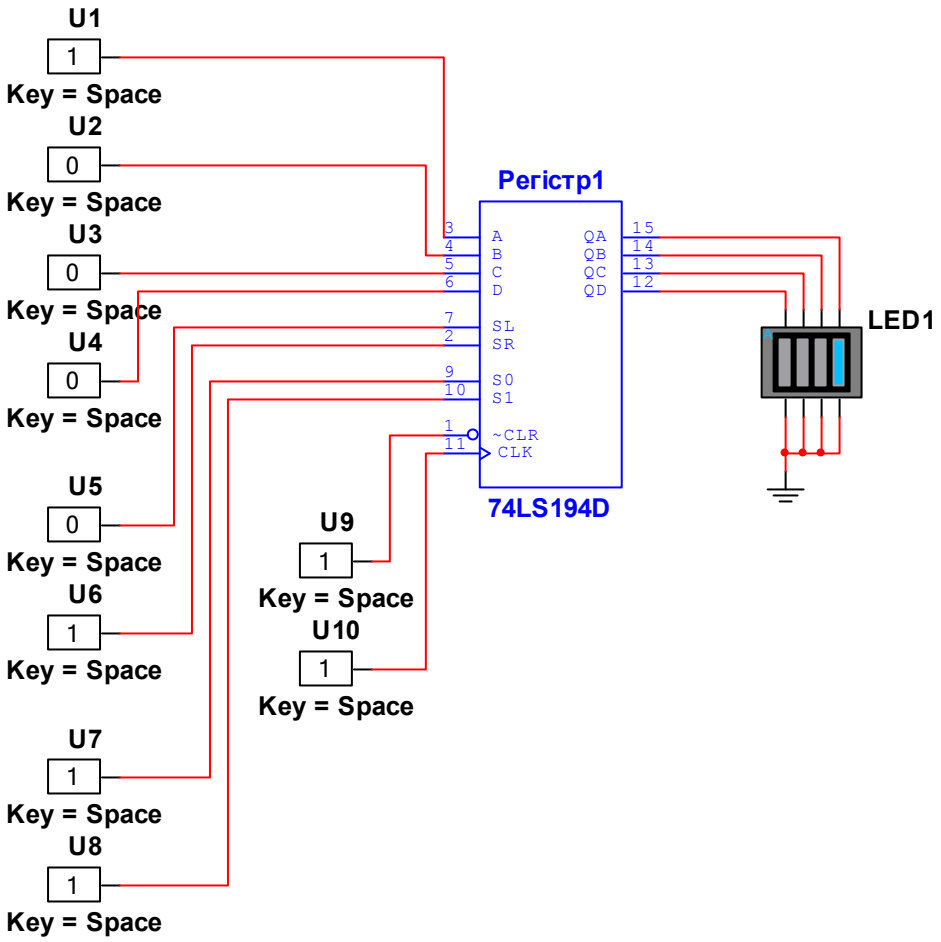

Рисунок 3.5 – Дослідження роботи регістру 4LS194D

Якщо на обидва входи S0 і S1 мікросхеми 74194 подається напруга високого рівня, то можлива послідовна завантаження даних з входів P0 - P3. Під час паралельного введення даних послідовні входи замкнені.

Послідовно чи паралельно введені дані надходять в регістр зсуву синхронно з перепадом напруги тактового імпульсу на вході Clock (вхід тактових імпульсів мікросхеми 74194) з низького рівня на високий (позитивний фронт). При цьому дані повинні перебувати на інформаційних входах до початку формування позитивного фронту тактового імпульсу.

Напруга низького рівня на входах S0 і S1 мікросхеми 74194 замикає вхід тактових імпульсів. Логічний стан на обох цих входах міняється лише тоді, коли на вхід тактових імпульсів надходить напруга високого рівня.

Результати дослідження режимів роботи регістру приведені в таблиці 3.1.

Таблиця 3.1 – Таблиця станів регістру 4LS194D

Строб Clok	Режими роботи		Скидання Clear	Функція
	S0	S1		
X	0	0	1	Немає змін
∫	1	0	1	Зсув вправо (Q0>Q3)
∫	0	1	1	Зсув вліво (Q3>Q0)
∫	1	1	1	Паралельна загрузка
X	X	X	0	Скидання

Регістр зсуву застосовується як накопичувач даних, перетворювач послідовного коду в паралельний і паралельного коду в послідовний.

Наступним важливим елементом, який застосовується в схемі ділення є мультиплексор 74157N. Схема дослідження його роботи приведена на рисунку 3.6.

Мікросхема 74157 містить чотири селектора даних з двома входами і одним виходом. За допомогою мікросхеми 74157 можна з даних, що надходять на чотири пари входів 1A/1B - 4A/4B, вибрати необхідну інформацію і передати її на один з відповідних виходів 1Q - 4Q. Якщо на дозволяючий вхід Enable мікросхеми 74157 подається напруга високого рівня, то на всіх виходах встановлюється напруга низького рівня незалежно від стану інших входів. Якщо на цей вхід подається напруга низького рівня, то стан виходів залежить від стану входу вибірки Select.

Якщо на вхід вибірки A/B мікросхеми 74157 подається напруга низького рівня. то виходи приймають той же рівень напруги, який надходить на входи A.

Якщо ж на вхід вибірки подасться напруга високого рівня, то виходи приймають той же логічний рівень, який надходить на входи B.

Важливим елементом схеми ділення є операція віднімання. Її пропонується реалізувати з використанням доповнюючого коду від'ємника - рисунку 3.7.

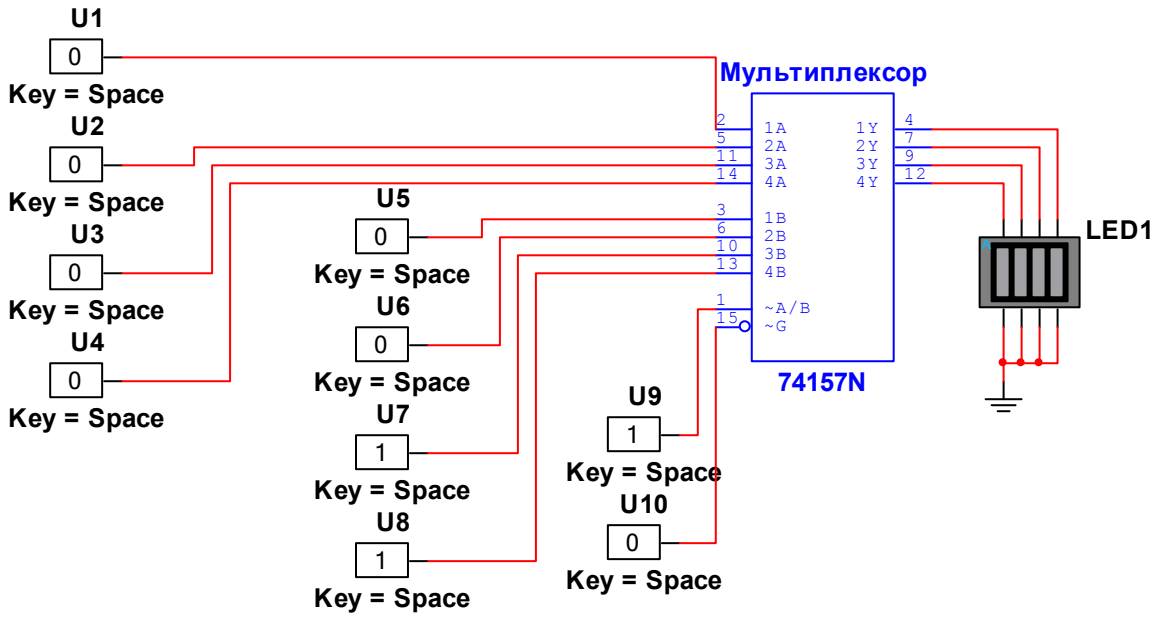

Рисунок 3.6 – Дослідження роботи мультиплексора 74157N

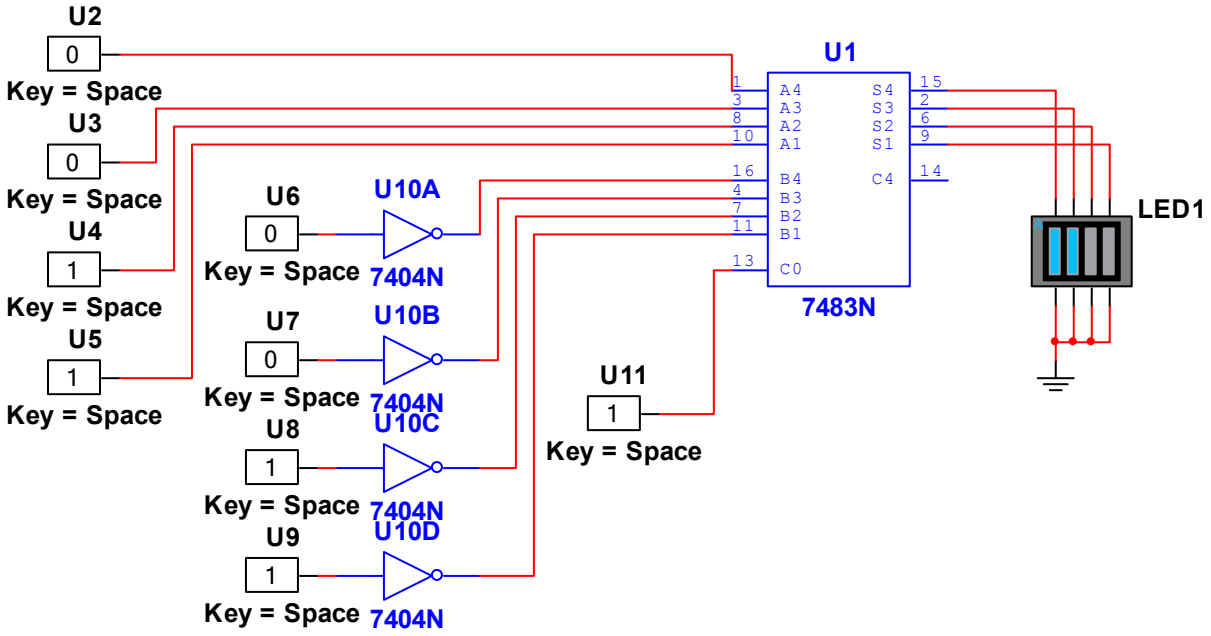

Рисунок 3.7 – Налаштування схеми віднімання

Якщо на вхід чотири розрядної схеми віднімання подати зменшуване 15 (1111) та від'ємник 3 (0011), то результат повинен дорівнювати 12 (1100). Цей результат ілюструється на елемента U2-U9 та ілюструється індикатором LED1.

Дослідження роботи елементів та фрагментів структурної схеми пристрою ділення двійкових чисел послужили основою для налаштування

функційної електричної схеми приведеної на рисунку 3.8. Наявність індикаторів LEDN дозволяє від слідкувати проходження сигналів по схемі.

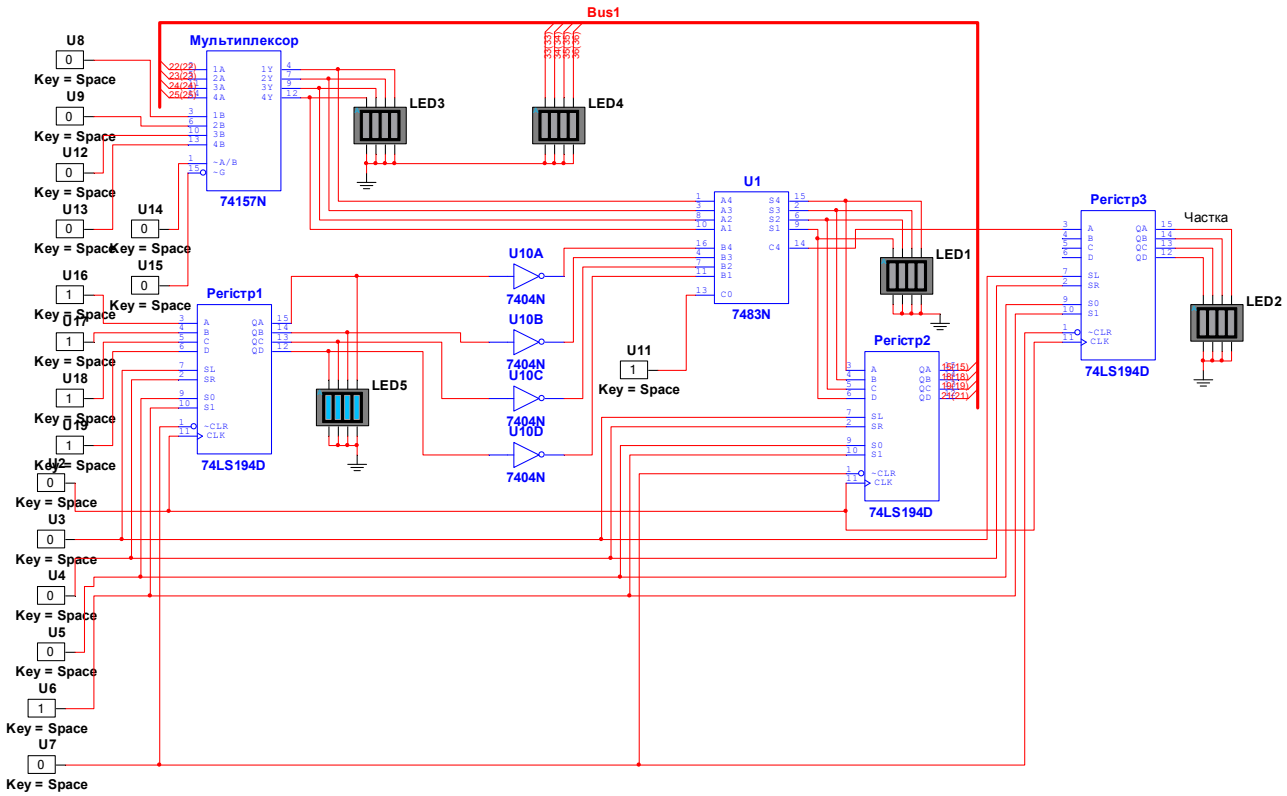

Рисунок 3.8 – Функційна схема налаштування пристрою ділення двійкових чисел

Таким чином, в даному розділі розроблено та налаштовано схему ділення двійкових чисел. Передбачається, що дана схема буде використана для розробки функційної електричної схеми апаратного засобу модулярного експоненціювання.

3.3 Розробка схеми апаратного засобу модулярного експоненціювання

При розробці електричної схеми апаратного засобу модулярного експоненціювання слід взяти до уваги те, що застосування NI Multisim має

певні обмеження, які стосуються складності розробки моделей електричних схем. Разом з тим, апаратна реалізація цифрових пристроїв на макетних платах є дуже громіздкою та вимагає значних матеріальних затрат на виготовлення та налаштування. Тому завданням даного розділу буде розробка функційної електричної схеми апаратного засобу модулярного експоненціювання, що відповідає структурній схемі приведеній на рисунку 2.2.

Крім розглянутих вище пристроїв множення та ділення в структурній схемі апаратного засобу модулярного експоненціювання використовуються пристрої контролю парності та цифрові компаратори. Схеми контролю парності випускаються в інтегральному виконанні, наприклад 74180D.

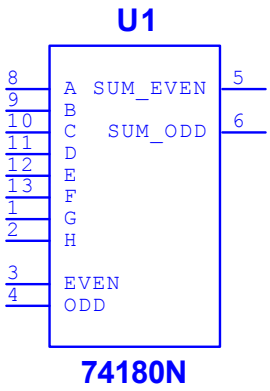

Рисунок 3.9 – Умовно-графічне позначення схеми контролю парності 74180D

Дана мікросхема видає сигнал контролю парності SUM_EVEN (5 pin) або непарності SUM_ODD (6pin). Входи EVEN та ODD служать в цілях нарощування розрядності.

В структурній схемі апаратного засобу модулярного експоненціювання використовуються цифрові компаратори. Цифровий компаратор видає три сигнали на виході: A>B, A<B та A=B в залежності від значення вхідних чисел. В базі даних NI Multisim наявні мікросхеми цифрових компараторів, зокрема 4-и розрядний компаратор 7485N.

Мікросхема 7485 порівнює два 4-розрядних слова і визначає співвідношення між ними. Обидва порівнюваних слова A і B надходять на відповідні входи мікросхеми 7485. Молодші розряди подаються на входи A1 і B1, а старші - на входи A4 і B4. Якщо необхідно порівняти тільки 4-розрядні слова, то на вхід перенесення мікросхеми 7485 A = B подається напруга високого, а на входи перенесення A> B і A <B - низького рівня. Якщо обидва слова рівні за величиною, на виході A = B формується напруга високого рівня. Якщо слово A більше слова B, на виході A> B формується напруга високого рівня. Якщо слово A менше слова B, на виході A <B встановлюється напруга високого рівня. На інших виходах формується напруга низького рівня.

Коли мікросхема 7485 порівнює 8-розрядні слова, то виходи першого ступеня 4-розрядного компаратора (молодші розряди) з'єднуються з входами перенесення другого ступеня. В цьому випадку результат порівняння отримують на виходах 4-розрядного компаратора старших розрядів.

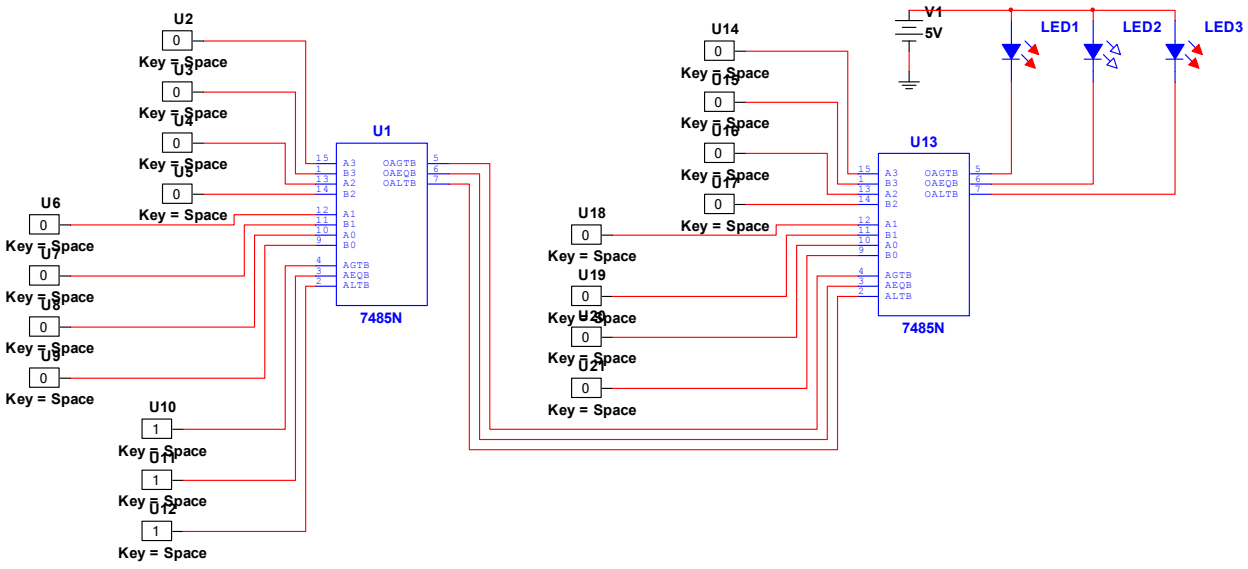

Рисунок 3.10 – Схема дослідження 8-ми розрядного компаратора на 7485D

Розроблену функційну електричну схему засобу модулярного експоненціювання приведено на ДП.КСМ.07174/14.00.00.000СЕ2. Дана схема є синтезом розроблених в третьому розділі функційних електричних схем

множення, ділення, віднімання та інших. Враховуючи велику кількість міжелементних з'єднань в функційній схемі використовуються джгути.

Таким чином, в даному розділі розроблено складові компоненти та функційну схему апаратного засобу модулярного експоненціювання.

4 ТЕХНІКО-ЕКОНОМІЧНИЙ РОЗДІЛ

Метою техніко – економічного розділу дипломного проекту є здійснення економічних розрахунків, спрямованих на визначення економічної доцільності апаратного засобу модулярного сканування. Потрібно визначити доцільність вибраного обладнання, провести розрахунок витрат на розробку даного проектного рішення, визначити прогнозовану ціну апаратного засобу, визначити показники економічної ефективності, зробити відповідні висновки.

4.1 Розрахунок капіталовкладень на розробку драйвера

При загальному підході до розрахунку капіталовкладень, які необхідні на розробку та впровадження апаратного засобу модулярного експоненціонування, можна записати:

$$K = K_{пр} + B_{пр} + B_{м} \quad (4.1)$$

де K – капіталовкладення на створення і впровадження;

$K_{пр}$ – витрати на виконання проектних робіт;

$B_{пр}$ – кошторисна вартість приладів та обладнання проектованого рішення;

Основними факторами при розрахунку витрат на виконання проектних робіт, що впливають на суму є: затрати часу на виконання проекту, необхідна кількість спеціалістів, їхня заробітна плата.

4.1.1 Розрахунок витрат на оплату праці

Витрати на оплату праці включають заробітну плату (ЗП) всіх категорій працівників, безпосередньо зайнятих на всіх етапах проектування. Розмір ЗП обчислюється на основі трудоємності відповідних робіт та середньої ЗП відповідних категорій працівників.

У розробці проектного рішення задіяні наступні спеціалісти - розробники, а саме: керівник проекту; студент-дипломник; консультант техніко-економічного розділу (таблиця 4.1).

Таблиця 4.1 - Вихідні дані для розрахунку витрат на оплату праці

Посада виконавців	Місячний оклад, грн.
Керівник ДП, викладач	6026
Консультант техніко-економічного розділу, доцент	6026
Студент	1100

Витрати на оплату праці розробників проекту визначаються за формулою (4.1):

$$B_{ОП} = \sum_{i=1}^{N} \sum_{j=1}^{M} n_{ij} \cdot t_{ij} \cdot C_{ij}, \quad (4.1)$$

де n_{ij} – чисельність розробників і-ої спеціальності j-го тарифного розряду, осіб;

t_{ij} – затрачений час на розробку проекту співробітником і-ої спеціальності j-го тарифного розряду, год;

C_{ij} – годинна ставка працівника і-ої спеціальності j-го тарифного розряду, грн.,

Годинну ставку працівника можна розрахувати за формулою:

$$C_{ij} = \frac{C_{ij}^{0}(1+h)}{РЧ_{i}}, \quad (4.2)$$

де $С_{ij}$ – основна місячна заробітна плата розробника і-ої спеціальності j-го тарифного розряду, грн.;

h – коефіцієнт, що визначає розмір додаткової заробітної плати (при умові наявності доплат);

$РЧ_i$ - місячний фонд робочого часу працівника і-ої спеціальності j-го тарифного розряду, год. (приймаємо 168 год.).

Коефіцієнт h, який визначає розмір додаткової заробітної плати, для керівника та консультанта техніко-економічного розділу дорівнює 1,47.

Середня годинна ставка керівника та консультанта техніко-економічного розділу ДП дорівнює:

$$C_{ij} = \frac{5470 \cdot (1 + 1{,}47)}{168} = 80{,}42 \frac{грн}{год}.$$

Середня годинна оплата студента дорівнює:

$$C_{ij} = \frac{1200}{168} = 7{,}14 \frac{грн}{год}$$

Витрати на оплату праці складають:

$$B_{ОП} = 20{,}5 \cdot 80{,}42 + 2 \cdot 80{,}42 + 144 \cdot 7{,}14 = 2837{,}45 \; грн.$$

Результати розрахунку записують до таблиці 4.2.

Таблиця 4.2 - Розрахунок витрат на оплату праці

Посада виконавців	Час розробки, год	Погодинна заробітна плата, грн/год.	Витрати на розробку, грн
Керівник ДП, доцент	16	80,42	1648,61
Консультант техніко-економічного розділу, доцент	2	80,42	160,84
Студент	144	7,14	1028
Разом			2837,45

4.1.2 Відрахування на соціальні заходи

Величну відрахувань у спеціальні державні фонди визначають у відсотковому співвідношенні від суми основної та додаткової заробітних плат. Згідно діючого нормативного законодавства єдиний соціальний внесок складає 16,4% від суми заробітної плати:

$$B_{Ф} = 0{,}164 \cdot B_{\text{ОП}}$$

$$B_{Ф} = \frac{16{,}4}{100} \cdot 2837{,}45 = 465{,}34 \ грн.$$

4.1.3 Розрахунок витрат на матеріали та комплектуючі

Загальна сума витрат на матеріальні ресурси (ВМ) визначається за формулою:

$$B_M = \sum_{i=1}^{n} K_i \cdot Ц_i \quad , \qquad (4.3)$$

де Кі - витрата і-го типу матеріалу, натуральні одиниці вимірювання;

Ці - ціна за одиницю і-го типу матеріалу, грн.;

і - тип матеріального ресурсу;

n - кількість типів матеріальних ресурсів.

Таблиця 4.3 - Зведені розрахунки матеріальних витрат

Найменування матеріальних ресурсів	Од. виміру	Факт. витрачено матеріалів	Ціна за одиницю, грн.	Сума, грн.	Транспортні витрати (10% від суми)	Загальна сума, грн.
Мікросхеми	шт.	24	30	720	72	792
Папір (формат А4)	уп.	2	80	160	16	176
Р а з о м						968

4.1.4 Витрати на використання комп’ютерної техніки

Витрати на використання комп’ютерної техніки складаються з витрат на амортизацію комп’ютерної техніки, витрат на користування програмним

забезпеченням, витрат на електроенергію, що споживається комп'ютером. За даними обчислювального центру ТНЕУ для комп'ютера типу IBM PC/ATX вартість години роботи дорівнює 5,23 грн. Середній щоденний час роботи на комп'ютері – 2 години. Розрахунок витрат на використання комп'ютерної техніки приведений в таблиці 4.4.

Таблиця 4.4- Розрахунок витрат на використання комп'ютерної техніки

Назва етапів робіт, при виконанні яких використовується комп'ютер	Час використання комп'ютера, год.	Витрати на використання комп'ютера грн.
Проведення досліджень та оформлення їх результатів	60	313,8
Оформлення техніко-економічного розділу	8	41,84
Оформлення ДП	12	62,76
Разом	80	418,4

Якщо для розробки КС купується і монтується спеціальне обладнання, то необхідно врахувати також витрати на доставку і монтаж. Ці витрати (в залежності від складності монтажу) можуть бути прийняті у розмірі 10-25% від витрат на придбання обладнання.

4.1.5 Накладні витрати

Накладні витрати проектних організацій включають три групи видатків: витрати на управління, загальногосподарські витрати, невиробничі витрати.

Вони розраховуються за встановленими відсотками до витрат на оплату праці.

Середньостатистичний відсоток накладних витрат приймемо 150% від заробітної плати:

$$H = 1{,}5 \cdot 2837{,}45 = 4256{,}17 \ грн.$$

4.1.6 Інші витрати

Інші витрати є витратами, які не враховані в попередніх статтях. Вони складають 10% від заробітної плати:

$$I = 2837{,}45 \cdot 0{,}1 = 283{,}75 \ грн \ .$$

Витрати на розробку проектного рішення дорівнюють:

$$К_{ПР}=В_{ОП} + В_{Ф} + В_{М} +В_{ЕЛ} + Н + І,$$

$$К_{ПР} \ = 2837{,}45+465{,}34+968+418{,}4+4256{,}17+283{,}75 = 9265 \ грн.$$

На підставі отриманих даних за окремими статтями складається кошторис витрат на розробку КС за формою, наведеною в таблиці 4.5.

Таблиця 4.5 - Кошторис витрат на розробку, відлагодження та дослідну експлуатацію КС

Статті витрат	Сума, *грн*.
1. Матеріальні витрати, в тому числі: матеріали електроенергія 2.Витрати на оплату праці 3.Відрахування на соціальні потреби 4.Накладні витрати 5. Інші витрати.	 968 418,4 2837,45 465,34 283,75 4256,17
РАЗОМ по кошторису	9230

4.2 Визначення прогнозованої ціни

Величина можливої (договірної) ціни КС повинна визначатися з урахуванням ефективності, якості і термінів її виконання на рівні, що

відповідає економічним інтересам замовника (споживача) і виконавця. Договірна ціна ($Ц_Д$) для прикладних КС розраховується за формулою:

$$Ц_Д = B_{КС} \cdot \left(1 + \frac{p}{100}\right), \quad (4.4)$$

де $B_{КС}$ – кошторисна вартість КС, *грн.*;

p - середній рівень рентабельності КС, % (приймається в розмірі 20-30% за погодженням з керівником).

$$Ц_Д = 9230 \cdot 1.3 = 11999 \text{ грн.}$$

4.2.1 Економічне обґрунтування вибору комплексу технічних і програмних засобів

Для впровадження більшості КС необхідно:

- ✓ придбання та встановлення засобів комп’ютерної техніки;
- ✓ придбання та інсталяція системного програмного забезпечення;
- ✓ інсталяція і адаптація спеціалізованого програмного забезпечення

Кожен з перерахованих пунктів допускає безліч різних варіантів, так як існує велика кількість конфігурацій комп'ютерів, обладнання та різноманітних програмних продуктів. Кожен з варіантів передбачає різні за величиною і структурою витрати.

4.3 Розрахунок зведених економічних показників

Економічна ефективність – це співвідношення між отриманим прибутком та затраченими коштами. Вона обчислюється за формулою (4.6):

$$E_ф = П_р/K_В \quad (4.6)$$

де $П_р$ – очікуваний прибуток ;

$К_В$ – кошторисна вартість.

Очікуваний прибуток можна розрахувати із співвідношення:

$$П_Р = Ц_Д – К_В.$$

$$П_Р = 11999 – 9230 = 2769 грн.$$

Після проведених розрахунків отримуємо:

$$Е_ф = 2769/9230 = 0.3$$

Термін окупності додаткових капітальних вкладень визначається як :

$$Т = 1/Е_ф = 1/0.3 = 3.3 роки. \quad (4.7)$$

Таблиця 4.6 - Зведені економічні показники розробки

Показник	Значення
Собівартість, грн.	8771,93
Плановий прибуток, грн.	2631,58
Ціна, грн.	11403,51
Економічна ефективність	0,3
Термін окупності, рік	3,3

Провівши аналіз розрахованих значень економічних показників робимо висновок, що розробка апаратного засобу модулярного експоненціонування криптоалгоритму RSA є економічно доцільною.

ВИСНОВКИ

В результаті виконання дипломного проекту на тему "Апаратний засіб модулярного експоненціювання криптоалгоритму RSA" отримано наступні результати:

1. Проведено аналіз симетричних та несиметричних методів захисту інформації.
2. Відмічено перспективність криптоалгоритму RSA та актуальність його апаратної реалізації.
3. Розглянуто принципи функціонування засобів модулярного експоненціювання, обґрунтовано застосування методу пониження степеня та здійснено постановку завдання.
4. Розроблено структурну схему апаратного засобу модулярного експоненціювання методом пониження степеня.
5. Здійснено декомпозицію апаратного засобу модулярного експоненціювання та висвітлено питання функціонування схем ділення, множення, схеми віднімання та цифрового.
6. Розроблено структури компонентів апаратного засобу та методики їх налаштування.
7. Обґрунтовано вибір компонентів, розроблено та налаштовано електричні функційні схеми компонентів апаратного засобу в середовищі NI Multisim.
8. Техніко-економічне обґрунтування показало економічну доцільність розробки апаратного засобу модулярного експоненціювання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Закон України "Про захист інформації в інформаційно-телекомунікаційних системах" - [Електронний ресурс]- Режим доступу: http://zakon2.rada.gov.ua/laws/show/80/94-%D0%B2%D1%80 - Заголовок з екрану.

2. Грайворонський М. В. Безпека інформаційно-комунікаційних систем / М. В. Грайворонський, О. М. Новіков. – К. : Видавнича група BHV, 2009. – 608 с.

3. Ємець В. Сучасна криптографія. Основні поняття / В. Ємець, А. Мельник, Р. Попович. – Львів : Бак, 2003. – 144 с.

4. Основы информационной безопасности. Учебное пособие для вузов / Е.Б. Белов, В.П. Лось, Р.В. Мещеряков, А.А. Шелупанов - М .: Горячая линия - Телеком, 2006-544 с.

5. Основи інформаційної безпеки / С. В. Кавун, О. А. Смірнов, В. Ф. Столбов – Кіровоград : Вид. КНТУ, 2012. – 414 с.

6. Хорев А.А. Защита информации от утечки по техническим каналам. Технические каналы утечки информации. М.: Гостехкомиссия РФ, 1998. 320с.

7. Коробейников А. Г. Математические основы криптологии : учебн. пособ. / А. Г. Коробейников, Ю. А. Гатчин. – СПб. : СПб ГУ ИТМО, 2004. – 106 с.

8. Кузнецов О. О. Захист інформації в інформаційних системах / О. О. Кузнецов, С. П. Євсеєв, О. Г. Король. – Х. : Вид. ХНЕУ, 2011. – 512 с.

9. Остапов С. Е. Основи криптографії / С. Е. Остапов, Л. О. Валь. – Чернівці : Книги XXI, 2008. – 188 с. 34. Поповский В. В. Защита информации в телекоммуникационных системах : учебник / В. В. Поповский, А. В. Персиков. – Х. : ООО "Ком- пания СМИТ", 2006. – Т. 1. – 292 с.

10. Методичні рекомендації до виконання дипломного проекту з освітньо-кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп'ютерна інженерія» фахового спрямування «Комп'ютерні системи та мережі» / О.М.Березький, Л.О.Дубчак, Р.Б.Трембач, Г.М.Мельник, Ю.М.Батько, С.В.Івасьєв / Під ред. О.М.Березького. – Тернопіль: ТНЕУ, 2016. – 65с.

11. Методичні вказівки до написання техніко-економічного розділу для дипломних проектів на здобуття освітньо - кваліфікаційного рівня «Бакалавр» напряму підготовки 6.050102 «Комп'ютерна інженерія» / І.Р.Паздрій. - Тернопіль: ТНЕУ, 2018. – 36с.

12. Устройства умножения и деления - Режим доступу: http://naf-st.ru/arti. - Заголовок з екрану

13. Бабич М.П., Жуков І.А. Комп'ютерна схемотехніка: Навчальний посібник.- К.:МК-Прес, 2004.-412с

14. Схемотехніка електронних систем. Цифрова схемотехніка. Підручник / В.І. Бойко, А.М. Гуржій, В.Я Жуйков та ін.-К.:Вища школа, 2004.-423с.

15. Прянишников В.А. Электроника: Полный курс лекцій. – СПб.Корона принт; М.: Бином – Пресс, 2006.- 416.

16. Терехин В.Б., Соловьев Ю.А. Моделирование электронных схем в программе Electronics Workbench. Ч. 1. Создание схем. Ч.2. Элементная база: лабораторный практикум. – Северск: СТИ ТПУ, 2000. – 244 с.

17. Шило В.Л. Популярные цифровые микросхемы: Справочник. 2 - е изд., испр. – Челябинск: Металлургия, Челябинское отд., 1989.- 352 с.

18. Лещенко М.Є. Основи мікроелектроніки / М.Є. Лещенко, В.Є. Овчаренко. – Х. : Нац. аерокосм. ун-т „Харк. авіац. ін-т", 2005.

19. Комп`ютерна електроніка: Навч. посібник. Частина I/II А.П.Оксанич, С.Е.Притчин, О.В.Вашерук.- Харків: "Компанія СМІТ", 2006.- 200с/256с.

20. Рябенький В.М., Жуйков В.Я., Гулий В.Д. Цифрова схемотехніка: Навч. Посібник. - Львів: Видавництво «Новий світ 2000», 2009.-736с.

21. Резисторы, конденсаторы, трансформаторы, дроссели, комутационные устройства РЭА: Справочник./ Н.Н. Акимов, Е.П. Ващуков, В.А. Прохоренко, Ю.П. Ходоренок. – Мн.:Беларусь, 1994.- 591с.

22. Токхейм Р. Основы цифровой электроники.- М.:Мир, 1989.

23. Устройства умножения и деления. [Електронний ресурс] – Режим доступу - http://naf-st.ru/articles/digit/multidev/

24. Никитин В.А. Схемотехника интегральных схем ТТЛ, ТТЛШ и КМОП: Учебное пособие. М.: НИЯУ МИФИ, 2010. – 64 с.